

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Уваров Александр Алексеевич
Должность: Ректор
Дата подписания: 30.06.2023 17:10:13
Уникальный программный ключ:
711a9132de03714c5095fbf220ceaf18d7d7d5b5



**Частное образовательное учреждение
высшего образования
БАЛТИЙСКИЙ ГУМАНИТАРНЫЙ
ИНСТИТУТ**

**КАФЕДРА ОБЩИХ МАТЕМАТИЧЕСКИХ И
ЕСТЕСТВЕННО-НАУЧНЫХ ДИСЦИПЛИН**

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
по ПМ.01 Разработка модулей программного обеспечения
для компьютерных систем**

Программы подготовки специалистов среднего звена
09.02.07 Информационные системы и программирование

Настоящая программа разработана в соответствии с Законом Российской Федерации «Об Образовании в Российской Федерации» от 29 декабря 2012 г. № 273-ФЗ, на основе требований ФГОС СПО специальности 09.02.07 Информационные системы и программирование. (приказ Минобрнауки России от 09.12.2016 г. № 1547).

Составитель: канд.тех.наук, доцент, Кузьмин Константин Иванович

Рассмотрено и утверждено на заседании кафедры
15.02.2023 протокол № 6.

Одобрено учебно-методическим советом вуза
15.02.2023 протокол № 5.

Разработка программных модулей

ПАСПОРТ КОМПЛЕКТА ОЦЕНОЧНЫХ СРЕДСТВ

Перечень формируемых компетенций:

- ОК 1. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам
- ОК 2. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.
- ОК 3 Планировать и реализовывать собственное профессиональное и личностное развитие.
- ОК 4 Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.
- ОК 5 Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.
- ОК 6 Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей
- ОК 7 Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях.
- ОК 8 Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности
- ОК 9 Использовать информационные технологии в профессиональной деятельности.
- ОК 10 Пользоваться профессиональной документацией на государственном и иностранном языках
- ПК 1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием
- ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием
- ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств
- ПК 1.4 Выполнять тестирование программных модулей
- ПК 1.5 Осуществлять рефакторинг и оптимизацию программного кода
- ПК 1.6 Разрабатывать модули программного обеспечения для мобильных платформ

№ п/п	Контролируемые разделы (темы)	Код контролируемой компетенции (или её части)	Наименование оценочного средства
1.	Структурное программирование	ОК 1 ОК 2 ОК 4	Лабораторная работа №1
	Оценка сложности алгоритмов сортировки, поиска, рекурсивных алгоритмов, эвристических алгоритмов	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5	
	Самостоятельная работа	ОК 1 ОК 2 ОК 4 ОК 9 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5	Творческое задание №1

2.	Объектно-ориентированное программирование	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5	Лабораторная работа №2
	Разработка приложения с классами. Разработка приложения с типом данных структура, коллекция, список. Использование регулярных выражений для валидации ввода данных	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6	Лабораторная работа №3-7
	Самостоятельная работа	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6	Творческое задание №2
3.	Паттерны проектирования	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6	Лабораторная работа №4
	Создание приложений на основе основных, порождающих, структурных и поведенческих шаблонов.	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6	Лабораторная работа №8-9
	Самостоятельная работа	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6	Практическое задание №3
4.	Событийно-управляемое программирование: разработка WPF-приложения с использованием для решения простых расчетных задач. разработка игрового приложения. разработка приложения с анимацией.	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 9 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5	Лабораторная работа №10-12, самостоятельная работа №1, контрольная работа №1, самостоятельная работа №2, контрольная работа №2,
	Самостоятельная работа	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 9 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5	Сообщение №1
5.	Оптимизация и рефакторинг кода	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6	Лабораторная работа №13, самостоятельная работа №3
	Самостоятельная работа	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6	Сообщение №2

6.	Разработка пользовательского интерфейса.	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6	Лабораторная работа №14, самостоятельная работа №4, контрольная работа №3
	Самостоятельная работа	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6	Сообщение №3
9.	Основы ADO.Net	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6	Лабораторная работа №15
10.	Промежуточная аттестация	ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6	Экзамен

Сформированность выше перечисленных компетенций предполагает, что в результате освоения дисциплины (профессионального модуля) обучающийся должен:

знать:

- основные этапы разработки программного обеспечения;
- основные принципы технологии структурного и объектно-ориентированного программирования;
- способы оптимизации и приемы рефакторинга;
- основные принципы отладки и тестирования программных продуктов

уметь:

- осуществлять разработку кода программного модуля на языках низкого и высокого уровней;
- создавать программу по разработанному алгоритму как отдельный модуль;
- выполнять отладку и тестирование программы на уровне модуля;
- осуществлять разработку кода программного модуля на современных языках программирования;
- уметь выполнять оптимизацию и рефакторинг программного кода;
- оформлять документацию на программные средства

иметь навыки и (или) опыт деятельности (владеть):

- в разработке кода программного продукта на основе готовой спецификации на уровне модуля;
- использовании инструментальных средств на этапе отладки программного продукта;
- проведении тестирования программного модуля по определенному сценарию;

- использовании инструментальных средств на этапе отладки программного продукта;
- разработке мобильных приложений

I ТЕКУЩИЙ КОНТРОЛЬ ПО МЕЖДИСЦИПЛИНАРНОМУ КУРСУ

➤ ОЦЕНОЧНОЕ СРЕДСТВО: Лабораторная работа №1

1. **Цель:** Формирование системы знаний об основных возможностях языка C# и базовых средствах и структурах данных для создания программ.

2. **Проверяемые компетенции** (код): ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5

3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

Составить программы для решения следующих задач в виде консольного приложения на языке C# в среде Microsoft Visual Studio и оценить их сложность:

1. Даны две целые переменные a, b. Составить программу, после исполнения которой значения переменных поменялись бы местами (новое значение a равно старому значению b и наоборот).

2. Даны целые числа a, b, c. Найти максимальное из этих трех чисел.

3. Даны натуральное n и действительные числа a_1, \dots, a_n . Все члены этой последовательности, начиная с первого положительного, уменьшить на 0,5.

4. Даны два массива $x[1] \leq x[2] \leq \dots \leq x[k]$ и $y[1] \leq y[2] \leq \dots \leq y[n]$. Соединить их в массив $z[1] \leq \dots \leq z[m]$ ($m = k+n$), каждый элемент должен входить в массив z столько раз, сколько раз он входит в общей сложности в массивы x и y).

5. Даны 2 двумерных массива найти наибольший элемент, встречающийся в обоих массивах.

6. Дан текст. Найти наибольшее количество цифр, идущих в нем подряд.

7. Считать текст из текстового файла, каждую строку, длина которой превышает заданное число заменить на «!!!»

4. Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарий к программному коду, сделаны выводы по работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно решена только часть заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

➤ ОЦЕНОЧНОЕ СРЕДСТВО: Творческое задание №1

1. **Цель:** Формирование системы знаний об основных возможностях языка C# и базовых средствах, и структурах данных для создания программ.

2. **Проверяемые компетенции** (код): ОК 1 ОК 2 ОК 4 ОК 9 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5

3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

Оптимизировать по выбору 2 из ранее разработанных программ для следующих задач:

1. Даны два массива $x[1] \leq x[2] \leq \dots \leq x[k]$ и $y[1] \leq y[2] \leq \dots \leq y[n]$. Соединить их в массив $z[1] \leq \dots \leq z[m]$ ($m = k+n$), каждый элемент должен входить в массив z столько раз, сколько раз он входит в общей сложности в массивы x и y .

2. Даны 2 двумерных массива найти наибольший элемент, встречающийся в обоих массивах.

3. Дан текст. Найти наибольшее количество цифр, идущих в нем подряд.

4. Считать текст из текстового файла, каждую строку, длина которой превышает заданное число заменить на «!!!»

Выбор объяснить.

4. **Критерии оценивания:**

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарии к программному коду, выполнено объяснение к работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарии к программному коду. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарии к программному коду. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно решена только часть заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

➤ ОЦЕНОЧНОЕ СРЕДСТВО: Лабораторная работа №2

1. **Цель:** Формирование системы знаний об основных принципах проектирования и разработки приложений с использованием классов и объектов.

2. **Проверяемые компетенции** (код): ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5

3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

Объявить класс и реализовать программу для решения следующей задачи:

Имеются сведения об автомобилях и их владельцах:

- Государственный регистрационный номер.
- Модель.
- Цвет.
- ФИО владельца.

Необходимо вводить указанные данные, хранить введенную информацию и удалять сведения об автомобиле и его владельце по запросу.

4. Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарий к программному коду, сделаны выводы по работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно решена только часть заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

➤ ОЦЕНОЧНОЕ СРЕДСТВО: Лабораторная работа №3

1. **Цель:** Формирование системы знаний об основных принципах проектирования и разработки Windows- приложений.

2. **Проверяемые компетенции (код):** ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6

3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

1. Реализовать программу, эмулирующую внешний вид экранной формы приложения *Microsoft Office Word*: окно *Печать*. Для этого:

1. Проанализировать заданную форму и выделить основные визуальные и не визуальные компоненты, используемые в ней;
2. Используя стандартные компоненты *Windows Forms*, как можно ближе, повторить внешний вид диалоговых окон или приложений;
3. Полученная эмуляция окна должна быть полностью работоспособной (то есть меню отображаться, кнопки нажиматься, пункты выбираться);
4. Кнопки, не выполняющие никаких действий, должны выдавать соответствующее сообщение;
5. Необходимо эмулировать не только главную форму, но и все подчиненные;
6. Окна с отображением файлов, наполнять по своему усмотрению;

7. После реализации экранной формы, уметь описать работу каждого из используемых компонентов.

2. Используя компонент «кнопка», изучить событийную модель C#

Для этого:

1. Разместить на форме четыре кнопки.
2. Щелчок на форме и <ALT-I> восстанавливают начальное состояние кнопок.
3. Двойной щелчок и <ALT-X> – закрывают форму. Предусмотреть вывод подтверждения закрытия.
4. Для каждой кнопки определить подсказку (Hint), описывающую функцию кнопки, причем подсказка появляется не стандартным образом, а вместе с именем и заголовком кнопки в специальной области формы при попадании указателя мыши на кнопку и гаснет при уходе указателя с кнопки.
5. С каждой из кнопок связать определенное действие, описанное ниже:

1-я кнопка прячет / показывает 2-ю;

2-я кнопка (учесть реальные ограничения, связанные с размером формы) сдвигает первую кнопку на 10 пикселей вверх;

3-я кнопка (1 – сама по себе, 2 – с Shift, 3 – с Ctrl) 1-увеличивает, 2-уменьшает, 3-переключает по кругу (из 3-х) шрифты на форме;

4-я кнопка вкл./выкл. системную кнопку.

3. Изучить основы работы с текстовой информацией в C#. Для этого:

1. Создать алгоритм решения следующей задачи: в данной строке подсчитать, сколько раз встречается определенная буква. Реализовать задачу средствами языка C#.
2. При выполнении задания обязательно использовать следующие компоненты: TextBox (ввод данных), Memo и TextBox (вывод результатов), Label (вывод пояснений), Menu (главное меню программы), Button (кнопка расчета, выход из программы).

3. Ввод данных в программу и необходимый расчет осуществлять после нажатия кнопки Enter в поле компонента ввода (TextBox) или нажатии на компонент Button.
 4. С помощью главного меню реализовать очистку компонентов от введенных данных и продублировать основные действия программы.
 5. Предусмотреть подтверждение закрытия приложения.
 6. В модальном окне вывести информацию о создателе программы.
4. Реализовать приложение для решения следующей задачи: Используя интерфейс программы, задать одномерный статический массив из N элементов произвольного значения. Провести поиск заданного по значению элемента массива методом простого перебора.

Для этого:

1. Входные, промежуточные и выходные данные организовать, используя компонент DataGridView.
2. Ввод данных реализовать тремя способами: вручную, случайно и из файла. Для выбора одного из вариантов ввода использовать компоненты RadioButton или ComboBox.
3. Все необходимые переменные для работы программы, должны быть запрошены через компоненты TextBox.
4. Результаты работы приложения отобразить на экране и сохранить в текстовый файл.
5. В модальном окне вывести информацию о разработчике.
6. Предусмотреть подтверждение закрытия приложения.
- 7.

4. Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написаны комментарии к программному коду, сделаны выводы по работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написаны комментарии к программному коду. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написаны комментарии к программному коду. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно решена только часть заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

➤ ОЦЕНОЧНОЕ СРЕДСТВО: Творческое задание №2

1. **Цель:** Формирование системы знаний об основных возможностях языка C# и базовых средствах, и структурах данных для создания программ.

2. **Проверяемые компетенции (код):** ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6

3. Пример оценочного средства (примерные тестовые задания, типовой вариант контрольной работы и др.)

Оптимизировать по выбору 2 из ранее разработанных программ лабораторной работы №3: выбрать одну из задач 1 или 2, вторую – одну из задач 3 или 4.

Выбор объяснить. Под оптимизацию могут попадать как программные алгоритмы, так и интерфейсы приложений.

4. Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарий к программному коду, выполнено объяснение к работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно решена только часть заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

Создать клиент-серверное приложение, позволяющее вести с пользователем игру в слова: пользователь называет слово, от сервера приходит слово, начинающееся на последнюю букву названного слова. Далее пользователь должен предложить слово, начинающееся с буквы, которой заканчивалось слово от сервера и т.д.

4. **Критерии оценивания:**

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарий к программному коду, сделаны выводы по работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно решена только часть заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

➤ **ОЦЕНОЧНОЕ СРЕДСТВО:** Лабораторная работа №5

1. **Цель:** Формирование системы знаний об основных принципах проектирования и разработки приложений, осуществляющих работу с базами данных, хранящихся на некотором локальном сервере.

2. **Проверяемые компетенции** (код): ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6

3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

Создать базу данных на сервере MySQL для хранения информации об участниках спортивных соревнований: наименование страны, название команды, Ф.И.О. игрока, игровой номер, возраст, рост, вес, а также реализовать обработку данных в базе из приложения WPF, позволяющую вывести информацию о самой молодой команде

4. **Критерии оценивания:**

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарий к программному коду, сделаны выводы по работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении

	присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно решена только часть заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

➤ **ОЦЕНОЧНОЕ СРЕДСТВО: Практическая работа №3**

1. **Цель:** Формирование системы знаний об основных возможностях языка C# и базовых средствах, и структурах данных для создания программ.

2. **Проверяемые компетенции (код):** ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6

3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

Доработать программный код предыдущей работы, оптимизировать запросы к базе данных, используя информацию дополнительной литературы.

4. **Критерии оценивания:**

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарий к программному коду, выполнено объяснение к работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно решена только часть заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

➤ **ОЦЕНОЧНОЕ СРЕДСТВО Лабораторная работа 6**

1. **Цель:** формирование системы знаний о html и об основных свойствах объектов в css

2. Проверяемые компетенции (код): ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 9 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5

3. Пример оценочного средства (*примерные тестовые задания, типовой вариант контрольной работы и др.*)

1. Создайте html файл, в который скопируйте любой текст. Расставьте по тексту заголовки всех 6 уровней (им не обязательно быть связанными с содержанием текста).

Создайте стилевой файл (с расширением css) со следующими параметрами

- выравнивание текста по ширине
- размер текста body 14px, h2-180% h3-160%, h4-140%, h5-120%, h6-100%
- отступы заголовков и всех абзацев после заголовка каждого уровня h2-10 h3-20, h4-30, h5-40, h6-50, всё в пикселях.
- отступы между заголовками и текстом слишком большие. Воспользуйтесь инструментами разработчика, чтобы выяснить причину.
- сделайте так, чтобы между заголовком и текстом было 5 пикселей, а между заголовком и текстом выше 10 пикселей.

2. Создайте новую страницу. Примените к ней следующие параметры

- фоновое изображение, на котором обои или стена.
- несколько блоков разного размера (3-4) внутри картинок.
- блоки располагаются так, чтобы казалось, что они расположены в произвольном порядке, как фото на стене.
- картинки имеют рамки различной толщины и цвета.

3. Создайте файл, в котором будет следующее:

- Несколько изображений с подписями, которые располагаются по 3 или 4 в ряд, ряда хотя бы 2.
- При наведении курсора на подпись его внешний вид меняется и появляется описание изображения и ссылка.
- Ссылки и описания под каждой картинкой ведут на отдельную страницу, на которой располагается таблица.
- Таблица заполнена и оформлена на ваше усмотрение. Оставить стиль по умолчанию нельзя.

4. Скопируйте в свою папку файл с заданиями и стилевые файлы, которые в нём использованы.

Исправьте список с заданиями: точки должны быть внутри блока залитого серым. Создайте для него отдельный класс, чтобы список в шапке не изменился.

1. Измените цвета ссылок (до нажатия и после) на те, что больше подходят.
2. Выясните, почему серая область не продолжается до конца текста и исправьте это.

4. Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, внешний вид страниц и компонентов, используемых на страницах соответствуют заданию. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно выполнена только часть заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

5. Рекомендуемый перечень вопросов для самостоятельной подготовки:

1. Синтаксис css
2. Блочная модель документа
3. Цвета
4. Шрифты
5. Положение элемента на странице
6. Идентификаторы, классы, псевдоклассы

➤ **ОЦЕНОЧНОЕ СРЕДСТВО** Самостоятельная работа №1

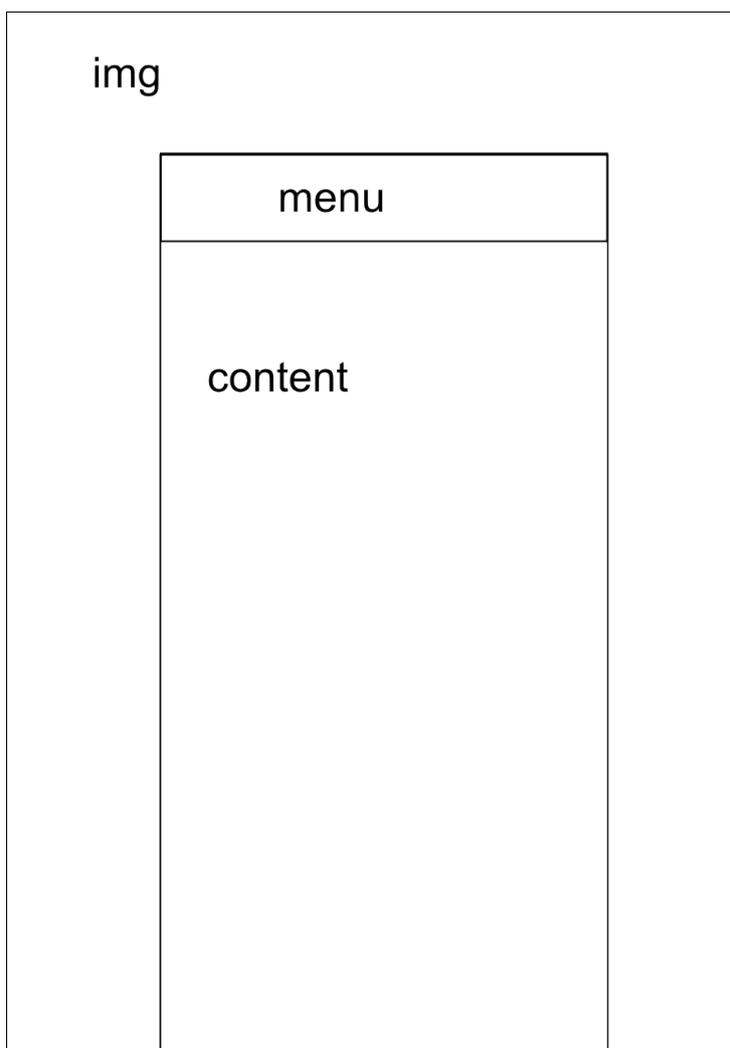
1. **Цель** формирование навыков вёрстки web-страниц.

2. **Проверяемые компетенции (код):** ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 9 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5

3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

Реализовать html страницу со следующим содержимым:

- Границ блоков нет;
- Фоновое изображение на всю ширину окна;
- Блок «контент» расположен по середине окна, ширина 750px, отступ от верхнего края окна 200px цвет блока rgb(250, 184, 187), есть тень;
- Цвет текста rgb(58, 58, 58). Заголовки подчёркиваются, цвет заголовков чёрный;
- Меню располагается в верхней части контента, состоит из 5 ссылок, выровненных посередине. Цвет ссылок rgb(237, 28, 36), шрифт жирный.



4. **Критерии оценивания:**

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, внешний вид страниц и компонентов, используемых на страницах соответствуют заданию. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно выполнена только часть заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

5. Рекомендуемый перечень вопросов для самостоятельной подготовки:

1. Синтаксис css
2. Блочная модель документа
3. Цвета
4. Шрифты
5. Положение элемента на странице
6. Идентификаторы, классы, псевдоклассы

➤ **ОЦЕНОЧНОЕ СРЕДСТВО** Контрольная работа №1

1. Цель: проверка уровня усвоения темы «Логическая разметка страницы и таблицы стилей»

2. Проверяемые компетенции (код): ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 9 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5

3. Пример оценочного средства (*примерные тестовые задания, типового варианта контрольной работы и др.*)

1. Зачем нужны теги, введённые в html 5 (article, footer и другие)?
2. Как будут выглядеть элементы расположенные внутри них?
3. html и css используются при создании web-страниц. Какова роль каждого из них?
4. Найдите и исправьте ошибки

```

p{color red
margin: 10 ;
}

```
5. В каком формате могут обозначаться цвета в css?
6. Зачем перечислены 3 шрифта?

```

body{
font-family: Verdana, Geneva, Arial, sans-serif;}

```

 - a) они все будут присутствовать на странице
 - b) они применятся в порядке перечисления, если не окажется первого будет применён второй
 - c) так не пишут, должен быть один шрифт
7. Обязательно ли в конце указывать семейство шрифтов? Почему?
8. Напишите комментарии к каждой строке. (к чему применятся правила)

```

h1, h2, p, q{}

```

- h2+p {}
h2~p {}
9. Как создать класс в css? Приведите пример. Нужно ли что-нибудь написать в html-файле?
 10. Как создать идентификатор в css? Приведите пример. Нужно ли что-нибудь написать в html-файле?
 11. Что такое наследование?
 12. Какие элементы будут иметь красный цвет текста?

```

<html>
<body>
<p>текст абзаца </p>
<form> <p> текст формы</p>
<input ... />
</form>
<p> текст абзаца <q> текст цитаты</q> текст абзаца </p>
<p class="class1"> текст абзаца <q> текст цитаты</q> </p>
</body>
</html>
css:
body {
color: #ff0000;
}
class1 {
color: blue;
}

```
 13. Какого цвета будет текст?

```

body{
color: red;}
p,h2,h3{
text-align: center;
color: blue;}
p{color: black;}

```

 - a) чёрный
 - b) красный
 - c) синий
 14. Текст какого цвета будет присутствовать на странице?

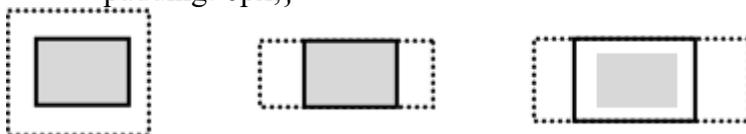
```

css:
p{ color: black};
.newclass{ color: red;}
p.newclass{ color: green;}
<body>
<p>...</p>
<p class=" newclass">...</p>
<p>...</p>
<p>...</p>
</body>

```

 - a) чёрный
 - b) чёрный и зелёный
 - c) зелёный и красный
 - d) все
 15. Опишите суть и назначение блочной модели.
 16. Выберите соответствующее коду изображение

```
div{
border: solid black 2px;
width: 400px;
margin: 0 20px;
padding: 0px;}
```



17. Что такое псевдокласс?
18. Подпишите комментарии

```
.button:hover{
background: #666666;
cursor: pointer;
}
.button:active {
background: #333333;
}
```

19. Позиция элемента может быть абсолютной и относительной, относительно чего?

4. Критерии оценивания (по 100-балльной системе оценивания):

Весовые коэффициенты по мере сложности, накладываемые на вопросы:

Вопросы 1, 2, 3, 5, 7, 9, 10, 11, 17, 19 – 0,3

Вопросы 4, 6, 8, 12, 13, 14, 16 – 0,7

Вопрос 15 – 0,11

Вопрос 18 – 0,1.

Каждый вопрос оценивается по 100 шкале, результат ответа на вопрос умножается на указанный коэффициент, далее полученные таким образом баллы по каждому вопросу суммируются

85-100 баллов (отлично)	Суммарно набрано 85-100 баллов (отлично)
70-84 баллов (хорошо)	Суммарно набрано 70-84 баллов (хорошо)
50-69 баллов (удовлетворительно)	Суммарно набрано 50-69 баллов
0-49 баллов (неудовлетворительно)	Суммарно набрано 0-49 баллов (неудовлетворительно)

5. Рекомендуемый перечень вопросов для самостоятельной подготовки:

1. Синтаксис css
2. Блочная модель документа
3. Цвета
4. Шрифты
5. Положение элемента на странице
6. Идентификаторы, классы, псевдоклассы

➤ ОЦЕНОЧНОЕ СРЕДСТВО: Сообщение №1

1. **Цель:** формирование системы знаний о html и об основных свойствах объектов в css

2. **Проверяемые компетенции (код):** ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 9 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5

3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

Подготовить сообщения на тему «Отличие html и html5», «Отличие css и css3», преимущества и недостатки одних технологий перед другими.

4. Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении и выводах присутствуют незначительные недочеты, получены достоверные результаты. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно описана только часть информации, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

➤ **ОЦЕНОЧНОЕ СРЕДСТВО** Лабораторная работа №7

1. **Цель** Формирование системы знаний о синтаксисе языка php и об основных принципах проектирования и разработки программ с использованием языка php

2. **Проверяемые компетенции (код):** ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6

3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

Введение

- Выведите на экран "Hello,World!".
- Зайдите в Z:\WebServers\home, создайте там папку назовите своей фамилией, в ней папку www а в ней файл с расширением index.php.
- Наберите в файле следующее. Сохраните файл.

```
<?php // начало php-кода
$n="World";// присваивание значения переменной
echo "Hello, ".$n "!"; // вывод на экран, оператор "." соединяет строки
?>//конец php-кода
```

- В адресной строке браузера наберите имя созданной папки и нажмите Enter.

Задания:

Результаты выполнения всех заданий должны выводиться на экран с пояснениями. Для квадратного уравнения: «решение уравнения $a*x^2+b*x+c$: $x_1=...$, $x_2=...$ » или «вещественных корней нет»

Числа и строки сначала выберите самостоятельно, а затем при сдаче работы подставьте предложенные преподавателем.

1. Выведите на экран "Hello, Vasia!"(или другое имя).
2. Решите квадратное уравнение.
 - a. Значения коэффициентов a, b, c выберите самостоятельно.
 - b. Передайте значения коэффициентов через адресную строку. Например, `Ivanov.php?/a=1&b=2&c=3`
3. Вычислите объём шара, выведите результат на экран.
4. Складывайте числа, начиная с n, до тех пор, пока сумма не достигнет S. Выведите результат на экран.
5. Вычислите x% от числа y. Выведите результат на экран.
6. Разделите строку на 4 части и выведите каждую из них в отдельную строку.
7. Разделите строку по знаку @.
8. Сравните, совпадают ли строки.
9. Расположите слова в алфавитном порядке.
10. Добавьте в текст символы перехода на новую строку.
11. Замените в тексте символы перехода на новую строку используемые с linux на те что используются в windows.

4. Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарий к программному коду, выполнено объяснение к работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно решена только часть заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

5. Рекомендуемый перечень вопросов для самостоятельной подготовки:

1. Синтаксис php
2. Функции работы со строками
3. Условный оператор
4. Циклы

➤ **ОЦЕНОЧНОЕ СРЕДСТВО** Лабораторная работа №8

1. **Цель:** Формирование системы знаний об основных принципах проектирования и разработки html-форм

2. **Проверяемые компетенции (код):** ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6

3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

0. Создайте форму для ввода данных.

0.1. Запустите denver. Сохраняйте все файлы на Z:\Webservers\home\www\

0.2. Создайте html-файл с формой, код которой приведён ниже.

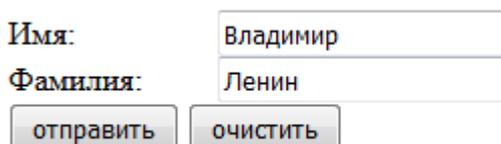
```
<form action="n.php" method="post">
<label id="q" for="firstname">Имя:</label>
<input type="text" value="Владимир" name="firstname"/> </br>
<label id="q" for="lastname" >Фамилия:</label>
<input type="text" value="Ленин" name="lastname"/> </br>
<input type="submit" value='отправить'>
</form>
```

0.3. Создайте стилевой файл, в который запишите

```
#q {
width: 100px;
display: inline-block;
}
```

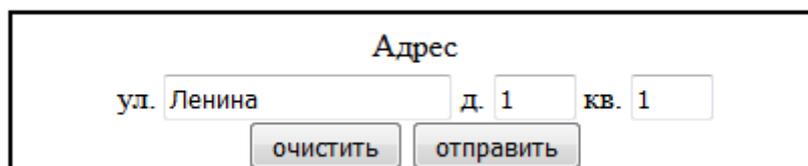
0.4. Создайте файл n.php

```
<?php
$n=$_POST['firstname'];
echo 'Ваше имя:'. $n;
$m=$_POST['lastname'];
echo 'Ваше имя:'. $m;
?>
```



1. Аналогично создайте следующие формы. Создайте стилевой файл,

1.1. Форма для ввода адреса. После нажатия на кнопку отправить появляется надпись, состоящая из отправленных значений и добавленных вами слов. Например: "ваш адрес ул. Ленина д. 1 кв. 1"



1.2. Форма для ввода коэффициентов уравнения. Соедините форму с файлом, который решает уравнение.

1.3. Форма для выбора 1 варианта. После нажатия на кнопку "отправить" должна появиться фраза: "вы выбрали вариант ..."

1.4. Форма для выбора нескольких вариантов. После нажатия на кнопку «отправить» должна появиться фраза: «вы выбрали ...»

2. Создайте поле для ввода текста. После отправки он должен отобразиться в новом окне со следующим форматированием.
 - а) Каждое предложение с новой строки.
 - б) Каждая строка, которая в поле начинается с символа "!" становится заголовком.
 - с) Первые n символов/слов текста.

4. Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарий к программному коду, формы и компоненты на формах соответствуют заданию по расположению и их функциональным возможностям, выполнено объяснение к работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении формы и компонентов на формах присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы в том числе о реализации той или иной функции.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно решена только часть заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

5. Рекомендуемый перечень вопросов для самостоятельной подготовки:

1. Поля для ввода текста
2. Элементы выбора
3. Загрузка файла
4. Кнопки

➤ **ОЦЕНОЧНОЕ СРЕДСТВО** Самостоятельная работа №2

1. **Цель** проверка уровня умения создания html-форм и построения логики проекта
2. **Проверяемые компетенции (код):** ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6

3. **Пример оценочного средства** (*примерные тестовые задания, типовой вариант контрольной работы и др.*)

Создайте страницу с тестом из 5 вопросов (можно больше). Должны быть вопросы с выбором как одного, так и нескольких вариантов ответов, и хотя бы один вопрос, где ответ нужно написать.

Создайте файл, который обрабатывает ответы и выдаёт результат, в зависимости от количества правильных ответов.

Создайте стилевой файл.

Вопросы придумайте самостоятельно. Наличие изображений приветствуется.

4. Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарий к программному коду, формы и компоненты на формах соответствуют заданию по расположению и их функциональным возможностям, выполнено объяснение к работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении формы и компонентов на формах присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы в том числе о реализации той или иной функции.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно решена только часть заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

5. Рекомендуемый перечень вопросов для самостоятельной подготовки:

1. Поля для ввода текста
2. Элементы выбора
3. Загрузка файла

4. Кнопки

➤ **ОЦЕНОЧНОЕ СРЕДСТВО** Лабораторная работа №9

1. **Цель:** формирование знаний о регулярных выражениях и умений по их использованию

2. **Проверяемые компетенции (код):** ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6

3. **Пример оценочного средства** (*примерные тестовые задания, типовой вариант контрольной работы и др.*)

1. Добавить в форму с именем и фамилией проверку того, что туда не вводятся цифры.
2. Добавить в форму с адресом проверку того, что в поле «улица» вводятся только буквы и цифры, в поля «номер дома» и «квартира» – только цифры.
3. Добавить в форму с уравнением проверку того, что вводятся только цифры.
4. Создайте форму в которую, будет вводиться номер телефона в формате +7(***)*пробел*****пробел*****. В ответ должно выводиться сообщение «ваш номер +71234567890» – только цифры
5. Проверьте, является ли данная строка адресом электронной почты.

4. Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарий к программному коду, формы и компоненты на формах соответствуют заданию по расположению и их функциональным возможностям, выполнено объяснение к работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении формы и компонентов на формах присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы в том числе о реализации той или иной функции.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно решена только часть заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

5. Рекомендуемый перечень вопросов для самостоятельной подготовки:

1. Символы, используемые в выражениях
2. Функции `php` для обработки выражений

➤ **ОЦЕНОЧНОЕ СРЕДСТВО** Лабораторная работа №10

1. **Цель** формирование знаний и умений строить запросы к базе данных

2. **Проверяемые компетенции (код):** ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9
ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6

3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

Создание базы данных и запросов по образцу.

1. После запуска Денвера наберите в адресной строке 'localhost'. Перейдите по ссылке <http://localhost/Tools/phpMyAdmin>. Можно сразу в адресной строке набрать этот адрес.
2. Создайте базу. Имя может выбрать любое.
3. Создайте таблицу, в которой будет 3 столбца : name, phone, home_office(личный или рабочий номер). Тип всех столбцов VARCHAR. Длина name – 20, phone – 12, home_office – 1.
4. Заполните таблицу.
5. Создайте php-файл, следующего содержания

```
<?php
$a=mysqli_connect('localhost', 'root', '', 'base1') or die('connection error');
$q="SELECT * FROM table1;";
$r=mysqli_query($a, $q) or die('error:'.mysqli_error());
while($n=mysqli_fetch_array($r)){
    echo $n['name'].' '.$n['phone'].'<br>';
}
mysqli_close($a);
?>
```

6. Создайте html-файл, а в нём ссылку

```
<a href="base.php">список контактов</a> (base.php – имя созданного на предыдущем шаге файла)
```

Проверьте, что ссылка работает

7. Создайте другой php-файл, который будет выводить на экран контакты, где номер отмечен как домашний (личный) – h или рабочий – o.

```

<p>
<ul>
<li><a href="base2.php/?k='o'">рабочий</a>
<li><a href="base2.php/?k='h'">домшний</a>
</ul>
</p>

```



СПИСОК КОНТАКТОВ

- рабочий
- домшний

```

<?php
$k=$_GET['k'];
$a=mysqli_connect('localhost', 'root', '', 'base1') or die ('ошибка соединения с базой');
$q="SELECT * FROM table1 WHERE home_office=$k;";
$r=mysqli_query($a, $q) or die ('ошибка запроса: ' . mysqli_error());
while($n=mysqli_fetch_array($r)){
echo '<p>'. $n[name]. " " . $n[phone]. '</p>';
}
?>

```

8. Создайте форму для поиска по имени

```

<?php
$find_name=$_POST['find_name'];
$a=mysqli_connect('localhost', 'root', '', 'base1') or die ('ошибка соединения с базой');
$q="SELECT * FROM table1 WHERE name='$find_name';";
$r=mysqli_query($a, $q) or die ('ошибка запроса: ' . mysqli_error());
while($n=mysqli_fetch_array($r){
echo $n[name]. " " . $n[phone]. "<br>";
}

```

9. Создайте форму для добавления нового контакта и php-файл, который выполняет этот запрос к базе
10. Сделайте возможность изменить имя
11. Сделайте возможность удалить контакт

4. Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарий к программному коду, формы и компоненты на формах соответствуют заданию по расположению и их функциональным возможностям, выполнено объяснение к работе. При этом выполнены все задания (или за исключением одного: 10 или 11-ого) Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении формы и компонентов на формах присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. При этом выполнены задания 1-8 и одно из 9-11. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. При этом выполнены задания 1-8. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы в том числе о реализации той или иной функции.
Оценка	Работа не всегда защищена с первого раза, верно решена только

«неудовлетворительно»	часть заданий 1-8 или решено меньшее количество заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.
-----------------------	---

5. Рекомендуемый перечень вопросов для самостоятельной подготовки:

1. Функции соединения с базой
2. SQL запросы
3. Отправка запроса
4. Обработка запроса

➤ **ОЦЕНОЧНОЕ СРЕДСТВО** Контрольная работа №2

1. Цель проверка уровня усвоения темы «Динамическая генерация страницы сервером и обработка запросов к базе данных»

2. Проверяемые компетенции (код): ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9
ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6

3. Пример оценочного средства (примерные тестовые задания, типовой вариант контрольной работы и др.)

1. Найдите ошибки в каждом из трёх примеров

a) \$str = "6476;bcvxfvsgjjtx56c5c";

n = strlen(\$str)/4;
substr(\$str, \$n)

b) \$a= hello, world!;
echo \$a;

c) \$d='D';
\$f='F';
\$w=\$d+\$f;

2. Чему равно \$c если \$a = 5, \$b = 8?

```
if($a>$b){
    $c=$a-$b;
echo $c;
}
elseif($a==$b){
echo 0;
}
else {
$c=$b-a;
echo $c;
```

3. Чем отличаются элементы <input type="submit"> и <input type="reset">?

4. Что пропущено? <form action="..." method="post">

- a) A.php
- b) A.txt
- c) A.html

5. За что отвечает атрибут action тега form?

6. В чём главное отличие методов post и get?

- a) Принципиально ничем: данные всё равно передаются на сервер

- b) GET передаёт данные, используя стартовую строку, а POST, используя тело http запроса
- c) Отличие в используемых конструкциях языка (массивы разные)
7. Отметьте верные утверждения
- a) С помощью метода post можно передать больше данных? чем с помощью get
- b) Количество передаваемых данных ограничено только сервером
- c) Один из этих методов более надёжен, при другом данные могут потеряться
- d) Данные передаваемые через адресную строку более уязвимы
- e) Верных утверждений нет
8. equation.php/a=3&b=5 Что должно быть в файле
- a) `x = $_POST['a'];`
`y = $_POST['b'];`
- b) `x = $_GET['a'];`
`y = $_GET['b'];`
- c) `a = GET['a'];`
`b = GET['b'];`
- d) всё не верно
9. Форма реализована следующим кодом:

```
<form action="X.php" method="post">
<label id="q" for="firstname">Имя:</label>
<input type="text" value="Владимир" name="firstname"/> </br>
<label id="q" for="lastname" >Фамилия:</label>
<input type="text" value="Ленин" name="lastname"/> </br>
<input type="submit" value='отправить'>
</form>
```

В каком из файлов написан код, который передаст данные серверу?

- a) File.php
- ```
<?php
$n=$_POST['firstname'];
echo 'Ваше имя:'. $n;
$m=$_POST['lastname'];
echo 'Ваше имя:'. $m;
?>
```
- b) X1.php
- ```
<?php
$name1=$_POST['name1'];
echo 'Ваше имя:'. $ name1;
$name2=$_POST['name2'];
echo 'Ваше имя:'. $ name2;
?>
```
- c) Name.php
- ```
<?php
$n=$_POST['firstname'];
echo 'Ваше имя:'. $firstname;
$m=$_POST['lastname'];
echo 'Ваше имя:'. $lastname;
?>
```
- d) X.php
- ```
<?php
$n=$_POST['firstname'];
```

```

echo 'Ваше имя:'. $n;
$m=$_POST['lastname'];
echo 'Ваше имя:'. $m;
?>

```

10. Выберите строки соответствующие выражению `/^\d{3}-\d{3}-\d{4}$/`
- (555) 935-2659;
 - (555)672-0953;
 - 555-343-8263;
 - 55-441-9005;
 - 555.903.6386;
 - 555-612-8527-8724
11. Что можно сказать про это выражение `/[1-9]\d{2}-\d{3}-\d{4}$/`
- Должны быть только цифры
 - Номер не должен начинаться с 0
 - Должно быть 10 цифр разделённых тире
12. Какое значение вернёт функция `preg_match('/^\(\d{3}\)\d{3}-\d{4}/', '123 4567')`
13. Укажите правильный порядок вызова функций
- `mysqli_connect`
 - `mysqli_close`
 - `mysqli_query`
14. Какие аргументы принимает функция `mysqli_connect`?
15. Что будет, если не использовать `mysqli_close()` :
16. Допишите недостающие части

```

mysqli_connect('localhost', 'root', '', 'base1') or die ('ошибка соединения с базой');
$q="INSERT INTO test (name, number) VALUES ('Даша', '345678');";
$r=mysqli_query($a, $q) or die ('ошибка обращения запроса к базе: ' . mysql_error($a));
mysqli_close( );

```

17. Зачем используется `die`?
- Иначе произойдёт ошибка
 - Чтобы узнать какая ошибка произошла
 - Чтобы остановить выполнение программы
18. Выберите правильный запрос, на отправку данных из приведённой ниже формы

```

<form action="1.php" method="post">
<p> введите текст </br></p>
<input type="text" name="name"/>
<label>имя</label>
<input type="text" name="number"/>
<label>номер</label>
<input type="reset" value="очистить"/>
<input type="submit" value="добавить">
</form>

```

- `$q="INSERT INTO test (name, number) VALUES ('$_POST['name'],'$_POST['number']');";`
 - `$q="INSERT INTO test (name, number) VALUES ('name','number');";`
 - `$q="INSERT INTO test (name, number) VALUES ('$name','$number');";`
19. Зачем в строке запроса две точки с запятой?
20. Какой запрос нужен для получения номера Даши?

Phone

Name	Number
Оля	111333
Даша	444555
Ксюша	222666

- a) SELECT Name FROM Phone WHERE Name="Даша";
 - b) SELECT * FROM Phone;
 - c) SELECT Number FROM Phone WHERE Name="Даша";
21. Какой запрос удалит первую строку из таблицы Phone?
- a) DELETE FROM Phone WHERE Name ="Оля";
 - b) DELETE FROM Phone WHERE Number ="111333";
 - c) DELETE FROM Phone LIKE "Оля";
22. Выберите запрос, для того чтобы изменить номер Ксюши
- a) INSERT INTO Phone (Name, Number) VALUES ('Ксюша', '888999');
 - b) UPDATE Phone SET Number='888999' WHERE Name='Ксюша';
 - c) SELECT Number FROM Phone WHERE Name='Ксюша';
4. **Критерии оценивания** (по 100-балльной системе оценивания):

Весовые коэффициенты по мере сложности, накладываемые на вопросы:

Вопрос 17 – 0,1

Вопросы 4,6,13,15,16,20 – 0,7

Вопросы 1,2,3,5,7,8,9,10,11,12,14,18,19,21, 22, 23 – 0,3

Каждый вопрос оценивается по 100 шкале, результат ответа на вопрос умножается на указанный коэффициент, далее полученные таким образом баллы по каждому вопросу суммируются

Оценка «отлично»	Суммарно набрано 85-100 баллов (отлично)
Оценка «хорошо»	Суммарно набрано 70-84 баллов (хорошо)
Оценка «удовлетворительно»	Суммарно набрано 50-69 баллов
Оценка «неудовлетворительно»	Суммарно набрано 0-49 баллов (неудовлетворительно)

5. Рекомендуемый перечень вопросов для самостоятельной подготовки:

1. Синтаксис php
2. Html-формы
3. http-запросы
4. Регулярные выражения
5. Функции для соединения с базой и отправки запроса
6. SQL запросы

➤ **ОЦЕНОЧНОЕ СРЕДСТВО** Самостоятельная работа №3

1. **Цель** закрепление навыков создания запросов в базу данных и обработки ответов.

2. Проверяемые компетенции (код): ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9
ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6

3. Пример оценочного средства (примерные тестовые задания, типовой вариант контрольной работы и др.)

Создание страницы и базы данных библиотеки.

1. Создайте базу со столбцами: автор, название, жанр, описание, год издания, обложка.
2. Заполните её. Должен быть хотя бы 1 автор у которого несколько книг (3-4), хотя бы один жанр, в котором несколько книг.
3. Создайте главную страницу с тремя формами: информация о книге по её названию, список всех книг автора, список всех книг жанра. Используйте LIKE в запросе чтобы можно было найти по одному слову из названия.
4. В ответ на запрос информации о книге должна появляться страница с названием книги и автора в качестве заголовка, изображением обложки слева и описанием справа.
5. В ответ на запрос всех книг автора должна появляться страница с автором в качестве заголовка и нумерованным списком, состоящим из названий книг и жанров.
6. В ответ на запрос всех книг жанра должна появляться страница с жанром в качестве заголовка и нумерованным списком, состоящим из авторов и названий книг.

Примечание: для запросов книг автора и жанра должен использоваться один php-файл как для домашнего и рабочего телефона.

7. Создайте и подключите стилевой файл

4. Критерии оценивания (по 100-балльной системе оценивания):

Задания 1-3 – 30 баллов

Задание 4 – 20 баллов

Задания 5, 6 – 10 баллов

Задание 7 – 30 баллов

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарии к программному коду, формы и компоненты на формах соответствуют заданию по расположению и их функциональным возможностям, выполнено объяснение к работе. При этом выполнены все задания. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении формы и компонентов на формах присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарии к программному коду. При этом выполнены задания 1-6. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарии к программному коду. При этом выполнены задания 1-4, 1-5. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы в том числе о реализации той или иной функции.
Оценка	Работа не всегда защищена с первого раза, верно решена только

«неудовлетворительно»	часть заданий 1-4 или решено меньшее количество заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.
-----------------------	---

5. Рекомендуемый перечень вопросов для самостоятельной подготовки:

1. Функции соединения с базой
2. SQL запросы
3. Отправка запроса
4. Обработка запроса

➤ **ОЦЕНОЧНОЕ СРЕДСТВО** Лабораторная работа №11

1. Цель: применение знаний о работе веб-браузера и сохраняемых им данных. И проверка умения разработки логики проекта.

2. Проверяемые компетенции (код): ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6

3. Пример оценочного средства (*примерные тестовые задания, типовый вариант контрольной работы и др.*)

Сделайте авторизацию на сайте библиотеки.

1. Создайте базу пользователей библиотеки, содержащую поля «логин», «пароль», «ключ». Пароли должны шифроваться.
2. Авторизованным пользователям должен быть доступен заказ книги. У не авторизованных, эта кнопка (ссылка) отсутствует.

4. Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарий к программному коду, формы и компоненты на формах соответствуют заданию по расположению и их функциональным возможностям, выполнено объяснение к работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении формы и компонентов на формах присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы в том числе о реализации той или иной функции.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно решена только часть заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

5. Рекомендуемый перечень вопросов для самостоятельной подготовки:

1. Сессия
2. Куки

➤ **ОЦЕНОЧНОЕ СРЕДСТВО:** Сообщение №2

1. **Цель:** Формирование системы знаний об основных возможностях языка C# и базовых средствах, и структурах данных для создания программ.

2. **Проверяемые компетенции (код):** ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6

3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

Написать сообщения на тему «Передача значений переменных с одной страницы данных на другую средствами php», «Использование php для реализации авторизации и запоминания сессии пользователя», «Использование php для создания e-mail рассылок». Привести примеры.

4. **Критерии оценивания:**

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарии к программному коду, выполнено объяснение к работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарии к программному коду. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарии к программному коду. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно решена только часть заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

➤ **ОЦЕНОЧНОЕ СРЕДСТВО** Лабораторная работа №12

1. **Цель:** формирование знаний о назначении, возможностях и использовании языка JavaScript для создания веб-страниц.

2. **Проверяемые компетенции (код):** ОК 2, ОК 3, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

1. Создайте страничку магазина: изображение товара, краткое описание под ним и кнопка «купить». После нажатия на эту кнопку стиль товара должен измениться. Сделайте 5 товаров.
2. Сделайте ответ от форм из предыдущих заданий на той же странице.
 - a) адрес
 - b) имя
 - c) уравнение решение должно происходить без обращения к серверу (без php).
3. Сделайте текстовое поле и кнопку, при нажатии на которую появляется следующее поле (сколько угодно раз).
Сделайте ещё одну кнопку, при нажатии которой текст, введённый в каждое поле, появится на этой же странице в виде списка.

4. Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарий к программному коду, формы и компоненты на формах соответствуют заданию по расположению и их функциональным возможностям, выполнено объяснение к работе. При этом выполнены все задания (или за исключением одного: 10 или 11-ого) Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении формы и компонентов на формах присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. При этом выполнены задания 1-8 и одно из 9-11. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. При этом выполнены задания 1-8. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы в том числе о реализации той или иной функции.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно решена только часть заданий 1-8 или решено меньшее количество заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

5. Рекомендуемый перечень вопросов для самостоятельной подготовки:

1. Объектная модель документа
2. Синтаксис JavaScript
3. Технология AJAX

➤ **ОЦЕНОЧНОЕ СРЕДСТВО:** Самостоятельная работа №4

1. **Цель:** формирование умений по применению технологии AJAX

2. Проверяемые компетенции (код): ОК 2, ОК 3, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

3. Пример оценочного средства (примерные тестовые задания, типовой вариант контрольной работы и др.)

После ввода логина и пароля пользователь остаётся на той же странице, становятся доступны некоторые функции.

4. Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарий к программному коду, формы и компоненты на формах соответствуют заданию по расположению и их функциональным возможностям, выполнено объяснение к работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении формы и компонентов на формах присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы в том числе о реализации той или иной функции.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно решена только часть заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

5. Рекомендуемый перечень вопросов для самостоятельной подготовки:

1. Объектная модель документа
2. Синтаксис JavaScript
3. Технология AJAX

➤ **ОЦЕНОЧНОЕ СРЕДСТВО** Контрольная работа №3

1. Цель: проверка уровня усвоения основ JavaScript

2. Проверяемые компетенции (код): ОК 2, ОК 3, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5

3. Пример оценочного средства (примерные тестовые задания, типовой вариант контрольной работы и др.)

1. Найдите ошибки в коде

```
<body>
document.write("Hello World")
</script>
<body>
```

2. Найдите ошибки в коде

```
d=(4,5,6);
document.write(d)
```

3. Нужны ли в конце строки точки с запятой?

4. В чём заключается объектная модель документа?

5. Перечислите объекты документа?

6. В чём заключается объектная модель браузера?

7. Перечислите объекты браузера?

8. Как происходит обращение к методу?

9. Перечислите операции с объектом документа?

10. Перечислите операции с объектом окна?

11. Что такое событие?

12. Что такое обработчик события?

13. В чём заключается технология AJAX?

14. Как можно получить доступ к объекту?

15. Где отображаются ошибки?

16. Что из себя представляет событие window.onload?

17. Чем отличаются переменные a и b?

```
function test(){
a = 123
var b = 456
```

18. Напишите комментарии к каждой строке

```
<script>
function f3(){
var x=document.getElementsByTagName("p")
res = document.getElementById("result")
res.innerHTML = 'id первого абзаца '+x[0].getAttribute("id")
res = document.getElementById("result2")
res.innerHTML = 'id второго абзаца '+x[1].getAttribute("id")
}
window.onload=f3
</script>
```

4. Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарий к программному коду, формы и компоненты на формах соответствуют заданию по расположению и их функциональным возможностям, выполнено объяснение к работе.
------------------	--

	При этом выполнены все задания (или за исключением одного 18-ого) Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении формы и компонентов на формах присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы в том числе о реализации той или иной функции.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно решена только часть заданий 1-8 или решено меньшее количество заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

5. Рекомендуемый перечень вопросов для самостоятельной подготовки:

1. Объектная модель документа
2. Объектная модель браузера
3. Объект document, его свойства и методы
4. Объект XMLHttpRequest, его свойства и методы
5. События

➤ ОЦЕНОЧНОЕ СРЕДСТВО: Сообщение №3

1. **Цель:** Формирование системы знаний об основных возможностях языка JavaScript.

2. **Проверяемые компетенции (код):** ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6

3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

Подготовка сообщений по теме библиотеки JavaScript, их функциональные возможности, назначение и примеры использования

4. **Критерии оценивания:**

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарий к программному коду, выполнено объяснение к работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.

Оценка «неудовлетворительно »	Работа не всегда защищена с первого раза, верно решена только часть заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.
-------------------------------------	---

➤ **ОЦЕНОЧНОЕ СРЕДСТВО:** Курсовая работа

1. **Цель:** формирование знаний и умений работать с литературой, обобщать и анализировать фактический материал, используя теоретические знания и практические навыки, полученные при освоении профессиональной образовательной программы.

2. **Проверяемые компетенции** (код): ОК 1 ОК 2 ОК 3 ОК 4 ОК 5 ОК 6 ОК 7 ОК 8 ОК 9 ПК 1.1 ПК 1.2 ПК 1.3 ПК 1.4 ПК 1.5 ПК 1.6

3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

1. Создание программы для общения по локальной сети с выделенным сервером.
2. Разработка электронного словаря с возможностью подключения к базе данных.
3. Составить программу для обучения переводу чисел из двоичной системы счисления в восьмеричную и шестнадцатеричную и обратно. Программа должна предлагать двоичное число, выбранное с помощью датчика случайных чисел, обучающийся - назвать число в восьмеричной и шестнадцатеричной системе счисления, система контролирует правильность ответов.
4. Написать программу построения биоритмов по данному дню рождения на заданный интервал времени
5. Разработка графического редактора (возможности: рисование линий, окружностей, квадратов)
6. Программирование системы управления файлами
7. Разработка программы-переводчика
8. Разработка программы-тренажера работы на клавиатуре. (Через определенный интервал времени (постепенно уменьшающийся) программа выводит случайный символ на экран. Пользователь должен нажимать соответствующие клавиши клавиатуры, стараясь успеть за выводом символов на экран.)
9. Текстовый редактор. Составить программное средство, реализующее стандартные функции обработки текстовых файлов. Реализовать возможность открытия, сохранения и создания новых файлов. Предоставить возможность пользователю выполнять форматирование текста и абзаца, осуществлять поиск и замену символов в тексте.
10. Разработка приложения "Помощник экзаменатора". Экзаменационные вопросы и ответы к ним хранятся в файлах на диске. Каждый вопрос имеет балл сложности. Необходимо подобрать пять вопросов из разных разделов курса, имеющих в сумме балл сложности N, и вывести их на экран. Предусмотреть тренировочный режим, работы, когда возможен вывод ответов на представленные вопросы. Доступ к тренировочному режиму работы предоставляется по паролю.
11. Разработка программы для сдачи теоретической части экзамена в ГИБДД
12. Разработка программы для регистрации нового пользователя и хранения его персональных данных в отдельном файле с возможностью выбора введения пароля вручную или его генерации, а также с возможностью изменения пароля и персональных данных пользователя

4.Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости приведены комментарии к программному коду, сделаны выводы по работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости приведены комментарии к программному коду. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости приведены комментарии к программному коду. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, в программе присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

II ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ

1. Форма проведения промежуточной аттестации: экзамен в 2 семестре.
2. Процедура проведения промежуточной аттестации

Экзамен

Студенты, успешно освоившие все предусмотренные лабораторные работы, а также успешно выполнившие текущие контрольные и самостоятельные работы (в соответствии с технологической картой дисциплины) при общем количестве баллов более 50 имеют право получить экзамен автоматически по результатам рейтинга.

Студенты, по окончании семестра набравшие менее 50 баллов выполняют задания до получения минимальной проходной оценки 50 баллов

В случае несогласия с текущей оценкой промежуточная аттестация по дисциплине осуществляется в традиционной форме по билетам, с учетом текущего рейтинга студента. Билет предполагает наличие двух вопросов. Повышение оценки возможно на 1 балл.

3. Перечень вопросов, письменных заданий и других материалов для промежуточной аттестации с типовыми примерами выполнения заданий

Перечень вопросов:

1. Типы программного обеспечения. Примеры прикладных приложений.
2. Требования к прикладному программному обеспечению.
3. Классификация языков программирования.
4. Типы и структуры данных в C#, приведение типов.
5. Программирование разветвляющихся алгоритмов в C#
6. Оператор выбора в C#
7. Программирование итерационных циклических алгоритмов в C#
8. Программирование арифметических циклических процессов в C#
9. Функции в C#
10. Массивы, объявление массивов, инициализация массивов
11. Строки и символы. Функции для обработки строк.
12. Файлы в C#

13. Структуры, описание структур, доступ к элементам структур.
14. Классы, описание классов, объявление элементов класса, работа с ними.
15. Форма как контейнер элементов управления для разработки интерфейса приложения. Основные свойства формы.
16. Кнопки как основной элемент интерфейса. Методы, используемые для обработки событий кнопок
17. Компоненты, используемые для ввода/вывода текста на форме
18. Флажки и переключатели
19. Таблицы, размещаемые на форме в C#
20. Диалоговые окна, вызов и обработка нажатий кнопок диалогового окна
21. События клавиатуры и мыши.
22. Назначение css
23. Css. Синтаксис, в коде html
24. Css. Блочная модель документа
25. Css. Цвета и шрифты
26. Css. Положение элемента на странице
27. Css. Наследование
28. Css. Приоритеты
29. Css. Классы, идентификаторы, псевдоклассы
30. Синтаксис php
31. Php. Область видимости переменных
32. Php. http-запрос
33. Php. Метод get
34. Php. Метод post
35. Php. Обработка html-форм
36. Php. Регулярные выражения: определение и функции
37. Php. Регулярные выражения: условные обозначения
38. Php. Соединение с базой данных
39. Php. Отправка запроса
40. Php. Обработка ответов от базы
41. Объектная модель документа
42. Объектная модель браузера
43. Объект document, его свойства и методы
44. Объект XMLHttpRequest, его свойства и методы
45. События

4. Система оценивания отдельных заданий и зачетной / экзаменационной работы в целом

Критерии оценивания экзаменационной работы

Оценка «отлично»	Представлен развернутый ответ на теоретический вопрос, а также поэтапное решение практического задания с пояснениями. Студент ориентируется в излагаемом материале, отвечает на дополнительные вопросы, связанные демонстрирует глубокие теоретические знания, знание первоисточников.
Оценка «хорошо»	Представлен достаточно развернутый ответ на теоретический вопрос, а также поэтапное решение практического задания с пояснениями. В решении практического задания могут быть допущены вычислительные ошибки, не искажающие лежащего в основе решения алгоритма Студент уверенно отвечает на дополнительные вопросы, демонстрирует достаточно высокий уровень теоретических знаний, знание первоисточников.

Оценка «удовлетворительно»	Представлен не полный ответ на теоретический вопрос, В решении практического задания могут быть допущены вычислительные ошибки, не искажающие лежащего в основе решения алгоритма. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Теоретический вопрос не раскрыт, в решении практического задания допущены существенные ошибки, ввиду незнания алгоритмов решения. Либо дан ответ только на один из вопросов билета. Студент затрудняется отвечать на дополнительные вопросы, в том числе непосредственно относящиеся к сути теоретического и практического вопросов билета.

Поддержка и тестирование программных модулей

ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ

Перечень формируемых компетенций:

- ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам
- ОК 02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.
- ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие.
- ОК 04. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами
- ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.
- ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, применять стандарты антикоррупционного поведения
- ОК 07.: Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях
- ОК 08. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности
- ОК 09. Использовать информационные технологии в профессиональной деятельности.
- ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языке
- ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.
- ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием
- ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств.
- ПК 1.4. Выполнять тестирование программных модулей.
- ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода.
- ПК 1.6. Разрабатывать модули программного обеспечения для мобильных платформ.

№ п/п	Контролируемые разделы (темы)	Код контролируемой компетенции (или её части)	Наименование оценочного средства
Раздел 1. Отладка и тестирование программного обеспечения			
1.	Тестирование	ОК 01-ОК 10, ПК 1.1.- ПК 1.6	Тест (итоговый)
2.	Тестирование «белым ящиком»	ОК 01-ОК 10, ПК 1.1.- ПК 1.6	Лабораторная работа №1 Лабораторная работа №2 Тест (итоговый)
3.	Тестирование «черным ящиком»	ОК 01-ОК 10, ПК 1.1.- ПК 1.6	Лабораторная работа №3 Тест (итоговый)

4.	Модульное тестирование	ОК 01-ОК 10, ПК 1.1.- ПК 1.6	Лабораторная работа №4 Тест (итоговый)
5.	Интеграционное тестирование	ОК 01-ОК 10, ПК 1.1.- ПК 1.6	Лабораторная работа №5 Тест (итоговый)
Раздел 2. Документирование			
6.	Оформление документации на программные средства с использованием инструментальных средств	ОК 01-ОК 10, ПК 1.1.- ПК 1.6	Лабораторная работа №6 Тест (итоговый)
7.	Промежуточная аттестация	ОК 01-ОК 10, ПК 1.1.- ПК 1.6	Дифференцированный зачет

Сформированность выше перечисленных компетенций предполагает, что в результате освоения дисциплины (профессионального модуля) обучающийся должен:

иметь практический опыт:

- в разработке кода программного продукта на основе готовой спецификации на уровне модуля;
- использовании инструментальных средств на этапе отладки программного продукта;
- проведении тестирования программного модуля по определенному сценарию;
- использовании инструментальных средств на этапе отладки программного продукта;
- разработке мобильных приложений

уметь:

- осуществлять разработку кода программного модуля на языках низкого и высокого уровней;
- создавать программу по разработанному алгоритму как отдельный модуль;
- выполнять отладку и тестирование программы на уровне модуля;
- осуществлять разработку кода программного модуля на современных языках программирования;
- уметь выполнять оптимизацию и рефакторинг программного кода;
- оформлять документацию на программные средства.

знать:

- основные этапы разработки программного обеспечения;
- основные принципы технологии структурного и объектно-ориентированного программирования;
- способы оптимизации и приемы рефакторинга;
- основные принципы отладки и тестирования программных продуктов.

I ТЕКУЩИЙ КОНТРОЛЬ ПО МЕЖДИСЦИПЛИНАРНОМУ КУРСУ

➤ **ОЦЕНОЧНОЕ СРЕДСТВО:** Лабораторная работа № 1. Тестирование программ

1. **Цель:** учиться составлять программы и проводить их тестирование
2. **Проверяемые компетенции** (код): ОК 01-ОК 10, ПК 1.1.- ПК 1.6
3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)
 - Составьте алгоритм и программу определения вида треугольника, задаваемого на плоскости координатами вершин, с выделением вспомогательных алгоритмов. Докажите его правильность.
 - Составьте алгоритм подсчета суммы элементов массива кратных шести. Докажите его правильность.
4. **Критерии оценивания:**

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, сделаны выводы по работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

➤ **ОЦЕНОЧНОЕ СРЕДСТВО:** Лабораторная работа № 2. Тестирование «белым ящиком»

1. **Цель:** формировать умения составлять систему тестов.
2. **Проверяемые компетенции** (код): ОК 01-ОК 10, ПК 1.1.- ПК 1.6
3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

Составьте системы тестов для решения следующих задач:

 - Известен расход по N видам горючего в каждом из M автохозяйств. Определите для каждого хозяйства вид горючего с наибольшим и с наименьшим расходом.
 - Определите пройдет ли кирпич с ребрами a,b,c в прямоугольное отверстие со сторонами X и Y. Просовывать кирпич в отверстие разрешается только так, чтобы каждое из его ребер было параллельно или перпендикулярно каждой из сторон отверстия.

Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан теоретический конспект, сделаны выводы по работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
------------------	--

Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан теоретический конспект. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан теоретический конспект. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

➤ **ОЦЕНОЧНОЕ СРЕДСТВО:** Лабораторная работа № 3.

Тестирование «черным ящиком»

- 1. Цель:** формировать умения составлять систему тестов, методом «черный ящик».
- 2. Проверяемые компетенции (код):** ОК 01-ОК 10, ПК 1.1.- ПК 1.6
- 3. Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

Составьте системы тестов для решения следующей задачи:

Формулировка задачи: Вычислите периметр и площадь прямоугольного треугольника по двум катетам.

Тесты участника по данной теме. При $a=2$, $b=3$. $P=8,60555127555$ $S=3$

Исходный текст программы: }

Program z1;

Var a,b,c,p,s: real;

Begin Repeat

 Writeln('Введите длину первого катета (a>0)');

 Read (a);

 Until a>0;

 Repeat

 Writeln('Введите длину второго катета (b>0)');

 Read (b);

 Until b>0;

 s:=a*b/2;

 c:= sqrt(sqr(a)+sqr(b));

 p:= a+b+c;

 Writeln('S=', s);

 Writeln('P=', p);

end.

4. Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан теоретический конспект, сделаны выводы по работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан теоретический конспект. Студент уверенно отвечает на

	вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан теоретический конспект. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

➤ **ОЦЕНОЧНОЕ СРЕДСТВО: Лабораторная работа № 4. Модульное тестирование**

- 1. Цель:** формировать умение разбивать программу на модули и проводить отладку отдельных модулей.
- 2. Проверяемые компетенции** (код) ОК 01-ОК 10, ПК 1.1.- ПК 1.6
- 3. Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

Составить программу вычисления среднего арифметического элементов массива, нахождения возрастающей последовательности из массива.

4. Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан теоретический конспект, сделаны выводы по работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан теоретический конспект. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан теоретический конспект. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

➤ **ОЦЕНОЧНОЕ СРЕДСТВО: Лабораторная работа №5. Интеграционное тестирование**

- 1. Цель:** формировать умение разбивать программу на модули и проводить отладку отдельных модулей.
- 2. Проверяемые компетенции** (код): ОК 01-ОК 10, ПК 1.1.- ПК 1.6
- 3. Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

Составьте системы тестов для решения следующей задачи методами Большой взрыв, Снизу вверх, Сверху вниз, Смешанный / сэндвич:

Определите пройдет ли кирпич с ребрами a,b,c в прямоугольное отверстие со сторонами X и Y. Просовывать кирпич в отверстие разрешается только так,

чтобы каждое из его ребер было параллельно или перпендикулярно каждой из сторон отверстия.

4. Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан теоретический конспект, сделаны выводы по работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан теоретический конспект. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан теоретический конспект. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

➤ **ОЦЕНОЧНОЕ СРЕДСТВО: Лабораторная работа №6. Оформление документации на программные средства с использованием инструментальных средств**

- 1. Цель:** Формирование умения составлять документацию.
- 2. Проверяемые компетенции (код):** ОК 01-ОК 10, ПК 1.1.- ПК 1.6
- 3. Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)
Составить и описать систему тестов используя программное обеспечение в соответствии с Единой системой программной документации.

4. Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан теоретический конспект, сделаны выводы по работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан теоретический конспект. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан теоретический конспект. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

II ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ ПО МЕЖДИСЦИПЛИНАРНОМУ КУРСУ

4. Форма проведения промежуточной аттестации: дифференцированный зачет.

5. Процедура проведения промежуточной аттестации

Дифференцированный зачет проводится в традиционной форме. К промежуточной аттестации допускаются студенты, освоившие на положительную оценку все обязательные виды запланированных учебных заданий

Примечание: Студенты, выполнившие на положительную оценку все обязательные виды запланированных учебных заданий, могут автоматически получить оценку в соответствии с набранными баллами.

Повышение оценки на зачете возможно только на 1 балл.

6. Пример оценочного средства (примерные тестовые задания, типовой вариант контрольной работы и др.)

1. Тестирование как часть процесса верификации программного обеспечения
2. Виды ошибок. Методы отладки
3. Методы тестирования
4. Классификация тестирования по уровням
5. Тестирование производительности
6. Регрессионное тестирование
7. Тестирование «белым ящиком»
8. Тестирование «черным ящиком»
9. Модульное тестирование
10. Интеграционное тестирование
11. Оформление документации на программные средства с использованием инструментальных средств

5. Система оценивания отдельных заданий и зачетной / экзаменационной работы в целом

Оценка «отлично»	Представлен развернутый ответ на теоретический вопрос, а также поэтапное решение практического задания с пояснениями. Студент ориентируется в излагаемом материале, отвечает на дополнительные вопросы, связанные демонстрирует глубокие теоретические знания, знание первоисточников.
Оценка «хорошо»	Представлен достаточно развернутый ответ на теоретический вопрос, а также поэтапное решение практического задания с пояснениями. В решении практического задания могут быть допущены вычислительные ошибки, не искажающие лежащего в основе решения алгоритма. Студент уверенно отвечает на дополнительные вопросы, демонстрирует достаточно высокий уровень теоретических знаний, знание первоисточников.
Оценка «удовлетворительно»	Представлен не полный ответ на теоретический вопрос, В решении практического задания могут быть допущены вычислительные ошибки, не искажающие лежащего в основе решения алгоритма. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Теоретический вопрос не раскрыт, в решении практического задания допущены существенные ошибки, ввиду незнания алгоритмов решения. Либо дан ответ только на один из вопросов билета. Студент затрудняется отвечать на дополнительные вопросы, в том числе непосредственно относящиеся к сути теоретического и практического вопросов билета.

**Разработка мобильных приложений
ПАСПОРТ ОЦЕНОЧНЫХ СРЕДСТВ**

Перечень формируемых компетенций:

- ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам
- ОК 02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.
- ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие
- ОК 04. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами
- ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.
- ОК 06.: Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, применять стандарты антикоррупционного поведения.
- ОК 07.: Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях.
- ОК 08.: Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности
- ОК 09. Использовать информационные технологии в профессиональной деятельности.
- ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языке
- ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.
- ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием
- ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств.
- ПК 1.4. Выполнять тестирование программных модулей.
- ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода.
- ПК 1.6: Разрабатывать модули программного обеспечения для мобильных платформ.

№ п/п	Контролируемые разделы (темы)	Код контролируемой компетенции (или её части)	Наименование оценочного средства
1	Раздел 1. Основные платформы и языки разработки мобильных приложений Раздел 2. Создание и тестирование модулей для мобильных приложений	ОК 01- ОК 10, ПК 1.1. – ПК 1.6	Лабораторная работа №1
		ОК 01- ОК 10, ПК 1.1. – ПК 1.6	Лабораторная работа №2
		ОК 01- ОК 10, ПК 1.1. – ПК 1.6	Лабораторная работа №3
		ОК 01- ОК 10, ПК 1.1. – ПК 1.6	Лабораторная работа №4

	Самостоятельная работа		
	Промежуточная аттестация	ОК 01- ОК 10, ПК 1.1. – ПК 1.6	Дифференцированный зачет

Сформированность выше перечисленных компетенций предполагает, что в результате освоения дисциплины (профессионального модуля) обучающийся должен:

3.1.	Знать:
3.1.1.	основные этапы разработки программного обеспечения; основные принципы технологии структурного и объектно-ориентированного программирования; способы оптимизации и приемы рефакторинга; основные принципы отладки и тестирования программных продуктов
3.2.	Уметь:
3.2.1.	осуществлять разработку кода программного модуля на языках низкого и высокого уровней; создавать программу по разработанному алгоритму как отдельный модуль; выполнять отладку и тестирование программы на уровне модуля; осуществлять разработку кода программного модуля на современных языках программирования; уметь выполнять оптимизацию и рефакторинг программного кода; оформлять документацию на программные средства
3.3.	Иметь навыки и (или) опыт деятельности (владеть):
3.3.1.	разработке кода программного продукта на основе готовой спецификации на уровне модуля; использовании инструментальных средств на этапе отладки программного продукта; проведении тестирования программного модуля по определенному сценарию; использовании инструментальных средств на этапе отладки программного продукта; разработке мобильных приложений

— I ТЕКУЩИЙ КОНТРОЛЬ ПО МЕЖДИСЦИПЛИНАРНОМУ КУРСУ

Лабораторные представляют собой Codelabs - пошаговые уроки по реализации какого-то функционала в android от google.

Для их сдачи и защиты необходимо создать аккаунт на [github](#)

Отчет по лабораторным включает в себя:

1. Титульный лист (ваше имя, номер лабораторной и все остальное по стандарту).
2. Описание работы (что было проделано).
3. Результат (Чему вы научились).
4. Ссылка на репозиторий с исходным кодом на вашем GitHub аккаунте.

➤ ОЦЕНОЧНОЕ СРЕДСТВО: Лабораторная работа №1

1. **Цель:** получения навыков создания тестовых сценариев на основе методики управления требованиями.
2. **Проверяемые компетенции** (код) ОК 1, ОК 2, ОК 4, ОК 9, ОК 10, ПК 1.2, ПК 1.6.
3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

Эта лаборатория кода является частью курса "Основы Android Kotlin". Вы получите максимальную отдачу от этого курса, если поработаете с кодовыми таблицами последовательно. Все кодовые таблицы курсов перечислены на [целевой странице кодовых таблиц Android Kotlin Fundamentals](#) .

Введение

Пока вы все настроили, и Android Studio создала для вас много кода. Прежде чем изменять весь этот код, важно знать, что вы только что создали, и как перемещаться по исходным файлам приложения для Android.

В этой кодовой лаборатории вы узнаете больше об основных компонентах приложения для Android и добавите простую интерактивность в приложение с помощью кнопки.

Что вы уже должны знать

- Как установить и открыть Android Studio.
- Как создать новый проект приложения.
- Как запустить приложение на эмуляторе или физическом устройстве.

Что ты узнаешь

- Как отредактировать файл макета приложения.
- Как создать приложение с интерактивным поведением.
- Много новой терминологии. Ознакомьтесь с [Глоссарием словаря, где вы найдете](#) понятные объяснения терминов и понятий.

Что ты будешь делать

- Изучите `MainActivity` файл Kotlin и файл макета занятия.
- Отредактируйте макет действия в XML.
- Добавьте `Button` элемент в макет занятия.
- Извлеките жестко запрограммированные строки в файл строковых ресурсов.

- Реализуйте методы обработчика кликов для отображения сообщений на экране, когда пользователь касается `Button`.

Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, сделаны выводы по работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно составлена только часть документа, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

➤ ОЦЕНОЧНОЕ СРЕДСТВО: Лабораторная работа №2

1. **Цель:** Формирование системы знаний об основных принципах создания технического задания, формирование умений разрабатывать программное обеспечение по конкретному техническому заданию

2. **Проверяемые компетенции** (код): ОК 1, ОК 2, ОК 4, ОК 9, ОК 10, ПК 1.2, ПК 1.6.

3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

Задание: Лабораторная работа №2 Архитектура программного продукта

1) Описать архитектуру программного продукта, визуализировать в любом формате. Сохранить либо в PDF, либо в JPEG.

2) Определить стек технологий, используемый для реализации проекта (ЯП, фреймворки библиотеки, ОС, серверное ПО, среда исполнения, языки разметки, облачные решения). Обосновать выбор стека. Объем - не менее 4000 знаков с пробелами.

Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, сделаны выводы по работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты. Студент уверенно

	отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно составлена только часть документа, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

➤ **ОЦЕНОЧНОЕ СРЕДСТВО: Лабораторная работа №3**

1. **Цель:** Формирование системы знаний об основных принципах создания технического задания, формирование умений разрабатывать программное обеспечение по конкретному техническому заданию

2. **Проверяемые компетенции** (код ОК 1, ОК 2, ОК 4, ОК 9, ОК 10, ПК 1.2, ПК 1.6.

3. **Пример оценочного средства** (примерные тестовые задания, типовой вариант контрольной работы и др.)

Спроектировать:

Пользовательский интерфейс (при наличии)

В любом графическом редакторе или приложении для прототипирования интерфейсов (Figma, Sketch, Lunacy). Прикрепить файл макета или прототипа, или ссылку на проект, если он выполнен в онлайн сервисе.

Сервер базы данных (при наличии)

Если проект предполагает создание одной базы данных, прикрепить файл базы данных. Сервер БД содержит несколько баз – прикрепить визуализацию в любом формате.

Структуру приложения (дизайн приложения)

Прикрепить структуру классов приложения в любом формате, с указанием базовых и производных классов, интерфейсов, пространств имен, модулей, а также в классах указать публичные методы и свойства (в C#) При использовании языков разметки и таблиц стилей – описать иерархию компонентов и элементов UI

Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, сделаны выводы по работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты. Студент демонстрирует достаточный уровень теоретических знаний, однако

	затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно составлена только часть документа, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

➤ **ОЦЕНОЧНОЕ СРЕДСТВО: Лабораторная работа №4**

1. Цель: Формирование системы знаний и умений по работе в составе коллектива разработчиков.

2. Проверяемые компетенции (код): ОК 1, ОК 2, ОК 4, ОК 9, ОК 10, ПК 1.2, ПК 1.6.

3. Пример оценочного средства (примерные тестовые задания, типовой вариант контрольной работы и др.)

Опираясь на предыдущую лабораторную работу выполнить анализ и обработать исключительные ситуации

Критерии оценивания:

Оценка «отлично»	Работа выполнена в срок, защищена с первого раза, полностью оформлена, получены достоверные результаты, при необходимости написан комментарий к программному коду, сделаны выводы по работе. Студент блестяще отвечает на вопросы, демонстрирует глубокие теоретические знания.
Оценка «хорошо»	Работа выполнена в срок, защищена с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент уверенно отвечает на вопросы, демонстрирует достаточно высокий уровень теоретических знаний.
Оценка «удовлетворительно»	Работа выполнена в срок, возможно, защищена не с первого раза, в оформлении присутствуют незначительные недочеты, получены достоверные результаты, при необходимости написан комментарий к программному коду. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Работа не всегда защищена с первого раза, верно решена только часть заданий, в оформлении присутствуют существенные недочеты. Студент затрудняется отвечать на вопросы.

II ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ

1. Форма проведения промежуточной аттестации: дифференцированный зачет.
2. Процедура проведения промежуточной аттестации

К промежуточной аттестации допускаются студенты, освоившие на положительную оценку все обязательные виды запланированных учебных заданий. Дифференцированный зачет проводится по написанию практического занятия. В билете два вопроса.

Примечание: Студенты, выполнившие на положительную оценку все обязательные виды запланированных учебных заданий, могут автоматически получить оценку в соответствии с набранными баллами.

3. Перечень заданий для подготовки к дифференцированному зачету

Задание

Разработать подсистему для регистрации пользователя: на начальном экране пользователю видны две ссылки «Авторизация» и «Регистрация».

При выборе ссылки «Регистрация» пользователь переходит на вкладку, внешний вид которой представлен на рисунке 1.

Заполните все поля и нажмите кнопку "Подтвердить"

Введите логин

Введите пароль

Подтвердите пароль

Введите фамилию

Введите имя

Введите ваш год рождения

Введите email

Выберите пол

Мужской Женский

Подтвердить

Рисунок 1. Форма регистрации нового пользователя

В ходе реализации проекта предусмотреть следующие ограничения:

1. Все поля обязательны для заполнения.
2. Логин задается текстом и должен иметь длину не менее 8 символов
3. Пароль задается текстом и цифрами и должен иметь длину не менее 8 символов. Пароли при вводе и подтверждении совпадают.
4. Фамилия и имя задаются текстом и должны иметь длину не менее 2 символов
5. Год рождения задается четырехзначным числом в диапазоне [1950; 2010]
6. Email должен быть длиной не менее 12 символов и иметь знак @ не в первой и не в последней позиции

7. Если данные введены верно, пользователь получает соответствующее сообщение. В случае ошибок обязательно информирование пользователя об ошибках и о том, как должно быть заполнено соответствующее поле.

4-Критерии оценивания дифференцированного зачета

Оценка «отлично»	Представлен развернутый ответ на теоретический вопрос, а также поэтапное решение практического задания с пояснениями. Студент ориентируется в излагаемом материале, отвечает на дополнительные вопросы, связанные демонстрирует глубокие теоретические знания, знание первоисточников.
Оценка «хорошо»	Представлен достаточно развернутый ответ на теоретический вопрос, а также поэтапное решение практического задания с пояснениями. В решении практического задания могут быть допущены вычислительные ошибки, не искажающие лежащего в основе решения алгоритма. Студент уверенно отвечает на дополнительные вопросы, демонстрирует достаточно высокий уровень теоретических знаний, знание первоисточников.
Оценка «удовлетворительно»	Представлен не полный ответ на теоретический вопрос, В решении практического задания могут быть допущены вычислительные ошибки, не искажающие лежащего в основе решения алгоритма. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Теоретический вопрос не раскрыт, в решении практического задания допущены существенные ошибки, ввиду незнания алгоритмов решения. Либо дан ответ только на один из вопросов билета. Студент затрудняется отвечать на дополнительные вопросы, в том числе непосредственно относящиеся к сути теоретического и практического вопросов билета.

Системное программирование

ПАСПОРТ ОЦЕНОЧНЫХ СРЕДСТВ

- ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам
- ОК 02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.
- ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие.
- ОК 04. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами
- ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.
- ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, применять стандарты антикоррупционного поведения
- ОК 07.: Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях
- ОК 08. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности
- ОК 09. Использовать информационные технологии в профессиональной деятельности.
- ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языке
- ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.
- ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием
- ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств.
- ПК 1.4. Выполнять тестирование программных модулей.
- ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода.
- ПК 1.6. Разрабатывать модули программного обеспечения для мобильных платформ.

	Контролируемые этапы	Код контролируемой компетенции	Наименование оценочного средства
	Раздел 1. Дисциплина системное программирование. Программы и программное обеспечение. Системное программирование. Этапы подготовки программы.	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	Устный опрос Тестирование
	Самостоятельная работа	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	Системное программирование Этапы подготовки программ.
	Раздел 2. Предложения языка Ассемблера. Регистры. Биты и байты ASCII . .Двоичные числа и шестнадцатеричное представление. Сегменты. Расширение набора команд. Способы адресации.	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	Письменная работа Тестирование

	Директивы. Директивы определения данных. Директива определения байта.(DB). Директива определения слова (DW). Директива определения двойного слова (DD).		
	Самостоятельная работа	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	Директива определения учетверенного слова (DQ). Директива определения десяти байт (DT) Непосредственные операнды. Директива EQU.
	Раздел 3. Сегментные регистры: CS, DS, SS и ES. Регистры общего назначения: AX, BX, CX и DX. Регистровые указатели: SP и BP. Индексные регистры: SI и DI. Регистр командного указателя: IP.. Флаговый регистр.	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	Устный опрос Тестирование
	Самостоятельная работа	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	Регистр командного указателя: IP Флаговый регистр
	Раздел 4. Арифметические операции... Обработка двоичных данных. Беззнаковые и знаковые данные. Умножение. Сдвиг регистровой пары DX:AX. Деление. Преобразование знака.	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	Письменная работа
	Самостоятельная работа	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	Обработка данных в форматах ASCII и BCD. Двоично-десятичный формат (BCD). Сдвиг и округление.
	Раздел 5. Команды обработки строк. Свойства операций над строками. Операции над строками. REP: Префикс повторения цепочечной команды .MOVS: Пересылка строк LODS: Загрузка строки STOS:	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	Проверочная работа Письменное тестирование
	Самостоятельная работа	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	Запись строки CMPS: Сравнение строк SCAS: Сканирование строк
	Раздел 6. Свойства операторов работы с экраном. Команда прерывания INT. Установка курсора. Очистка экрана. Использование символов возврата каретки, конца строки и табуляции для вывода на экран .Расширенные возможности экранных операций.	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	

	Самостоятельная работа	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	Расширенный ASCII код Другие операции ввода/вывода. Ввод с клавиатуры по команде BIOS INT 16H. Функциональные клавиши Цвет и графика
	Раздел 7. Ввод программы и подготовка программы для выполнения Ассемблирование программы. Компоновка программы. Выполнение программы.	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	
	Самостоятельная работа	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	Файл перекрестных ссылок
	Раздел 8. Команда JMP Команда LOOP Флаговый регистр Команды условного перехода Процедуры и оператор CALL. Сегмент стека Команды логических операций: AND, OR, XOR, TEST, NOT Изменение строчных букв на заглавные.	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	
	Самостоятельная работа	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	Команды сдвига и циклического сдвига Организация программ
	Раздел 9. Атрибуты EXTRN и PUBLIC. Компоновка программ на языке. C ++ и Ассемблере. Выполнение COM программы. Выполнение EXE программы .	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	
	Самостоятельная работа	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	Функции загрузки и выполнения программы
0	Раздел 10. Выполнение программ. Начинаем работать. Определение данных Машинная адресация . Определение размера памяти.	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	
	Самостоятельная работа	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	Специальные средства отладчика .
1	Раздел 11. Макропроцессоры. Основные понятия. Сравнение макросредств и подпрограмм. Некоторые возможности макроязыка. Локальные переменные макроопределения. Присваивание значений переменным макроопределения.	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	

	Самостоятельная работа	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	Глобальные переменные макроопределения метки Уникальные метки
2	Раздел 12. Кросс системы. Вычислительные системы. Модель регистров. Модель оперативной памяти.	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	
	Самостоятельная работа	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	Модель процессора
3	Раздел 13. Классификация ошибок программирования Ошибки при задании необходимых начальных условий для отдельных программ	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	
	Самостоятельная работа	ОК 01 - ОК 10, ПК 1.1. – ПК 1.6.	Распознавание ошибок Ассемблером Распространенные ошибки в драйверах ввода /вывода Распространенные ошибки в программах прерывания

Знать:

- основные этапы разработки программного обеспечения;
- основные принципы технологии структурного и объектно-ориентированного программирования;
- основные принципы отладки и тестирования программных продуктов;
- методы и средства разработки технической документации.

Уметь:

- осуществлять разработку кода программного модуля на современных языках программирования;
- создавать программу по разработанному алгоритму как отдельный модуль;
- выполнять отладку и тестирование программы на уровне модуля;
- оформлять документацию на программные средства;
- использовать инструментальные средства для автоматизации оформления документации.

Иметь практический опыт:

- разработки алгоритма поставленной задачи и реализации его средствами автоматизированного проектирования;
- разработки кода программного продукта на основе готовой спецификации на уровне модуля;
- использования инструментальных средств на этапе отладки программного продукта;
- проведения тестирования программного модуля по определенному сценарию;

1 ТЕКУЩИЙ КОНТРОЛЬ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ (МОДУЛЮ)

➤ ОЦЕНОЧНОЕ СРЕДСТВО

➤ ПР 1 Контрольный тест «Системное программирование»

1. **Цель** – проверить навыки по разработке кода программного модуля на языках программирования низкого уровня;



- **Уметь:** – осуществлять разработку кода программного модуля на современных языках программирования;
– создавать программу по разработанному алгоритму как отдельный модуль;

2. **Проверяемые компетенции (код):** ОК 2 -3. ПК 1.1- 1.2.

- **3. Пример оценочного средства** (*примерные тестовые задания, типовой вариант контрольной работы и др.*)

Фонд тестовых заданий по дисциплине «Системное программирование»

Тестирование проводится по двум разделам курса на практических занятиях. Студент получает контрольный лист с вариантом теста, составленного по вопросам, которые приведены ниже. В каждом варианте 25 вопросов. Время на подготовку ответов 25 минут. Каждый студент заполняет контрольный лист теста, который возвращает вместе с заданием. Вся группа тестируется одновременно.

Тестирование «Системное программирование»

Вопрос 1 Системная программа – это

- a. программа, предназначенная для поддержания работоспособности СОИ или повышения эффективности ее использования.
- b. программа, предназначенная для решения задачи или класса задач в определенной области применения СОИ
- c. системная программа, реализующая набор функций управления, который включает в себя управление ресурсами и взаимодействие с внешней средой СОИ, восстановление работы системы после проявления неисправностей в технических средствах.

Вопрос 2 Исходный модуль – это

- a. программа или функционально завершенный фрагмент программы, предназначенный для хранения, трансляции, объединения с другими программными модулями и загрузки в оперативную память.
- b. программный модуль на исходном языке, обрабатываемый транслятором и представляемый для него как целое, достаточное для проведения трансляции.
- c. программный модуль, получаемый в результате трансляции исходного модуля.

Вопрос 3 Автокод – это

- a. символьный язык программирования, предложения которого по своей структуре в основном подобны командам и обрабатываемым данным конкретного машинного языка.
- b. язык программирования, понятия и структура которого удобны для восприятия человеком.
- c. язык программирования, предназначенный для представления программы в форме, позволяющей выполнять ее непосредственно техническими средствами обработки информации.

Вопрос 4... - реализация смысла некоторого синтаксически законченного текста, представленного на конкретном языке.

- a. Интерпретация
- b. Трансляция
- c. Компиляция

Вопрос 5 Программное обеспечение – это

- a. совокупность программ СОО и программных документов, необходимых для их эксплуатации
- b. комплекс программ, которые обеспечивают управление компонентами компьютерной системы
- c. компьютерные программы, написанные для пользователей или самими пользователями, для задания компьютеру конкретной работы

Вопрос 6 Регистр - это

- a. Ячейка в оперативной памяти
- b. Ячейка памяти процессора
- c. Адресуемая ячейка памяти

Вопрос 7 Какие регистры относятся к сегментным:

- a. Es
- b. Dх
- c. Ip
- d. Cs
- e. Al

Вопрос 8 Команда mov –

- a. Команда помещения в стек
- b. Команда перемещения данных
- c. Команда пересылки байта

Вопрос 9 Какая команда относится к логическим командам:

- a. Sub
- b. Or
- c. Inc

Вопрос 10 Команда ... начинает выполнение с новой ветки в любом случаи

- a. Безусловного перехода
- b. Условного перехода
- c. Цикла

Вопрос 11 Объявляется непосредственно внутри ассемблера - ...

- a. Локальная метка
- b. Глобальная метка

с. Системная метка

Вопрос 12 Прерывание, которое возникает при изменении какого-либо устройства:

- a. Внутренние
- b. Аппаратное
- c. Программное

Вопрос 13

1. Какой командой вызывается процедура:

- a. Proc
- b. Call
- c. Ret

Вопрос 14 Директива – это

- a. Символическое имя для определения места в памяти, где размещены данные
- b. Блок команд, который может быть вызван из любого места основной программы
- c. Процесс приостановке программы

Вопрос 15 Какой командой вызывается прерывания:

- a. In
- b. Inc
- c. Int

Вопрос 16 Определить ошибки в программе:

- 1. Var X: integer;
- 2. Begin
- 3. X:=100;
- 4. asm
- 5. Mov bl, 25
- 6. Mov ax, X
- 7. Div ax,bl
- 8. Mov X,bl
- 9. end;
- 10. writeln ('X=',X);
- 11. end.

- a. Строка 5,7
- b. Строка 6,7
- c. Строка 7,8

Вопрос 17 Определить результат выполнения программы:

```
Begin  
asm  
Mov Ah,02h  
Mov Dl,'*'  
int 21h  
end;  
end.
```

Вопрос 18

Управляющая программа - это

- а. программа, предназначенная для поддержания работоспособности СООИ или повышения эффективности ее использования.
- б. программа, предназначенная для решения задачи или класса задач в определенной области применения СООИ
- с. системная программа, реализующая набор функций управления, который включает в себя управление ресурсами и взаимодействие с внешней средой СООИ, восстановление работы системы после проявления неисправностей в технических средствах.

Вопрос 19 Объектный модуль – это

- а. программа или функционально завершенный фрагмент программы, предназначенный для хранения, трансляции, объединения с другими программными модулями и загрузки в оперативную память.
- б. программный модуль на исходном языке, обрабатываемый транслятором и представляемый для него как целое, достаточное для проведения трансляции.
- с. программный модуль, получаемый в результате трансляции исходного модуля.

Вопрос 20 Язык высокого уровня – это

- а. символьный язык программирования, предложения которого по своей структуре в основном подобны командам и обрабатываемым данным конкретного машинного языка.
- б. язык программирования, понятия и структура которого удобны для восприятия человеком.
- с. язык программирования, предназначенный для представления программы в форме, позволяющей выполнять ее непосредственно техническими средствами обработки информации.

Вопрос 21 ... - преобразование программы, представленной на одном языке программирования, в программу на другом языке программирования, в определенном смысле равносильную первой.

- а. Интерпретация
- б. Трансляция
- с. Компиляция

Вопрос 22 Прикладное программное обеспечение – это

- а. совокупность программ СООИ и программных документов, необходимых для их эксплуатации
- б. комплекс программ, которые обеспечивают управление компонентами компьютерной системы
- с. компьютерные программы, написанные для пользователей или самими пользователями, для задания компьютеру конкретной работы

Вопрос 23 Язык Ассемблер - это

- а. система записи программы с детализацией до отдельной машинной команды
- б. язык программирования, понятия и структура которого удобны для восприятия человеком
- с. система, образуемая языком программирования, компилятором или интерпретатором программ

Вопрос 24 Какие регистры относятся к регистрам данных:

- а. Es
- б. Dx
- с. Ip
- д. Cs

е. А1

Вопрос 25 Команда add –

- а. Команда сложения
- б. Команда умножения
- с. Команда вычитания

4. Критерии оценивания:

Оценка «отлично»	ставится студенту, если студент из 25 вопросов ответил правильно от 23 до 25 вопросов.
Оценка «хорошо»	ставится студенту, если студент из 25 вопросов ответил правильно от 18 до 22 вопросов.
Оценка «удовлетворительно»	ставится, если студент из 25 вопросов ответил правильно от 13 до 17 вопросов.
Оценка «неудовлетворительно»	ставится, если студент из 25 вопросов ответил правильно на 12 вопросов и менее.

5. Рекомендуемый перечень вопросов для самостоятельной подготовки:

1. Системное программирование. Этапы подготовки программы .
2. Программирование на языке Ассемблера. Предложения языка Ассемблера .
3. Регистры. Биты и байты ASCII
4. Двоичные числа. Шестнадцатеричное представление.
5. Сегменты. Расширение набора команд.
6. Способы адресации. Директивы. Директивы определения данных.
7. Директива определения байта (DB). Директива определения слова (DW).
8. Директива определения двойного слова (DD). Директива определения учетверенного слова (DQ).
9. Директива определения десяти байт (DT). Непосредственные операнды Директива EQU.
10. Сегментные регистры: CS, DS, SS и ES.
11. Регистры общего назначения: AX, BX, CX и DX.
12. Регистровые указатели: SP и BP. Индексные регистры: SI и DI .
13. Регистр командного указателя IP. Флаговый регистр.
14. Обработка двоичных данных. Беззнаковые и знаковые данные.
15. Ассемблер. Умножение. Сдвиг регистровой пары DX:AX Деление.
16. Ассемблер. Преобразование знака Обработка данных в форматах ASCII и BCD.
17. Ассемблер. Двоично-десятичный формат (BCD)Сдвиг и округление.
18. Ассемблер. Команда прерывания INT. Установка курсора. Очистка экрана .
19. Ассемблер. Использование символов возврата каретки, конца строки и табуляции для вывода на экран
20. Ассемблер. Расширенные возможности экранных операций Расширенный ASCII код.
21. Ассемблер. Другие операции ввода/вывода Ввод с клавиатуры по команде BIOS INT 16H.
22. Ассемблер. Функциональные клавиши Цвет и графика.
23. Ассемблер. Межсегментные вызовы. Атрибуты EXTRN и PUBLIC.
24. Компоновка программ на языке C++ и Ассемблере.

ОЦЕНОЧНОЕ СРЕДСТВО

➤ ПР2 Контрольный тест «Системное программирование»

1. Цель – проверить навыки по разработке кода программного модуля на языках программирования низкого уровня;

➤ Уметь: – – создавать программу по разработанному алгоритму как отдельный модуль;

– выполнять отладку и тестирование программы на уровне модуля;

➤ Знать:

➤ 2. Проверяемые компетенции (код): ОК 5 -9. ПК 1.5- 1.6.

➤ 3. Пример оценочного средства (примерные тестовые задания, типовой вариант контрольной работы и др.)

Вопрос 1 Управляющая программа - это

d. программа, предназначенная для поддержания работоспособности СООИ или повышения эффективности ее использования.

e. программа, предназначенная для решения задачи или класса задач в определенной области применения СООИ

f. системная программа, реализующая набор функций управления, который включает в себя управление ресурсами и взаимодействие с внешней средой СООИ, восстановление работы системы после проявления неисправностей в технических средствах.

Вопрос 2 Объектный модуль – это

d. программа или функционально завершенный фрагмент программы, предназначенный для хранения, трансляции, объединения с другими программными модулями и загрузки в оперативную память.

e. программный модуль на исходном языке, обрабатываемый транслятором и представляемый для него как целое, достаточное для проведения трансляции.

f. программный модуль, получаемый в результате трансляции исходного модуля.

Вопрос 3 Язык высокого уровня – это

d. символьный язык программирования, предложения которого по своей структуре в основном подобны командам и обрабатываемым данным конкретного машинного языка.

e. язык программирования, понятия и структура которого удобны для восприятия человеком.

f. язык программирования, предназначенный для представления программы в форме, позволяющей выполнять ее непосредственно техническими средствами обработки информации.

Вопрос 4 ... - преобразование программы, представленной на одном языке программирования, в программу на другом языке программирования, в определенном смысле равносильную первой.

d. Интерпретация

e. Трансляция

f. Компиляция

Вопрос 5 Прикладное программное обеспечение – это

d. совокупность программ СООИ и программных документов, необходимых для их эксплуатации

e. комплекс программ, которые обеспечивают управление компонентами компьютерной системы

f. компьютерные программы, написанные для пользователей или самими пользователями, для задания компьютеру конкретной работы

Вопрос 6 Язык Ассемблер - это

- d. система записи программы с детализацией до отдельной машинной команды
- e. язык программирования, понятия и структура которого удобны для восприятия человеком
- f. система, образуемая языком программирования, компилятором или интерпретатором программ

Вопрос 7 Какие регистры относятся к регистрам данных:

- f. Es
- g. Dх
- h. Ip
- i. Cs
- j. Al

Вопрос 8 Команда add –

- d. Команда сложения
- e. Команда умножения
- f. Команда вычитания

Вопрос 9 Какая команда относится к арифметическим командам:

- a. In
- b. Or
- c. Inc

Вопрос 10 Команда ... повторяет действия несколько раз

- a. Безусловного перехода
- b. Условного перехода
- c. Цикла

Вопрос 11 Ссылка на определенное место в программе - ...

- a. Адресация
- b. Метка
- c. Переход

Вопрос 12 Прерывание, которое возникает при существовании ошибок в программе:

- a. Внутренние
- b. Аппаратное
- c. Программное

Вопрос 13 Какой командой осуществляется начало процедуры:

- a. Proc
- b. Call
- c. Ret

Вопрос 14 Прерывания – это

- a. Символическое имя для определения места в памяти, где размещены данные
- b. Блок команд, который может быть вызван из любого места основной программы
- c. Процесс приостановки программы

Вопрос 15 Какие команды относятся к директивам:

- a. DD
- b. DE
- c. DB
- d. DC
- e. DW

Вопрос 16 Определить ошибки в программе:

- 1. Var a;boolean;
- 2. begin
- 3. asm
- 4. Mov AX,00111011b
- 5. Or Al,00001111b
- 6. Mov a,AL
- 7. end;
- 8. Writeln('A=',A);
- 9. end.

- a. Строка 1,4
- b. Строка 1,5
- c. Строка 4,6

Вопрос 17 Определить результат выполнения программы:

```
Var x:integer;  
Begin  
X:=50;  
asm  
mov ax,x  
sub ax,13  
sub ax,4  
sub ax,20  
mov x,ax  
end;  
writeln('x=',x);  
end.
```

- a. 13
- b. 14
- c. 15

Вопрос 18

Управляющая программа - это

- g. программа, предназначенная для поддержания работоспособности СООИ или повышения эффективности ее использования.
- h. программа, предназначенная для решения задачи или класса задач в определенной области применения СООИ
- i. системная программа, реализующая набор функций управления, который включает в себя управление ресурсами и взаимодействие с внешней средой СООИ, восстановление работы системы после проявления неисправностей в технических средствах.

Вопрос 19 Объектный модуль – это

- g. программа или функционально завершенный фрагмент программы, предназначенный для хранения, трансляции, объединения с другими программными модулями и загрузки в оперативную память.
- h. программный модуль на исходном языке, обрабатываемый транслятором и представляемый для него как целое, достаточное для проведения трансляции.
- i. программный модуль, получаемый в результате трансляции исходного модуля.

Вопрос 20 Язык высокого уровня – это

- g. символьный язык программирования, предложения которого по своей структуре в основном подобны командам и обрабатываемым данным конкретного машинного языка.
- h. язык программирования, понятия и структура которого удобны для восприятия человеком.
- i. язык программирования, предназначенный для представления программы в форме, позволяющей выполнять ее непосредственно техническими средствами обработки информации.

Вопрос 21 ... - преобразование программы, представленной на одном языке программирования, в программу на другом языке программирования, в определенном смысле равносильную первой.

- g. Интерпретация
- h. Трансляция
- i. Компиляция

Вопрос 22 Прикладное программное обеспечение – это

- g. совокупность программ СОО и программных документов, необходимых для их эксплуатации
- h. комплекс программ, которые обеспечивают управление компонентами компьютерной системы
- i. компьютерные программы, написанные для пользователей или самими пользователями, для задания компьютеру конкретной работы

Вопрос 23 Язык Ассемблер - это

- g. система записи программы с детализацией до отдельной машинной команды
- h. язык программирования, понятия и структура которого удобны для восприятия человеком
- i. система, образуемая языком программирования, компилятором или интерпретатором программ

Вопрос 24 Какие регистры относятся к регистрам данных:

- k. Es
- l. Dх
- m. Ip
- n. Cs
- o. Al

Вопрос 25 Команда add –

- g. Команда сложения
- h. Команда умножения
- i. Команда вычитания

4. Критерии оценивания:

Оценка «отлично»	ставится студенту, если студент из 25 вопросов ответил правильно от 23 до 25 вопросов.
Оценка «хорошо»	ставится студенту, если студент из 25 вопросов ответил правильно от 18 до 22 вопросов.
Оценка «удовлетворительно»	ставится, если студент из 25 вопросов ответил правильно от 13 до 17 вопросов.
Оценка «неудовлетворительно»	если студент из 25 вопросов ответил правильно на 12 вопросов и менее.

5. Рекомендуемый перечень вопросов для самостоятельной подготовки:

1. Выполнение СОМ программы Выполнение EXE программы на языке С++ и Ассемблере.
2. Ассемблер. Функции загрузки и выполнения программы Основные понятия.
3. Сравнение макросредств и подпрограмм. Некоторые возможности Макроязыка.
4. Локальные переменные, макроопределения. Присваивание значений переменным макроопределения.
5. Ассемблер. Глобальные переменные макроопределения Уникальные метки.
6. Кросс системы. Вычислительные системы. Модель регистров.
7. Модель оперативной памяти Модель процессора.
8. Ошибки программирования. Классификация ошибок программирования.
9. Ошибки при задании необходимых начальных условий для отдельных программ.
10. Распознавание ошибок Ассемблером. Распространенные ошибки в драйверах ввода /вывода.
11. Распространенные ошибки Ассемблера в программах прерывания.

➤ ОЦЕНОЧНОЕ СРЕДСТВО

➤ ПР 3 Контрольный тест «Системное программирование»

1. Цель – проверить навыки по разработке кода программного модуля на языках программирования низкого уровня;



- Уметь: - создавать программу по разработанному алгоритму как отдельный модуль;
– выполнять отладку и тестирование программы на уровне модуля;

- Знать:

- 2. Проверяемые компетенции (код): ОК 6 -9. ПК 1.4- 1.6.

- 3. Пример оценочного средства (примерные тестовые задания, типовой вариант контрольной работы и др.)

Вопрос 1 Управляющая программа - это

- а. программа, предназначенная для поддержания работоспособности СОИ или повышения эффективности ее использования.

- b. программа, предназначенная для решения задачи или класса задач в определенной области применения СОО
- c. системная программа, реализующая набор функций управления, который включает в себя управление ресурсами и взаимодействие с внешней средой СОО, восстановление работы системы после проявления неисправностей в технических средствах.

Вопрос 2 Программный модуль – это

- a. программа или функционально завершенный фрагмент программы, предназначенный для хранения, трансляции, объединения с другими программными модулями и загрузки в оперативную память.
- b. программный модуль на исходном языке, обрабатываемый транслятором и представляемый для него как целое, достаточное для проведения трансляции.
- c. программный модуль, получаемый в результате трансляции исходного модуля.

Вопрос 3 Машинный язык – это

- a. символьный язык программирования, предложения которого по своей структуре в основном подобны командам и обрабатываемым данным конкретного машинного языка.
- b. язык программирования, понятия и структура которого удобны для восприятия человеком.
- c. язык программирования, предназначенный для представления программы в форме, позволяющей выполнять ее непосредственно техническими средствами обработки информации.

Вопрос 4 ... - преобразование программы на машинный язык.

- a. Интерпретация
- b. Трансляция
- c. Компиляция

Вопрос 5 Системное программное обеспечение– это

- a. совокупность программ СОО и программных документов, необходимых для их эксплуатации
- b. комплекс программ, которые обеспечивают управление компонентами компьютерной системы
- c. компьютерные программы, написанные для пользователей или самими пользователями, для задания компьютеру конкретной работы

Вопрос 6 Адресация предназначена ...

- a. Для указания адреса ячеек
- b. Для обмена данными
- c. Для помещения данных в регистры

Вопрос 7 Назовите регистр указания команд:

- a. Es
- b. Dх
- c. Ip
- d. Cs
- e. Al

Вопрос 8 Команда pop –

- a. Команда помещения в стек
- b. Команда перемещения данных
- c. Команда извлечения из стека

Вопрос 9 Какая команда относится к битовым командам:

- a. Bt
- b. Or
- c. Inc

Вопрос 10 Команда ... начинает выполнение с новой ветки в случае, если условие ложно

- a. Безусловного перехода
- b. Условного перехода
- c. Цикла

Вопрос 11 Объявляется в Паскале - ...

- a. Локальная метка
- b. Глобальная метка
- c. Системная метка

Вопрос 12 Прерывание, которое возникает при обнаружении ошибок в работе процессора:

- a. Внутренние
 - b. Аппаратное
 - c. Программное
1. Какой командой осуществляется возврат из процедуры:
- a. Proc
 - b. Call
 - c. Ret

Вопрос 13 Подпрограмма – это

- a. Символическое имя для определения места в памяти, где размещены данные
- b. Блок команд, который может быть вызван из любого места основной программы
- c. Процесс приостановке программы

Вопрос 14 Какие функции относятся к прерыванию 21H:

- a. 02h
- b. 09h
- c. 08h
- d. 2ah
- e. 5fh

Вопрос 15 Определить ошибки в программе:

- 1. Var
- 2. str:='gruppa\$';
- 3. asm
- 4. Mov Cx,3
- 5. @Lab:
- 6. Mov Ah,9
- 7. Mab Dx,offset str
- 8. int 21h
- 9. Loop @Lab
- 10. end;
- 11. readln;
- 12. end.

- a. Строка 4,6

b. Строка 4,7

c. Строка 6,7

Вопрос 16 Определить результат выполнения программы:

```
Begin
asm
Mov ax,6;
Mov bx,8;
Mov dx, ax;
cmp ax, bx;
jae Li;
Mov dx, bx;
Li:
mov dx,ax
end;
end.
```

a. 6

b. 8

c. 14

Вопрос 17 Определить результат выполнения программы:

```
Begin
asm
Mov Ah,02h
Mov Dl,'*'
int 21h
end;
end.
```

Вопрос 18

Управляющая программа - это

j. программа, предназначенная для поддержания работоспособности СООИ или повышения эффективности ее использования.

k. программа, предназначенная для решения задачи или класса задач в определенной области применения СООИ

l. системная программа, реализующая набор функций управления, который включает в себя управление ресурсами и взаимодействие с внешней средой СООИ, восстановление работы системы после проявления неисправностей в технических средствах.

Вопрос 19 Объектный модуль – это

j. программа или функционально завершенный фрагмент программы, предназначенный для хранения, трансляции, объединения с другими программными модулями и загрузки в оперативную память.

k. программный модуль на исходном языке, обрабатываемый транслятором и представляемый для него как целое, достаточное для проведения трансляции.

l. программный модуль, получаемый в результате трансляции исходного модуля.

Вопрос 20 Язык высокого уровня – это

j. символьный язык программирования, предложения которого по своей структуре в основном подобны командам и обрабатываемым данным конкретного машинного языка.

- к. язык программирования, понятия и структура которого удобны для восприятия человеком.
- л. язык программирования, предназначенный для представления программы в форме, позволяющей выполнять ее непосредственно техническими средствами обработки информации.

Вопрос 21 ... - преобразование программы, представленной на одном языке программирования, в программу на другом языке программирования, в определенном смысле равносильную первой.

- ж. Интерпретация
- к. Трансляция
- л. Компиляция

Вопрос 22 Прикладное программное обеспечение – это

- ж. совокупность программ СОО и программных документов, необходимых для их эксплуатации
- к. комплекс программ, которые обеспечивают управление компонентами компьютерной системы
- л. компьютерные программы, написанные для пользователей или самими пользователями, для задания компьютеру конкретной работы

Вопрос 23 Язык Ассемблер - это

- ж. система записи программы с детализацией до отдельной машинной команды
- к. язык программирования, понятия и структура которого удобны для восприятия человеком
- л. система, образуемая языком программирования, компилятором или интерпретатором программ

Вопрос 24 Какие регистры относятся к регистрам данных:

- р. Es
- q. Dx
- г. Ip
- с. Cs
- т. Al

Вопрос 25 Команда add –

- ж. Команда сложения
- к. Команда умножения
- л. Команда вычитания

4. Критерии оценивания:

Оценка «отлично»	ставится студенту, если студент из 25 вопросов ответил правильно от 23 до 25 вопросов.
Оценка «хорошо»	ставится студенту, если студент из 25 вопросов ответил правильно от 18 до 22 вопросов.
Оценка «удовлетворительно»	ставится, если студент из 25 вопросов ответил правильно от 13 до 17 вопросов.
Оценка «неудовлетворительно»	если студент из 25 вопросов ответил правильно на 12 вопросов и менее.

5. Рекомендуемый перечень вопросов для самостоятельной подготовки:

1. Выполнение СОМ программы Выполнение EXE программы на языке С++ и Ассемблере.
2. Ассемблер. Функции загрузки и выполнения программы Основные понятия.
3. Сравнение макросредств и подпрограмм. Некоторые возможности Макроязыка.
4. Локальные переменные, макроопределения. Присваивание значений переменным макроопределения.
5. Ассемблер. Глобальные переменные макроопределения Уникальные метки.
6. Кросс системы. Вычислительные системы. Модель регистров.
7. Модель оперативной памяти Модель процессора.
8. Ошибки программирования. Классификация ошибок программирования.
9. Ошибки при задании необходимых начальных условий для отдельных программ.
10. Распознавание ошибок Ассемблером. Распространенные ошибки в драйверах ввода /вывода.
11. Распространенные ошибки Ассемблера в программах прерывания.

➤ **ОЦЕНОЧНОЕ СРЕДСТВО**

ПР 4 Деловая (ролевая) игра
по дисциплине «**Системное программирование**»

1. Цель - Закрепление и расширение следующих знаний студентов по знанию основных понятий в области системного программирования.

- Уметь: – выполнять отладку и тестирование программы на уровне модуля;
– оформлять документацию на программные средства;
– использовать инструментальные средства для автоматизации оформления документации.

Знать: – основные принципы отладки и тестирования программных продуктов;
– методы и средства разработки технической документации.

➤ **2. Проверяемые компетенции (код): ОК 7 - 9. ПК 1.5-1.6.**

➤ **3. Пример оценочного средства (типовой вариант деловой игры.)**

Системное программирование

Программа - это данные, предназначенные для управления конкретными компонентами системы обработки информации (СОИ) в целях реализации определенного алгоритма.

Определения даются по: ГОСТ 19781-90. Обеспечение систем обработки информации программное. Термины и определения. - М.:Изд-во стандартов, 1990.

Обратить внимание: программа - это данные. Один из основных принципов машины фон Неймана - то, что и программы, и данные хранятся в одной и той же памяти. Сохраняемая в памяти программа представляет собой некоторые коды, которые могут рассматриваться как данные. Возможно, с точки зрения программиста программа - активный компонент, она выполняет некоторые действия. Но с точки зрения процессора команды программы - это данные, которые процессор читает и интерпретирует. С другой стороны программа - это данные с точки зрения обслуживающих программ, например, с точки зрения компилятора, который на входе получает одни данные - программу на языке высокого уровня (ЯВУ), а на выходе выдает другие данные - программу в машинных кодах.

Программное обеспечение (ПО) - совокупность программ СОИ и программных

документов, необходимых для их эксплуатации

Существенно, что ПО - это программы, предназначенные для многократного использования и применения разными пользователями. В связи с этим следует обратить внимание на ряд необходимых свойств ПО.

1. Необходимость документирования. По определению программы становятся ПО только при наличии документации. Конечный пользователь не может работать, не имея документации. Документация делает возможным тиражирование ПО и продажу его без его разработчика. По Бруксу ошибкой в ПО является ситуация, когда программное изделие функционирует не в соответствии со своим описанием, следовательно, ошибка в документации также является ошибкой в программном изделии.
2. Эффективность. ПО, рассчитанное на многократное использование (например, ОС, текстовый редактор и т.п.) пишется и отлаживается один раз, а выполняется многократно. Таким образом, выгодно переносить затраты на этап производства ПО и освобождать от затрат этап выполнения, чтобы избежать тиражирования затрат.
3. Надежность. В том числе:
 - Тестирование программы при всех допустимых спецификациях входных данных
 - Защита от неправильных действий пользователя
 - Защита от взлома - пользователи должны иметь возможность взаимодействия с ПО только через легальные интерфейсы.

"Ошибки в системе возможны из-за сбоев аппаратуры, ошибок ПО, неправильных действий пользователя. Первые - неизбежны, вторые - вероятны, третьи - гарантированы". Появление ошибок любого уровня не должно приводить к краху системы. Ошибки должны вылавливаться диагностироваться и (если их невозможно исправить) превращаться в корректные отказы.

Системные структуры данных должны сохраняться безусловно. Сохранение целостности пользовательских данных желательно.

4. Возможность сопровождения. Возможные цели сопровождения - адаптация ПО к конкретным условиям применения, устранение ошибок, модификация. Во всех случаях требуется тщательное структурирование ПО и носителем информации о структуре ПО должна быть программная документация. Адаптация во многих случаях м.б. передоверена пользователю - при тщательной отработке и описании сценариев инсталляции и настройки. Исправление ошибок требует развитой сервисной службы, собирающей информацию об ошибках и формирующей исправляющие пакеты. Модификация предполагает изменение спецификаций на ПО. При этом, как правило, должны поддерживаться и старые спецификации. Эволюционное развитие ПО экономит вложения пользователей.

Системное программирование

Системная программа - программа, предназначенная для поддержания работоспособности СОИ или повышения эффективности ее использования.

Прикладная программа - программа, предназначенная для решения задачи или класса задач в определенной области применения СОИ.

В соответствии с терминологией, системное программирование - это процесс разработки системных программ (в т.ч., управляющих и обслуживающих).

С другой стороны, по определению Гегеля система - единое целое, состоящее из множества компонентов и множества связей между ними. Тогда системное программирование - это разработка программ сложной структуры.

Эти два определения не противоречат друг другу, так как разработка программ сложной структуры ведется именно для обеспечения работоспособности или повышения эффективности СОО.

Зафиксированное в ГОСТ подразделение ПО на системное и прикладное является до некоторой степени устаревшим. Сегодняшнее деление предусматривает по меньшей мере три градации ПО:

- Системное
- Промежуточное
- Прикладное

Промежуточное ПО (middleware) мы определяем как совокупность программ, осуществляющих управление вторичными (конструируемыми самим ПО) ресурсами, ориентированными на решение определенного (широкого) класса задач. К такому ПО относятся менеджеры транзакций, серверы БД, серверы коммуникаций и другие программные серверы. С точки зрения инструментальных средств разработки промежуточное ПО ближе к прикладному, так как не работает на прямую с первичными ресурсами, а использует для этого сервисы, предоставляемые системным ПО. С точки зрения алгоритмов и технологий разработки промежуточное ПО ближе к системному, так как всегда является сложным программным изделием многократного и многоцелевого использования и в нем применяются те же или сходные алгоритмы, что и в системном ПО.

Современные тенденции развития ПО состоит в снижении объема как системного, так и прикладного программирования. Основная часть работы программистов выполняется в промежуточном ПО. Снижение объема системного программирования определено современными концепциями ОС, объектно-ориентированной архитектурой и архитектурой микроядра, в соответствии с которыми большая часть функций системы выносятся в утилиты, которые можно отнести и к промежуточному ПО. Снижение объема прикладного программирования обусловлено тем, что современные продукты промежуточного ПО предлагают все больший набор инструментальных средств и шаблонов для решения задач своего класса.

Значительная часть системного и практически все прикладное ПО пишется на языках высокого уровня, что обеспечивает сокращение расходов на их разработку/модификацию и переносимость.

Системное ПО подразделяется на системные управляющие программы и системные обслуживающие программы.

Управляющая программа - системная программа, реализующая набор функций управления, который включает в себя управление ресурсами и взаимодействие с внешней средой СОО, восстановление работы системы после проявления неисправностей в технических средствах.

Программа обслуживания (утилита) - программа, предназначенная для оказания услуг общего характера пользователям и обслуживающему персоналу СОО.

Управляющая программа совместно с набором необходимых для эксплуатации системы утилит составляют операционную систему (ОС).

Кроме входящих в состав ОС утилит могут существовать и другие утилиты (того же или стороннего производителя), выполняющие дополнительное (опционное) обслуживание. Как правило, это утилиты, обеспечивающие разработку программного обеспечения для операционной системы.

Система программирования - система, образуемая языком программирования, компилятором или интерпретатором программ, представленными на этом языке, соответствующей документацией, а также вспомогательными средствами для подготовки программ к форме, пригодной для выполнения.

Этапы подготовки программы

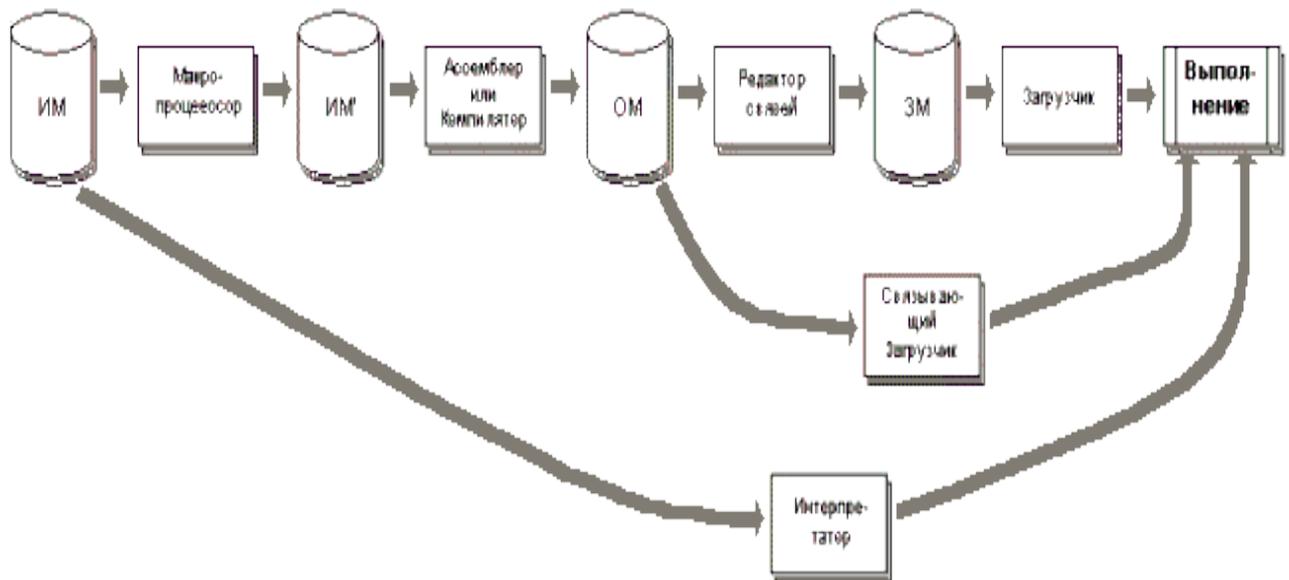
При разработке программ, а тем более - сложных, используется принцип модульности, разбиения сложной программы на составные части, каждая из которых может подготавливаться отдельно. Модульность является основным инструментом структурирования программного изделия, облегчающим его разработку, отладку и сопровождение.

Программный модуль - программа или функционально завершенный фрагмент программы, предназначенный для хранения, трансляции, объединения с другими программными модулями и загрузки в оперативную память.

При выборе модульной структуры должны учитываться следующие основные соображения:

- Функциональность - модуль должен выполнять законченную функцию
- Несвязность - модуль должен иметь минимум связей с другими модулями, связь через глобальные переменные и области памяти нежелательна
- Специфицируемость - входные и выходные параметры модуля должны четко формулироваться

На рисунке показаны этапы, которые проходит программа от своего написания до выполнения



Программа пишется в виде исходного модуля, на рисунке - файл ИМ.

Исходный модуль - программный модуль на исходном языке, обрабатываемый транслятором и представляемый для него как целое, достаточное для проведения трансляции.

Первым (не для всех языков программирования обязательным) этапом подготовки программы является обработка ее Макропроцессором (или Препроцессором). Макропроцессор обрабатывает текст программы и на выходе его получается новая редакция текста (на рис. - ИМ'). В большинстве систем программирования Макропроцессор совмещен с транслятором, и для программиста его работа и промежуточный ИМ' "не видны". Следует иметь в виду, что Макропроцессор выполняет обработку текста, это означает, с одной стороны, что он "не понимает" операторов языка программирования и "не знает" переменных программы, с другой, что все операторы и переменные Макроязыка (тех выражений в программе, которые адресованы Макропроцессору) в промежуточном ИМ' уже отсутствуют и для дальнейших этапов обработки "не видны". Так, если Макропроцессор заменил в программе некоторый текст А на текст В, то транслятор уже видит только текст В, и не знает, был этот текст написан

программистом "своей рукой" или подставлен Макропроцессором.

Следующим этапом является трансляция.

Трансляция - преобразование программы, представленной на одном языке программирования, в программу на другом языке программирования, в определенном смысле (логическом, функциональном) равносильную первой.

Как правило, выходным языком транслятора является машинный язык целевой вычислительной системы. (Целевая ВС - та ВС, на которой программа будет выполняться.)

Машинный язык - язык программирования, предназначенный для представления программы в форме, позволяющей выполнять ее непосредственно техническими средствами обработки информации.

Трансляторы - общее название для программ, осуществляющих трансляцию. Они подразделяются на Ассемблеры и Компиляторы - в зависимости от исходного языка программы, которую они обрабатывают. Ассемблеры работают с Автокодами или языками Ассемблера, Компиляторы - с языками высокого уровня.

Автокод - символьный язык программирования, предложения которого по своей структуре в основном подобны командам и обрабатываемым данным конкретного машинного языка.

Язык Ассемблера - язык программирования, который представляет собой символьную форму машинного языка с рядом возможностей, характерных для языка высокого уровня (обычно включает в себя макросредства).

Язык высокого уровня - язык программирования, понятия и структура которого удобны для восприятия человеком.

Объектный модуль - программный модуль, получаемый в результате трансляции исходного модуля.

Поскольку результатом трансляции является модуль на языке, близком к машинному, в нем уже не остается признаков того, на каком исходном языке был написан программный модуль. Это создает принципиальную возможность создавать программы из модулей, написанных на разных языках. Специфика исходного языка, однако, может сказываться на физическом представлении базовых типов данных, способах обращения к процедурам/функциям и т.п. Для совместимости разноязыковых модулей должны выдерживаться общие соглашения.

Большая часть объектного модуля - команды и данные машинного языка именно в той форме, в какой они будут существовать во время выполнения программы. Однако, программа в общем случае состоит из многих модулей. Поскольку транслятор обрабатывает только один конкретный модуль, он не может должным образом обработать те части этого модуля, в которых запрограммированы обращения к данным или процедурам, определенным в другом модуле. Такие обращения называются внешними ссылками. Те места в объектном модуле, где содержатся внешние ссылки, транслируются в некоторую промежуточную форму, подлежащую дальнейшей обработке. Говорят, что объектный модуль представляет собой программу на машинном языке с неразрешенными внешними ссылками.

Разрешение внешних ссылок выполняется на следующем этапе подготовки, который обеспечивается Редактором Связей (Компоновщиком). Редактор Связей соединяет вместе все объектные модули, входящие в программу. Поскольку Редактор Связей "видит" уже все компоненты программы, он имеет возможность обработать те места в объектных модулях, которые содержат внешние ссылки. Результатом работы Редактора Связей является загрузочный модуль.

Загрузочный модуль - программный модуль, представленный в форме, пригодной для загрузки в оперативную память для выполнения.

Загрузочный модуль сохраняется в виде файла на внешней памяти. Для выполнения программа должна быть перенесена (загружена) в оперативную память. Иногда при этом

требуется некоторая дополнительная обработка (например, настройка адресов в программе на ту область оперативной памяти, в которую программа загрузилась). Эта функция выполняется Загрузчиком, который обычно входит в состав операционной системы.

Возможен также вариант, в котором редактирование связей выполняется при каждом запуске программы на выполнение и совмещается с загрузкой. Это делает Связывающий Загрузчик. Вариант связывания при запуске более расходный, т.к. затраты на связывание тиражируются при каждом запуске. Но он обеспечивает:

- большую гибкость в сопровождении, так как позволяет менять отдельные объектные модули программы, не меняя остальных модулей;
- экономию внешней памяти, т.к. объектные модули, используемые во многих программах не копируются в каждый загрузочный модуль, а хранятся в одном экземпляре.

Вариант интерпретации подразумевает прямое исполнение исходного модуля.

Интерпретация - реализация смысла некоторого синтаксически законченного текста, представленного на конкретном языке.

Интерпретатор читает из исходного модуля очередное предложение программы, переводит его в машинный язык и выполняет. Все затраты на подготовку тиражируются при каждом выполнении, следовательно, интерпретируемая программа принципиально менее эффективна, чем транслируемая. Однако, интерпретация обеспечивает удобство разработки, гибкость в сопровождении и переносимость.

Примеры интерпретаторов: языки процедур (sell, REXX), JVM.

Не обязательно подготовка программы должна вестись на той же вычислительной системе и в той же операционной среде, в которых программа будет выполняться. Системы, обеспечивающие подготовку программ в среде, отличной от целевой называются кросс-системами. В кросс-системе может выполняться вся подготовка или ее отдельные этапы:

- Макрообработка и трансляция
- Редактирование связей
- Отладка

Типовое применение кросс-систем - для тех случаев, когда целевая вычислительная среда просто не имеет ресурсов, необходимых для подготовки программ, например, встроенные системы.

Программные средства, обеспечивающие отладку программы на целевой системе можно также рассматривать как частный случай кросс-системы.

Порядок деловой игры. Все студенты разбиваются на группы по 3—4 человека. Одна группа будет представлять экспертную комиссию, а остальные — рабочие группы.

Предполагается, что экспертная комиссия — это представители высшего звена (директор по производству, директор по маркетингу, директор по логистике, коммерческий директор и др.). Рабочая группа — это представители среднего звена (маркетологи, логисты, сбытовики), которым после проведенного экстренного консилиума было выдано задание: разработать проект реорганизации подсистемы распределения с учетом сложившихся внутренних и внешних условий.

Таким образом, каждая рабочая группа должна самостоятельно изучить ситуацию и последовательно ответить на предлагаемые ниже вопросы. В результате у каждой группы получится свой вариант развития событий, который ей предстоит представить на суд экспертной комиссии.

Экспертная комиссия в свою очередь должна разработать систему оценки проектов на основе сформулированного ниже задания и оценить все проекты соответствующе.

По итогам работы экспертная комиссия выносит решение о победившем проекте.

Примечание. Для получения адекватного результата предлагается ввести в игру координатора экспертной комиссии, которым должен выступать преподаватель.

4. Критерии оценивания

Оценка «отлично»	студент являлся активным участником игры и принимал участие в последующем ее анализе, выявляя и исправляя допущенные ошибки;
Оценка «хорошо»	студент являлся только активным участником игры и не принимал участие в ее анализе;
Оценка «удовлетворительно»	студент являлся пассивным участником игры, но принимал активное участие в ее анализе, выявляя и исправляя допущенные ошибки.
Оценка «неудовлетворительно»	выставляется студенту, если он являлся пассивным участником игры и не принимал участие в ее последующем анализе.

5. Рекомендуемый перечень вопросов для самостоятельной подготовки:

1. Что представляет сой язык *Ассемблера*?
2. Назовите основные операторы *Ассемблера* ?
- 3 В чем основные различия *Автокода и Ассемблера* ?
- 4 Что представляет сой язык программный модуль, получаемый в результате трансляции исходного модуля?
- 5 Что представляет собой *Системная программа* ?

II ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

1. Форма проведения промежуточной аттестации: дифференцированный зачет в 3 семестре.
2. Перечень вопросов для аттестации

Вопросы для зачета

1. Системное программирование. Этапы подготовки программы .
2. Программирование на языке Ассемблера. Предложения языка Ассемблера .
3. Регистры. Биты и байты ASCII
4. Двоичные числа. Шестнадцатеричное представление.
5. Сегменты. Расширение набора команд.
6. Способы адресации. Директивы. Директивы определения данных.
7. Директива определения байта (DB). Директива определения слова (DW).
8. Директива определения двойного слова (DD). Директива определения учетверенного слова (DQ).
9. Директива определения десяти байт (DT). Непосредственные операнды Директива EQU.
10. Сегментные регистры: CS, DS, SS и ES.
11. Регистры общего назначения: AX, BX, CX и DX.
12. Регистровые указатели: SP и BP. Индексные регистры: SI и DI .
13. Регистр командного указателя IP. Флаговый регистр.
14. Обработка двоичных данных. Беззнаковые и знаковые данные.
15. Ассемблер. Умножение. Сдвиг регистровой пары DX:AX Деление.
16. Ассемблер. Преобразование знака Обработка данных в форматах ASCII и BCD.
17. Ассемблер. Двоично-десятичный формат (BCD)Сдвиг и округление.
18. Ассемблер. Команда прерывания INT. Установка курсора. Очистка экрана .
19. Ассемблер. Использование символов возврата каретки, конца строки и табуляции для вывода на экран
20. Ассемблер. Расширенные возможности экранных операций Расширенный ASCII код.
21. Ассемблер. Другие операции ввода/вывода Ввод с клавиатуры по команде BIOS INT 16H.

22. Ассемблер. Функциональные клавиши Цвет и графика.
23. Ассемблер. Межсегментные вызовы. Атрибуты EXTRN и PUBLIC.
24. Компоновка программ на языке С++ и Ассемблере.
25. Выполнение COM программы Выполнение EXE программы на языке С++ и Ассемблере.
26. Ассемблер. Функции загрузки и выполнения программы Основные понятия.
27. Сравнение макросредств и подпрограмм. Некоторые возможности Макроязыка.
28. Локальные переменные, макроопределения. Присваивание значений переменным макроопределения.
29. Ассемблер. Глобальные переменные макроопределения Уникальные метки.
30. Кросс системы. Вычислительные системы. Модель регистров.
31. Модель оперативной памяти Модель процессора.
32. Ошибки программирования. Классификация ошибок программирования.
33. Ошибки при задании необходимых начальных условий для отдельных программ.
34. Распознавание ошибок Ассемблером. Распространенные ошибки в драйверах ввода /вывода.
35. Распространенные ошибки Ассемблера в программах прерывания.

3. Критерии оценивания заданий к дифференцированному зачету

Оценка «отлично»	Представлен развернутый ответ на теоретический вопрос, а также поэтапное решение практического задания с пояснениями. Студент ориентируется в излагаемом материале, отвечает на дополнительные вопросы, связанные демонстрирует глубокие теоретические знания, знание первоисточников.
Оценка «хорошо»	Представлен достаточно развернутый ответ на теоретический вопрос, а также поэтапное решение практического задания с пояснениями. В решении практического задания могут быть допущены вычислительные ошибки, не искажающие лежащего в основе решения алгоритма Студент уверенно отвечает на дополнительные вопросы, демонстрирует достаточно высокий уровень теоретических знаний, знание первоисточников.
Оценка «удовлетворительно»	Представлен не полный ответ на теоретический вопрос, В решении практического задания могут быть допущены вычислительные ошибки, не искажающие лежащего в основе решения алгоритма. Студент демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.
Оценка «неудовлетворительно»	Теоретический вопрос не раскрыт, в решении практического задания допущены существенные ошибки, ввиду незнания алгоритмов решения. Либо дан ответ только на один из вопросов билета. Студент затрудняется отвечать на дополнительные вопросы, в том числе непосредственно относящиеся к сути теоретического и практического вопросов билета.

Проектная деятельность и управление проектами

ПАСПОРТ ОЦЕНОЧНЫХ СРЕДСТВ

- ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам
- ОК 02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.
- ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие.
- ОК 04. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами
- ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенно стей социального и культурного контекста.
- ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, применять стандарты антикоррупционного поведения
- ОК 07.: Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях
- ОК 08. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности
- ОК 09. Использовать информационные технологии в профессиональной деятельности.
- ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языке
- ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.
- ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием
- ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств.
- ПК 1.4. Выполнять тестирование программных модулей.
- ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода.
- ПК 1.6. Разрабатывать модули программного обеспечения для мобильных платформ.

Перечень формируемых компетенций:

№ п/п	Контролируемые разделы (темы)	Код контролируемой компетенции	Наименование оценочного средства
1	Создание учебного проекта.	ОК 01-ОК 10, ПК 1.1.- ПК 1.6.	Практическая работа №1
	Самостоятельная работа		
2	Создание нового проект на основе одного из имеющихся в программе шаблонов.	ОК 01-ОК 10, ПК 1.1.- ПК 1.6.	Практическая работа №2
	Самостоятельная работа		
3	Составление расписаний выполнения работ в проекте.	ОК 01-ОК 10, ПК 1.1.- ПК 1.6.	Практическая работа №3

№ п/п	Контролируемые разделы (темы)	Код контролируемой компетенции	Наименование оценочного средства
	Самостоятельная работа		
4	Создание контрольного проекта.	ОК 01-ОК 10, ПК 1.1.- ПК 1.6.	Практическая работа №4
	Самостоятельная работа		
5	Настройка ресурсов проекта в программе	ОК 01-ОК 10, ПК 1.1.- ПК 1.6.	Практическая работа №5
	Самостоятельная работа		
6	Анализ и оптимизация плана проекта. Формы и отчеты проектов.	ОК 01-ОК 10, ПК 1.1.- ПК 1.6.	Практическая работа №6
	Самостоятельная работа		
7	Формирование бизнес-цели индивидуального проекта. Разработка устава проекта.	ОК 01-ОК 10, ПК 1.1.- ПК 1.6.	Практическая работа №7
	Самостоятельная работа		
8	Подготовка проектных документов: плана и паспорта.	ОК 01-ОК 10, ПК 1.1.- ПК 1.6.	Практическая работа №8
	Самостоятельная работа		
9	Реализация индивидуального проекта в программной среде.	ОК 01-ОК 10, ПК 1.1.- ПК 1.6.	Практическая работа №9
	Самостоятельная работа		
10	Оформление презентации проекта Microsoft Office PowerPoint.	ОК 01-ОК 10, ПК 1.1.- ПК 1.6.	Практическая работа №10
	Самостоятельная работа		
11	Промежуточная аттестация	ОК 01-ОК 10, ПК 1.1.- ПК 1.6.	Экзамен

Сформированность выше перечисленных компетенций предполагает, что в результате освоения дисциплины (профессионального модуля) обучающийся должен:

В результате освоения учебной дисциплины обучающийся должен *знать*:

- основные принципы и методы организации, планирования и управления проектами;
- терминологию и основные нормы и стандарты, регулирующие деятельность организаций в области планирования и управления проектами;
- принципы разработки концепции и целей проекта;
- процедуру структуризации проекта;
- порядок разработки сметы проекта;

- процедуру подготовки и заключения контрактов, организации оптимальной процедуры закупок и поставок;
- принципы управления рисками проекта;
- методики управления временем и стоимостью проекта;
- методы контроля за ходом реализации проекта.

В результате освоения учебной дисциплины обучающийся должен **уметь**:

- осуществить системное планирование проекта на всех фазах его жизненного цикла;
- рассчитать график проекта с помощью инструментов календарного и сетевого планирования;
- управлять взаимодействиями в проекте;
- обеспечить эффективный контроль и регулирование, а также управление изменениями;
- использовать программные продукты для целей управления проектами;
- применять полученные в процессе обучения знания в практической деятельности по планированию и организации проектов в организациях.

В результате освоения учебной дисциплины обучающийся должен **владеть**:

- методами планирования проектов;
- методами бюджетирования проектов;
- методами анализа проектов;
- методами контроля за ходом реализации проектов.

I. ТЕКУЩИЙ КОНТРОЛЬ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ «ПРОЕКТНАЯ ДЕЯТЕЛЬНОСТЬ И УПРАВЛЕНИЕ ПРОЕКТАМИ»

ОЦЕНОЧНОЕ СРЕДСТВО: Практические работы

6. **Цель:** Проверить и систематизировать знания по основам проектной деятельности и управлением проектами.

7. **Проверяемые компетенции (код):** ОК 01-ОК 10, ПК 1.1.- ПК 1.6.

3. Пример оценочного средства:

Практическая работа №1

Создание проекта

График реализации проекта

Процесс разработки графика реализации проекта рассмотрим на примере, создадим график реализации проекта Консалтинговый центр.

Чтобы получить график реализации проекта надо:

- ☐☐☐ Задать дату начала реализации проекта
- ☐☐☐ Составить список задач проекта
- ☐☐☐ Связать задачи проекта
- ☐☐☐ Составить список ресурсов, необходимых для реализации проекта
- ☐☐☐ Назначить задачам ресурсы
- ☐☐☐ Вывести график реализации проекта на принтер

Дата начала реализации проекта

Чтобы задать дату начала реализации проекта, надо:

1. Из меню **Проект** выбрать команду **Сведения о проекте**.
2. В поле **Дата начала** появившегося диалогового окна ввести дату начала реализации проекта (для проекта Консалтинговый центр – 01.02.2023)
3. Щелкнуть на кнопке **ОК**.

Задачи проекта

Задача – это некоторая работа, которую надо выполнить.

Работа над проектом начинается с составления списка задач (работ), которые надо выполнить для достижения цели проекта. Сложные задачи, как правило, разбивают на несколько более простых задач. Таким образом, получается иерархический список задач проекта.

При составлении списка задач проекта используют метод, который часто называют методом «разбиения задач». Суть метода заключается в том, что сначала составляют список общих задач, затем общие задачи разбиваются на подзадачи, которые, в свою очередь, также могут быть разбиты на подзадачи более низкого уровня. Разбиение задач на подзадачи завершают, когда на выполнение задачи можно назначить конкретный ресурс (исполнителя).

Основной характеристикой каждой задачи проекта является длительность – время, необходимое для ее выполнения задачи. В общем случае длительность измеряется в рабочих днях.

Следует обратить внимание, что длительность общей задачи (задачи, которая разделена на подзадачи) определяется длительностью и порядком следования задач, из которых эта задача состоит. Длительность обобщенных задач вычисляет программное обеспечение.

Длительность подчиненной задачи нижнего уровня (элементарной задачи), т.е. задачи у которой нет подчиненных задач, определяется временем необходимым для ее выполнения. Длительность подчиненной задачи задает менеджер проекта на основе нормативной документации или своего опыта.

Задачи проекта (график реализации) принято изображать в виде диаграммы Ганта. В левой части диаграммы отображается список задач, в правой – прямоугольники, изображающие задачи. Расположение прямоугольников дает наглядное представление о последовательности выполнения задач проекта.

Ниже, в табл. 1, приведены задачи проекта **Консалтинговый центр**. Обратите внимание, что в качестве первой задачи указано название проекта. Также обратите внимание, что в таблице указана длительность только элементарных задач. Длительность общих задач вычислит программное обеспечение.

Таблица 1. Задачи проекта Консалтинговый центр

Задача	Длительность	Ответственный
Консалтинговый центр		
Компьютеры		
Покупка	2	Директор
Доставка	1	Лаборант
Монтаж	4	Лаборант
Локальная сеть		
Монтаж	2	Лаборант
Настройка	2	Лаборант
Реклама		
Подготовка рекламных материалов	3	Директор
Размещение	4	Секретарь
Учебно-методическое обеспечение		
Подготовка	10	Директор
Тиражирование	5	Секретарь
Начало работы	0	

Для задач, у которых нет подчиненных задач, в колонку **Длительность** введите время, необходимое для выполнения задачи, или время, в течение которого задача должна быть выполнена (длительность задачи). Единица измерения длительности задачи задается буквой русского алфавита (ч - час, н - неделя, м - месяц), которая вводится сразу за числом. Например, если длительность задачи **Подготовка рекламных материалов** равна трем дням, то в ячейку **Длительность** этой задачи следует ввести 3д.

На первом этапе работы над проектом, когда задачи проекта представлены в виде списка, программное обеспечение устанавливает, что все задачи начинаются одновременно, в момент начала реализации проекта. Позже, когда будут установлены *связи* между задачами, колонка **Начало** будет содержать правильную дату начала выполнения задач, колонка **Окончание** – дату завершения, а в колонке **Длительность** будет отражена длительность всех задач, в том числе и обобщенных.

В качестве примера на рис. 1 приведен вид окна **Диаграмма Ганта** после ввода нескольких задач проекта. Обратите внимание, что в колонке **Длительность**, после слова **день** отображается знак вопроса. Это показывает, что длительность задачи установлена «по умолчанию», а не задана пользователем явно.



Рис. 1. Список задач проекта

Способ ввода информации в ячейки таблицы стандартный. Используя клавиши перемещения курсора, выберите нужную ячейку (выбранная клетка выделяется рамкой) и печатайте текст. Для перехода к другой ячейке используйте клавиши перемещения курсора или щелкните левой кнопкой мышки в нужной клетке таблицы. Чтобы изменить содержимое ячейки, нажмите на клавиатуре клавишу <F2> и редактируйте текст обычным образом.

Корректировка списка задач проекта

Во время работы над проектом часто возникает необходимость внести изменения в список задач проекта: добавить новую задачу (в том числе не только в конец списка), удалить ошибочно введенную, изменить порядок следования задач.

Добавление задачи

Чтобы добавить в список новую задачу, надо выделить задачу (щелкнуть левой кнопкой мышки на находящемся в первой колонке номере задачи), перед которой нужно поместить новую задачу, и нажать клавишу <Insert>. В результате этих действий в список задач будет добавлена пустая строка, в которую можно ввести новую задачу.

Удаление задачи

Чтобы удалить задачу, надо выделить эту (щелкнуть левой кнопкой мышки на находящемся в первой колонке номере задачи) и из меню **Правка** выбрать команду **Удалить задачу** или нажать клавишу .

Перемещение задачи

Чтобы переместить задачу (или группу следующих одна за другой задач) в другое место списка, надо выделить нужную задачу (задачи) и из меню **Правка** выбрать команду **Вырезать задачу**. Затем выделить задачу, перед которой надо поместить выделенные на предыдущем шаге задачи, и из меню **Правка** выбрать команду **Вставить задачу**.

Вместо команд **Вырезать задачу** и **Вставить задачу** можно использовать находящиеся на панели инструментов соответствующие стандартные командные кнопки.

Формирование структуры проекта

Представление задач проекта в виде простого списка не достаточно наглядно. Простой список не отражает структуру проекта, связи между задачами, не позволяет видеть главные и подчиненные задачи. Гораздо удобнее задачи

проекта представить в виде иерархического списка, в котором задачи разделены по уровням.

Обычно во время работы над проектом сначала формулируется цель проекта (главная задача). Затем главная задача разбивается на фазы (крупные задачи), фазы разбиваются на задачи, задачи - на подзадачи более низкого уровня и т.д. до тех пор, пока не будут определены все задачи, необходимые для завершения проекта. Таким образом, проект можно рассматривать как совокупность общих и подчиненных задач. Общая задача – своего рода заголовок, она суммирует задачи нижнего уровня.

Перевести задачу с одного уровня иерархии на другой, сделать ее подчиненной или перевести на более высокий уровень (если задача является подчиненной) можно при помощи команд **На уровень выше** и **На уровень ниже**, кнопки активизации которых находятся на панели **Формирование**.



На уровень выше На уровень ниже

Чтобы понизить уровень задачи, сделать ее подчиненной, надо выделить эту задачу (щелкнуть на ее номере) и щелкнуть на кнопке **На уровень ниже**, на которой изображена направленная вправо стрелка.

Чтобы повысить уровень задачи, сделать ее обобщенной, для задач за ней следующих, надо выделить эту задачу и щелкнуть на кнопке **На уровень выше**, на которой изображена направленная влево стрелка.

Сформируйте структуру проекта. Сделайте задачи **Компьютеры - Локальная сеть - Реклама - УМО** подчиненными задаче **Консалтинговый центр**, задачи **Монтаж и Настройка** – подчиненными задаче **Локальная сеть** и т.д.

На диаграмме Ганта задачи, у которых нет подчиненных задач, изображаются прямоугольниками, общие задачи – скобками (рис. 2). Следует обратить внимание, если подчиненные задачи начинаются одновременно, то длительность общей задачи полагается равной длительности наиболее длительной подчиненной задачи. О том, как связать подчиненные задачи, т.е. указать, что одна задача должна начинаться, например, после окончания другой, будет сказано позже.



Рис. 2. Задачи проекта, представленные в виде иерархического списка

Проект можно просматривать с различной степенью детализации. Управление режимом отображения выполняется при помощи командных кнопок. Чтобы скрыть задачи, подчиненные текущей, щелкните на кнопке **Скрыть подзадачи**. Чтобы увидеть скрытые подзадачи, щелкните на кнопке **Показать подзадачи**. Чтобы увидеть все скрытые задачи проекта, щелкните на кнопке **Show All Tasks** (Показать все задачи).



Следует обратить внимание, что при копировании или перемещении обобщенной задачи все подчиненные задачи также копируются или перемещаются. Если надо переместить только обобщенную задачу, то сначала надо перевести подчиненные задачи на уровень обобщенной.

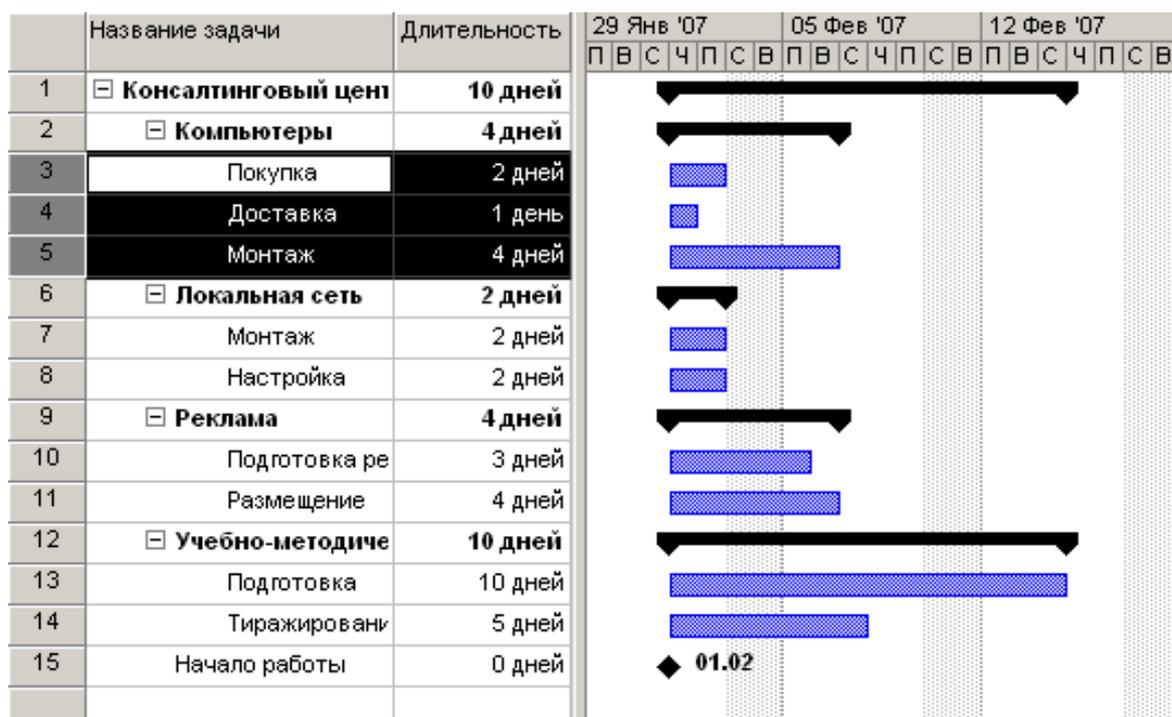
Связи между задачами

Задачи проекта связаны между собой во времени. Например, некоторые задачи не могут быть начаты до тех пор, пока не завершится некоторая предыдущая задача, другие могут выполняться одновременно. Поэтому после того как будет составлен список задач и задачи распределены по уровням (определены общие задачи и подзадачи), необходимо установить связи между задачами.

Чтобы связать несколько следующих в списке друг за другом несколько задач (в простейшем случае две задачи), надо:

1. Выделить задачи, которые надо связать (рис. 3). Для этого установить курсор мышки на номер первой задачи цепочки связываемых задач (обычно это задача с меньшим номером), нажать левую кнопку мышки и, удерживая ее нажатой, переместить курсор на номер последней задачи цепочки, отпустить кнопку мышки. В результате этих действий будут выделены несколько строк таблицы.

2. Щелкнуть на командной кнопке **Связать задачи** (рис. 4) или из меню **Правка** выбрать команду **Связать задачи**.



*Рис. 3. Чтобы связать задачи проекта, нужно выделить эти задачи и щелкнуть на кнопке **Связать задачи***



Рис. 4. Кнопки установки связи и разрыва связи между задачами

Чтобы связать задачи, которые не в списке не следуют одна за другой, надо:

1. Выделить первую задачу цепочки (щелкнуть левой кнопкой мышки на номере задачи).
2. Нажать клавишу <Ctrl> и, удерживая ее нажатой, сделать щелчок левой кнопкой мышки на номерах остальных задач цепочки.
3. Щелкнуть на командной кнопке **Связать задачи** или из меню **Правка** выбрать команду **Связать задачи**.

В качестве примера на рис. 5 приведен вид диаграммы Ганта после установки связей между задачами проекта.



Рис. 5. Диаграмма Ганта – график реализации проекта
Время задержки (опережения) выполнения задачи

Иногда между завершением одной задачи и началом другой должно пройти некоторое время. Например, после покрытия лаком, пол должен сохнуть 48 часов. Следовательно, задача *Оборудование* может начаться только через 48 часов после завершения задачи *Ремонт*. Такая ситуация моделируется при помощи связывания задач связью типа ОН и заданием запаздывания для задачи приемника. Для приведенного примера в поле Предшественники задачи Оборудование надо ввести 2ОН+2д, где: 2 - номер задачи Ремонт, ОН - тип связи между задачами, 2д - величина задержки начала выполнения задачи Оборудование после завершения задачи Ремонт. Время запаздывания также можно ввести во вкладке Предшественники окна Сведения о задаче, которое становится доступным в результате щелчка на соответствующей командной кнопке или в результате выбора в меню Проект команды Сведения о задаче.

Контрольные точки

Существуют особые задачи, которые называются контрольными точками. Эти задачи представляют собой важные события, а не задачи как таковые. Типичным примером контрольной точки является событие **Начало работы**. Чтобы ввести в график реализации контрольную точку, надо ввести задачу и задать, что время ее выполнения равно нулю.

Ресурсы проекта

Для выполнения проекта необходимы ресурсы – люди и оборудование. Чтобы задача была выполнена, на ее выполнение должен быть назначен соответствующий ресурс.

Процесс назначения ресурсов на выполнение задач называется распределением ресурсов. Здесь следует обратить внимание, что программное обеспечение позволяет рассчитать стоимость выполнения задач путем учета стоимости ресурсов, назначенных на их выполнение. Однако, на начальном этапе планирования проекта целью распределения ресурсов является назначение ответственных за решение задач, а не вычисление стоимости задач, путем детального учета рабочего времени, необходимого для решения задачи. Поэтому здесь не рассматриваются вопросы вычисления стоимости задач, контроля загрузки ресурсов.

Список ресурсов

Перед тем как приступить к распределению ресурсов, надо составить список ресурсов. Для этого надо в меню **Вид** выбрать команду **Лист ресурсов** и в появившуюся таблицу ввести названия и характеристики ресурсов. Применительно к рассматриваемой задаче достаточно ввести только названия ресурсов. В качестве названия ресурса можно указать должность или имя (фамилию) участника проекта.

Ресурсами проекта **Консалтинговый центр** являются директор, бухгалтер, секретарь и лаборант. Предполагается, что лаборант отвечает за обслуживание вычислительной техники (поэтому ему можно поручить вопросы, связанные с покупкой и настройкой вычислительной техники). Секретарю можно поручить решение задач связанных с рекламой, директор возьмет на себя задачи покупки оборудования, разработки рекламных материалов и методического обеспечения.

Распределение ресурсов

Чтобы приступить к распределению ресурсов, надо активизировать режим отображения диаграммы Ганта – выбрать в меню **Вид** команду **Диаграмма Ганта**. Затем надо выбрать задачу (щелкнуть на ее номере) и щелкнуть на командной

кнопке **Назначить ресурсы**. 

Затем в появившемся окне **Назначение ресурсов** (рис. 6) надо выбрать ресурс, в поле Единицы ввести долю рабочего времени, который ресурс будет тратить на решение задачи, и щелкнуть сначала на кнопке **Назначить**, затем – на кнопке **Закрыть**. Следует обратить внимание, что ресурсы следует назначать только элементарным задачам (задачам нижнего уровня).

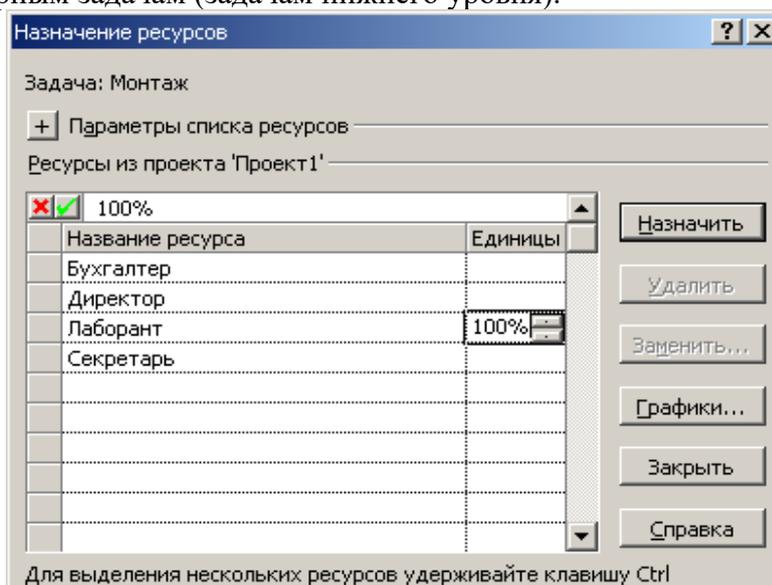


Рис. 6. Назначение ресурса на решение задачи

Сохранение проекта

Чтобы сохранить проект надо:

1. В меню **Файл** выбрать команду **Сохранить**.
2. В открывшемся окне **Сохранение документа** выбрать папку, в которой надо сохранить проект.
3. В поле **Имя файла** ввести имя файла проекта.
4. Щелкнуть на кнопке **Сохранить**.

Базовый план

После того как график реализации будет готов (согласован с участниками проекта и утвержден руководством), необходимо сохранить его как базовый план.

Для этого в меню **Сервис** надо выбрать команду **Отслеживание»Сохранить базовый план**.

С этого момента все изменения будут фиксироваться в рабочем плане, а базовый план будет оставаться неизменным. Это дает возможность сравнивать текущее состояние проекта с запланированным.

Печать графика реализации проекта

График реализации проекта можно распечатать. Перед тем как приступить непосредственно к печати графика реализации проекта можно скрыть ненужные столбцы таблицы задач проекта (как правило, на графике реализации проекта выводятся названия задач и даты начала их выполнения). Чтобы это сделать, надо установить курсор мыши на заголовок столбца, сделать щелчок правой кнопки и из контекстного меню выбрать команду **Скрыть столбец**.

Если время реализации проекта значительно, то график будет напечатан на нескольких листах, что не всегда удобно. Чтобы уменьшить количество листов, на которых будет выведен график, надо изменить масштаб его отображения. Наиболее просто это можно сделать при помощи командных кнопок **Увеличить** и **Уменьшить**,



Увеличить Уменьшить

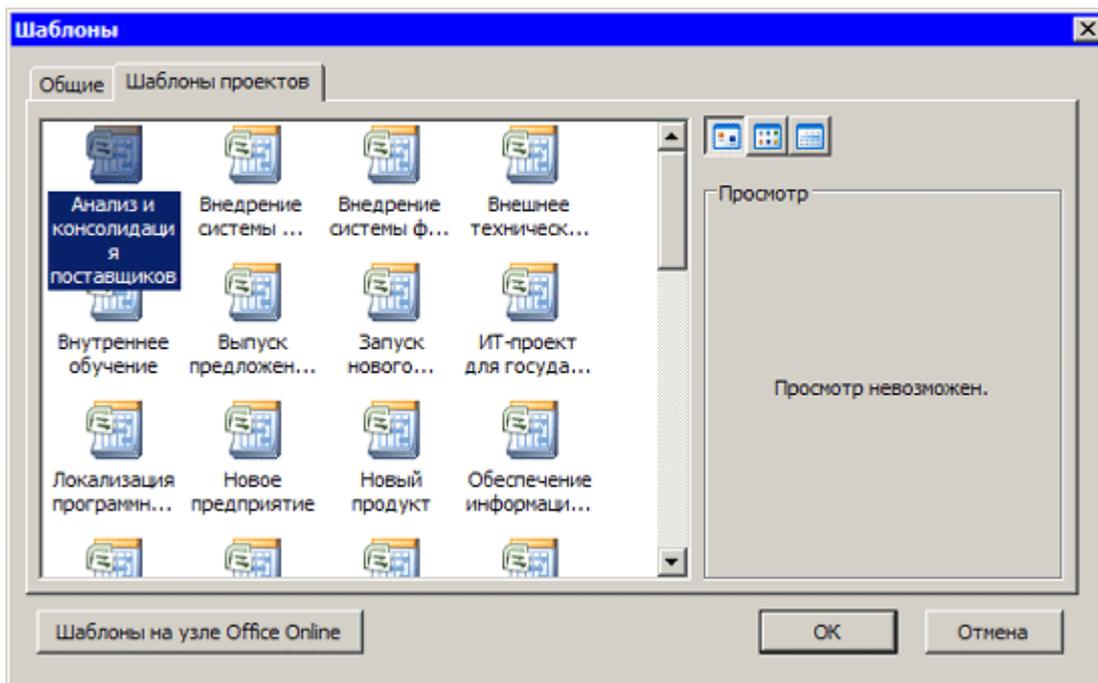
первая из которых увеличивает масштаб отображения графика, вторая – уменьшает.

Чтобы напечатать график реализации проекта, выберите из меню **Файл** команду **Печать**. В результате этого действия на экране появляется диалоговое окно **Печать**, в котором можно задать количество печатаемых копий, активизировать режим настройки принтера и режим просмотра результата печати (кнопка **Просмотр**). Печать активизируется в результате щелчка на кнопке **ОК**.

Практическая работа №2

1. Создание нового проекта на основе одного из имеющихся в программе шаблонов

Создавать новый проект целесообразнее всего на основе одного из имеющихся в программе шаблонов. Для этого в самой программе необходимо нажать в меню «Файл» на вкладку «Создать» и выбрать из имеющихся, например на компьютере. Просмотрев список доступных шаблонов, вы сможете получить представление о том, какие задачи удобно решать средствами этой программы. Так, есть шаблоны для создания таких проектов, как разработка программного обеспечения, строительство торгового комплекса, подготовка годового отчета, расширение отдела обслуживания клиентов и т.д. Если среди более чем четырех десятков шаблонов не нашлось подходящего, вы можете выбрать тот, который больше всего подходит по смыслу, или же создать проект "с чистого листа".



Создав документ на основе шаблона, вы увидите множество разнообразных полей, которые уже заполнены. Рассмотрим в качестве примера проект переезда компании в новый офис. Если компания достаточно велика, то на это может уйти несколько месяцев. Проект можно разделить на несколько этапов и включить в каждый из них те задания, которые необходимо выполнить. Например, такими этапами могут быть: от двух до шести месяцев до дня переезда, один-два месяца до переезда, две-четыре недели до переезда, от одного дня до двух недель, день переезда и день после переезда.

1	☐ Переезд офиса	127 дней	Пн 01.01.07	Вт 26.06.07
2	☑ От двух до шести месяцев до дня переезда	77 дней	Пн 01.01.07	Вт 17.04.07
19	☑ Один-два месяца до переезда	28 дней	Ср 18.04.07	Пт 25.05.07
43	☑ Две-четыре недели до переезда	10 дней	Пн 28.05.07	Пт 08.06.07
62	☑ От одного дня до двух недель до переезда	10 дней	Пн 11.06.07	Пт 22.06.07
102	☑ День переезда	1 день	Пн 25.06.07	Пн 25.06.07
108	☑ После переезда	1 день	Вт 26.06.07	Вт 26.06.07

Для каждого этапа определяется набор задач, например, за две-четыре недели до переезда необходимо заказать систему безопасности, подключить международную телефонную связь, избавиться от старой офисной мебели и т.д.

43	☑ Две-четыре недели до переезда	10 дней	Пн 28.05.07	Пт 08.06.07	
44	Определение стоимости активов, перемещаемых на нов	3 дней	Пн 28.05.07	Ср 30.05.07	42 Бухгалтер
45	Ознакомление с расценками на страхование нового офис	6 дней	Чт 31.05.07	Чт 07.06.07	44 Руководитель офиса
46	Страхование нового офиса	1 день	Пт 08.06.07	Пт 08.06.07	45 Руководитель офиса
47	Заказ системы безопасности	1 день	Пн 28.05.07	Пн 28.05.07	42 Руководитель офиса
48	Заказ ключей, карточек доступа	1 день	Пн 28.05.07	Пн 28.05.07	42 Руководитель офиса
49	Заключение договора со службой питания	1 день	Пн 28.05.07	Пн 28.05.07	42 Руководитель офиса
50	Отправка уведомления о смене адреса всем поставщик	1 день	Пн 28.05.07	Пн 28.05.07	42 Администратор почтовой системы
51	Отправка уведомления о смене адреса всем заказчикам	1 день	Пн 28.05.07	Пн 28.05.07	42 Администратор почтовой системы
52	Отправка уведомления о смене адреса всем абонентам	1 день	Пн 28.05.07	Пн 28.05.07	42 Администратор почтовой системы
53	Уведомление всех арендодателей о перемещении аренд	1 день	Пн 28.05.07	Пн 28.05.07	42 Руководитель офиса
54	Заказ чеков с новым адресом	1 день	Пн 28.05.07	Пн 28.05.07	42 Руководитель офиса
55	Подключение международной телефонной связи	1 день	Пн 28.05.07	Пн 28.05.07	42 Администратор систем связи
56	Перевод бесплатных телефонных номеров или заказ нов	1 день	Пн 28.05.07	Пн 28.05.07	42 Администратор систем связи
57	Планирование избавления от старого офисного оборудо	3 дней	Пн 28.05.07	Ср 30.05.07	42 Руководитель офиса
58	Избавление от старого офисного оборудования	1 день	Пн 28.05.07	Пн 28.05.07	42 Руководитель офиса
59	Планирование избавления от старой офисной мебели	3 дней	Пн 28.05.07	Ср 30.05.07	42 Руководитель офиса
60	Избавление от старой офисной мебели	1 день	Пн 28.05.07	Пн 28.05.07	42 Руководитель офиса
61	Окончание третьего этапа	0 дней	Пт 08.06.07	Пт 08.06.07	44,45,46

Для каждой задачи указывается, сколько дней понадобится на ее решение, какая задача обязательно должна быть выполнена перед ней, выбирается человек, ответственный за выполнение. Очень удобно, что если время, требуемое на выполнение задачи, будет уменьшено или увеличено, то и общее время работы над проектом сразу же изменится соответствующим образом.

Щелкнув по задаче, можно вызвать окно, в котором при необходимости указываются ее расширенные настройки. Например, если пока что точно неизвестно, сколько времени займет выполнение задачи, то можно установить флажок "Предварительная оценка". Тут же указывается приоритет задачи и процент ее выполнения.

Для некоторых задач важнее всего сроки исполнения, для других сроки не так важны, зато обязательно нужно вписаться в определенный бюджет. Все эти нюансы могут быть учтены при настройке параметров каждой задачи. При необходимости для задач можно устанавливать ограничения по срокам, выбирая фиксированную дату начала или окончания или же указывая дату, не раньше или не позже которой нужно начать или завершить работу над ней. На вкладке "Предшественники" можно просмотреть список задач, которые должны быть выполнены для того, чтобы можно было перейти к данной задаче. Если выполнение предварительных задач задерживается, это можно увидеть тут.

Ид.	Название задачи	Тип	Запаздывание
64	Прокладка телефонных линий	Окончание-начало (ОН)	0д
65	Установка телефонной системы	Окончание-начало (ОН)	0д
66	Назначение новых номеров телефонов	Окончание-начало (ОН)	0д
67	Заказ новых почтовых бланков и конвертов	Окончание-начало (ОН)	0д
68	Заказ новых визитных карточек	Окончание-начало (ОН)	0д
69	Заказ новых бланков	Окончание-начало (ОН)	0д
70	Планирование отключения коммунальных услуг	Окончание-начало (ОН)	0д
71	Ремонт нового офиса	Окончание-начало (ОН)	0д
72	Уборка нового офиса	Окончание-начало (ОН)	0д
73	Определение ближайших абонентских ящиков	Окончание-начало (ОН)	0д
74	Инвентаризация имеющейся офисной мебели (общая)	Окончание-начало (ОН)	0д

На вкладке "Ресурсы" можно указать тех сотрудников, которые будут работать над выполнением задачи. Во многих случаях удобно снабжать задачи заметками, которые могут помочь в их выполнении. Например, если стоит задача по избавлению от устаревшей офисной техники, в заметках можно указать, что ее можно решить, проведя аукцион среди сотрудников компании.

57			я от старого офисного оборудо
58		Заметки: 'Организация может провести аукцион для сотрудников.'	фисного оборудования
59			я от старой офисной мебели

Интересно, что если для задачи не указано время, требуемое на ее выполнение, программное обеспечение автоматически пометит ее как веху. Вехой, например, может быть задача под названием "Окончание второго этапа". В категории "Предшественники" для такой задачи можно указать те задачи, которые необходимо выполнить в рамках второго этапа, сама же эта задача станет предшественником для всех задач третьего этапа, к которому нужно будет перейти после завершения второго. При необходимости как веху можно пометить любую задачу, а не только ту, для которой указана нулевая продолжительность.

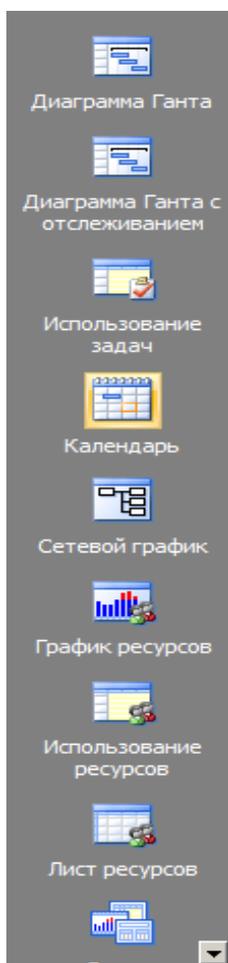
Для оценки продолжительности выполнения проекта пользователю не нужно строить формулы, так как в программе есть календарь, на котором можно отмечать даты выполнения задач. Работа с календарем очень удобна: например, если сроки сдвинулись, можно просто захватить мышью ту или иную задачу и передвинуть ее на другую дату. На календаре также можно устанавливать связи между задачами. Существует три основных типа связей: когда задачи должны выполняться параллельно и полностью совпадать по времени, когда время их выполнения может частично совпадать и когда вторая должна начинаться лишь тогда, когда будет завершена первая.

Упражнение № 3. Меню «Вид»

В программном обеспечении предусмотрено множество вариантов просмотра информации о проекте. Для выбора нужного представления можно использовать меню "Вид". Если вы часто переключаетесь между несколькими режимами просмотра, то удобно отобразить панель представлений, которая появится в левой части экрана.

По умолчанию в программе используется представление данных в виде "Диаграммы Ганта". В нем можно увидеть начальные и конечные даты выполнения всех задач, продолжительность, которая определена для каждой из них, ответственных за выполнение.

В режиме "Сетевой график" этапы проекта отображаются в виде элементов диаграммы, которые связаны между собой стрелками. В этом режиме удобно отслеживать зависимости между этапами проекта.

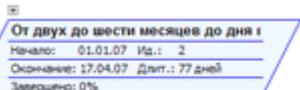


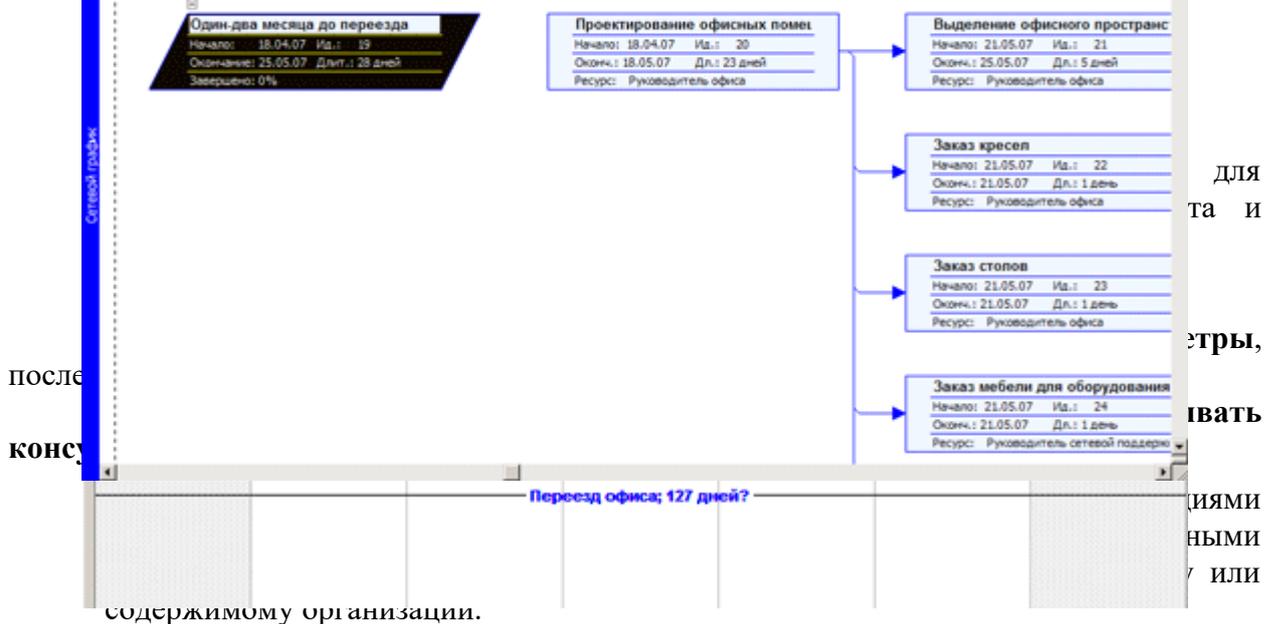
Режим "Диаграмма Ганта с отслеживанием" позволяет работать с таблицей и визуальными данными одновременно. В этом режиме те задачи, выполнение которых наиболее важно, выделены красным, что сразу позволяет оценить, что наиболее важно в данный момент. Менее важные задачи выделяются синим.

В программном обеспечении есть также несколько режимов, в которых удобно работать с ресурсами, необходимыми для выполнения проекта. Так, можно быстро посмотреть, какой будет зарплата веб-мастера, сколько рабочих дней уйдет у бухгалтера на подготовку документов, на какой срок необходимо нанимать внештатного программиста и т.д.

Упражнение № 4. Использование консультанта для начала нового проекта

Для помощи в создании нового проекта в программном обеспечении включен консультант, содержащий простые для





Для того чтобы развернуть собственный консультант в организации, необходимо изменить консультант в глобальном корпоративном шаблоне.

1. Для этого в меню **Сервис** выберите пункт **Параметры**, после чего перейдите на вкладку **Интерфейс**.
2. В группе **Страница функциональных возможностей и макета консультанта** раздела **Параметры консультанта** щелкните **Страница пользователя**, а затем в поле **URL** введите адрес страницы. Кроме того, чтобы найти нужную страницу в диалоговом окне **Открыть** можно нажать кнопку **Обзор**.
3. В группе **Содержимое консультанта** щелкните **Использовать пользовательское содержимое**, а затем введите имя XML-файла в поле **XML-файл пользовательского содержимого**. Кроме того, чтобы перейти к нужному файлу можно нажать кнопку **Обзор**.

Создание нового проекта с помощью консультанта.

1. После создания нового проекта на панели инструментов **Консультант** выберите **Задачи**, а затем щелкните **Определение проекта**.
2. Чтобы ввести сведения о проекте, включающие дату начала проекта и параметры совместной работы, выполните инструкции консультанта.
3. На панели инструментов **Консультант** выберите **Задачи**, а затем щелкните **Определение рабочего времени проекта**.
4. Чтобы ввести рабочее время для проекта, следуйте инструкциям на панели инструментов «Консультант».

Практическая работа №3

Составление расписаний выполнения работ

При запуске программное обеспечение выглядит так, как

на рисунке 1.

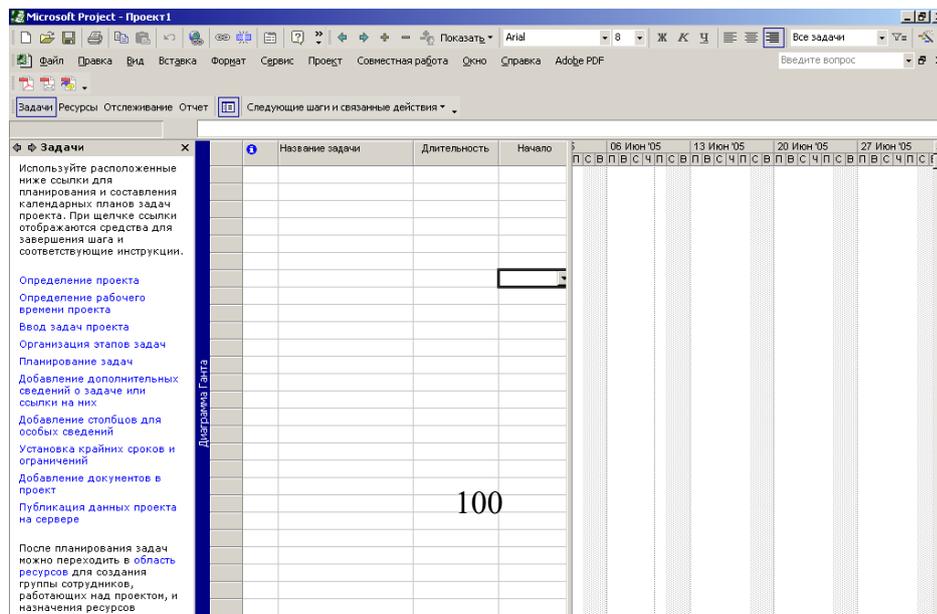


Рисунок 1. Программное обеспечение после первого запуска

Сверху расположена строка меню, под ней – панель инструментов, ниже – рабочая область программы.

Среди панелей инструментов есть особая панель *Консультант (Project Guide)*, которая расположена над рабочей областью программы. Названия кнопок этой панели соответствуют основным объектам, с которыми может работать: *Задачи (Tasks)*, *Ресурсы (Resources)*, *Отслеживание (Track)*, *Отчет (Report)*. При щелчке по любой из этих кнопок на панели в левой части рабочей области, которая называется *Областью задач (Task Pane)* отображается список возможных действий с выбранным объектом.

Упражнение № 1. Настройка параметров программы

Перед началом работы необходимо настроить некоторые параметры программы. Для перехода к настройке нужно выбрать команду *Сервис → Параметры (Tools → Options)*. На экране появится диалоговое окно настроек с несколькими закладками, на которых сгруппированы параметры, определяющие работу программы. Почти на каждой закладке есть кнопка *По умолчанию (Set as default)*, позволяющая сохранить сделанные настройки так, чтобы они автоматически применялись во всех последующих проектах.

На рисунке 2 представлена закладка *Вид (View)*.

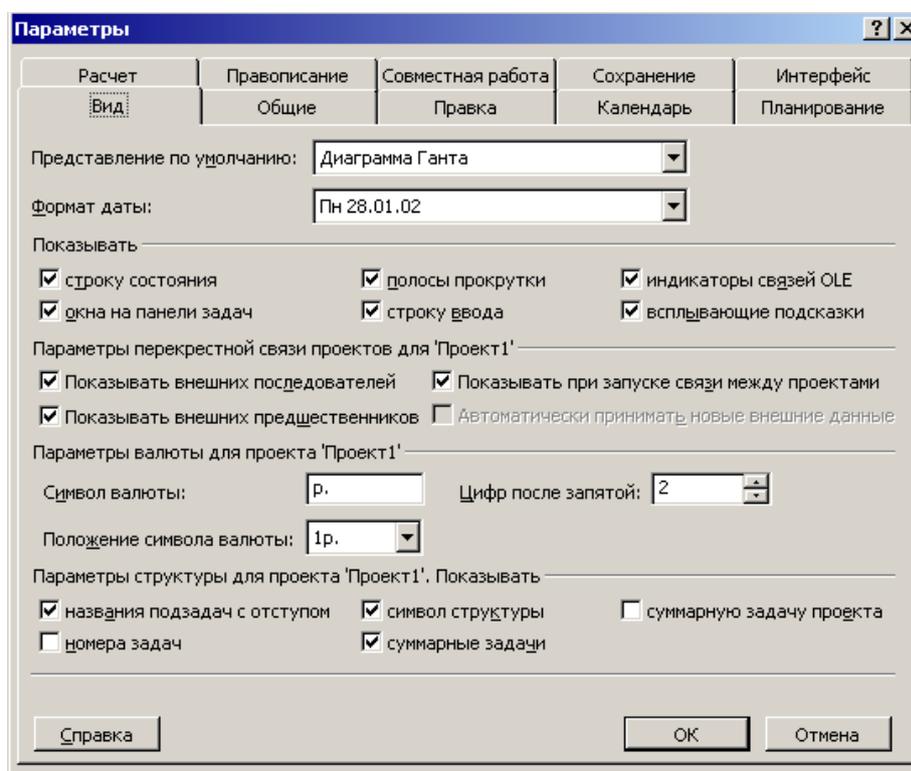


Рисунок
Закладка
окна

«Параметры»

В списке *Представление по умолчанию (Default view)* указывается, какое представление открываться при создании нового проекта или при запуске программы. Представление – это способ отображения проектных данных для

просмотра и редактирования, который может содержать в себе таблицы, диаграммы и формы. В представлении на экране отображается ограниченный набор нужной информации о проекте, что облегчает ее просмотр и редактирование.

С помощью списка *Формат даты (Date format)* можно установить формат представления дат.

В разделе *Показывать (Show)* расположены флажки, управляющие отображением стандартных и нестандартных элементов интерфейса программы.

Закладка *Общие (General)* (рисунок 3) состоит из трех разделов. В разделе *Общие параметры* расположены элементы настройки общих параметров программы, в третьем *Общие параметры для...(General options for...)* – элементы настройки общих параметров открытого проекта. В разделе *Мастер планирования (Planning Wizard)* определяется, каким образом будут отображаться подсказки при составлении плана проекта.

Прежде чем начать составление плана проекта необходимо ввести в диалоговом окне начальные данные о проекте. Флажок *Запрос на ввод сведений о проекте для новых проектов (Prompt for project info for new projects)* обеспечивает автоматический вывод на экран необходимого диалогового окна при создании нового проекта.

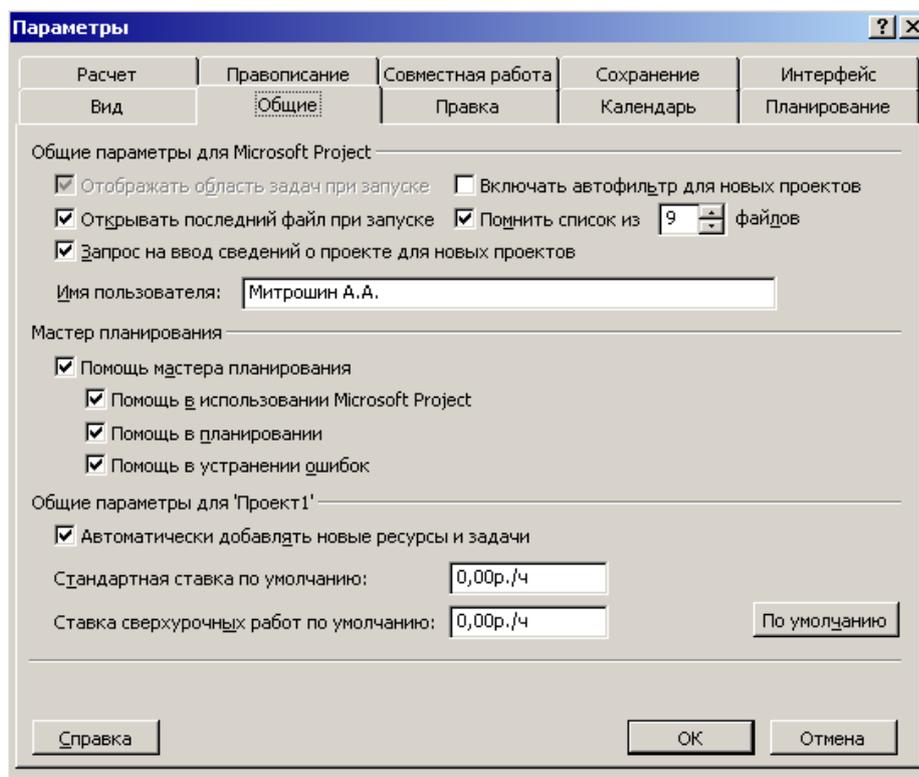


Рисунок «Общие»

3. Закладка окна

«Параметры»

В программном обеспечении имеется особый режим просмотра таблиц – Автофильтр (AutoFilter), при котором в заголовке каждой таблицы отображается кнопка для ее сортировки. Поскольку этот режим требуется не всегда и его можно включить кнопкой *Форматирование (Formatting)* панели инструментов, флажок *Включить автофильтр для новых проектов (Set AutoFilter for new projects)* удобнее сбросить.

В последнем блоке настроек нужно установить флажок *Автоматически добавлять новые ресурсы и задачи (Automatically add new resources and tasks)*. Это обеспечит удобный ввод данных в проект.

Настройки редактирования помещены в закладке *Правка(Edit)* (рисунок 4) и сгруппированы в двух разделах (вверху закладки), один из которых содержит элементы настройки, относящиеся к программе целиком, а второй (внизу закладки) к открытому в данный момент проекту.

Флажок *Перетаскивание ячеек (Allow cell drag and drop)* определяет, будет ли можно перетаскивать ячейки таблиц с помощью мыши, а флажок *Переход к следующему полю после ввода (Move selection after enter)* – будет ли перемещаться курсор в следующую ячейку, после того как при редактировании текущей ячейки нажата клавиша Enter. Флажок *Правка прямо в ячейке(Edit directly to cell)* определяет, можно ли редактировать данные непосредственно в ячейке таблицы. Если этот флажок сбросить, то для редактирования данных будет необходимо выделить ячейку и вводить ее содержимое в строку ввода.

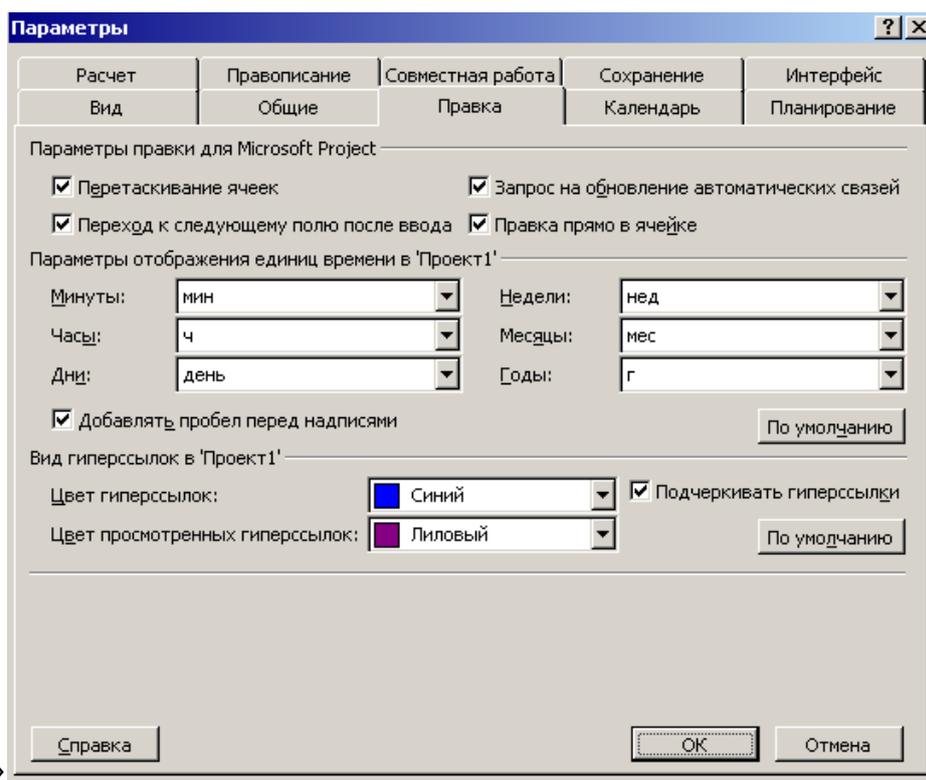


Рисунок 4.
«Правка»
«Параметры»

Закладка
окна

Флажок *Запрос на обновление автоматических связей (Ask to update automatic links)* управляет способом обновления объектов из других файлов, внедренных в файл проекта. Обновление может происходить автоматически или по запросу. Для автоматического обновления внедренных объектов флажок необходимо сбросить.

От состояния флажка *Добавлять пробел перед надписями (Add space before label)* зависит, будет ли вставляться пробел между количеством единиц и их обозначением. По умолчанию флажок установлен, и сбрасывать его не желательно.

Закладка *Календарь* (рисунок 5) используется для ввода, просмотра и изменения параметров даты и времени. Все эти параметры являются локальными и сохраняются вместе с текущим проектом. Имеются следующие возможности.

- Задание первого дня недели и первого месяца финансового года для данного проекта.
- Задание времени начала и окончания по умолчанию для тех ограничений задач, в которых вводится дата, но не вводится время.
- Задание количества часов в дне или неделе, а также количества дней в месяце для ввода значений длительности и трудозатрат.

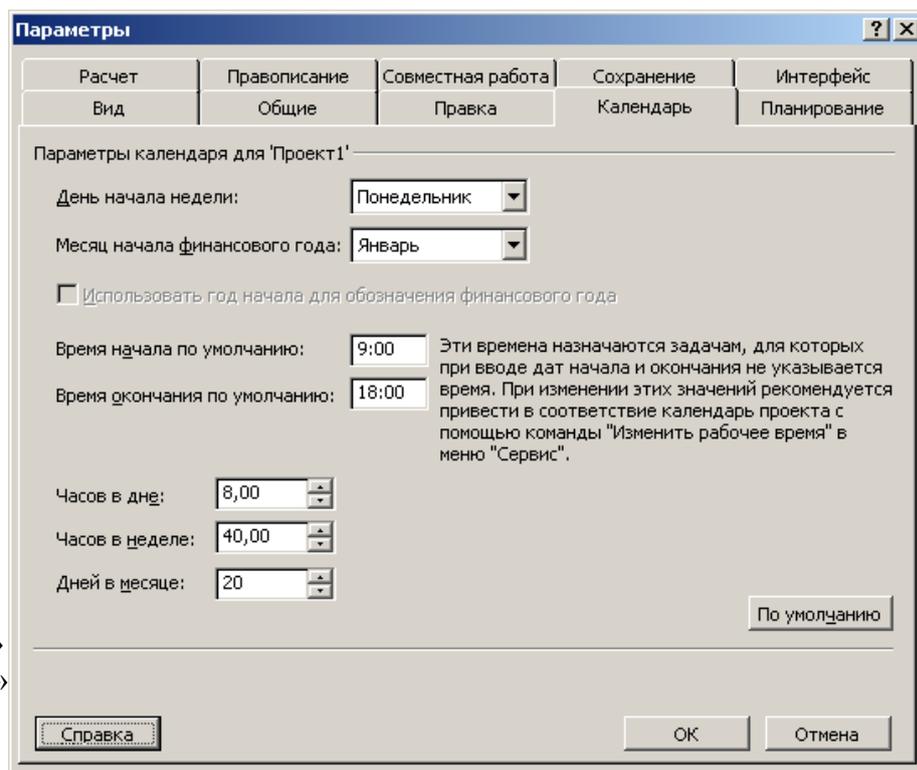


Рисунок 5.
«Календарь»
«Параметры»

Закладка
окна

Изменение параметров данного диалогового окна не влияет ни на календарь проекта, ни на календари рабочего времени ресурсов. Оно влияет только на преобразование длительностей в соответствующие временные интервалы. Заданный на этой вкладке временной интервал применяется также к преобразованию длительности в значения трудозатрат. Подобным же образом осуществляется преобразование временных значений полей Часов в неделю и Дней в месяце.

Закладка *Планирование* (рисунок 6) используется для ввода, просмотра и изменения параметров планирования задач. Возможны как установка глобальных параметров планирования для программы в целом, так и установка локальных параметров планирования для текущего проекта, включая даты начала новых задач, единицы измерения времени, используемые для длительности и трудозатрат, и стандартный тип задачи.

Длительность вводится в - этот параметр определяет единицу измерения времени, используемую по умолчанию в поле «Длительность». Программа будет вводить ее автоматически, если пользователь при вводе значения в поле «Длительность» не укажет единицу измерения времени. Если в поле «Длительность» ввести другую единицу измерения времени, программа будет автоматически вводить новую единицу, если значение времени будет введено без указания единицы измерения.

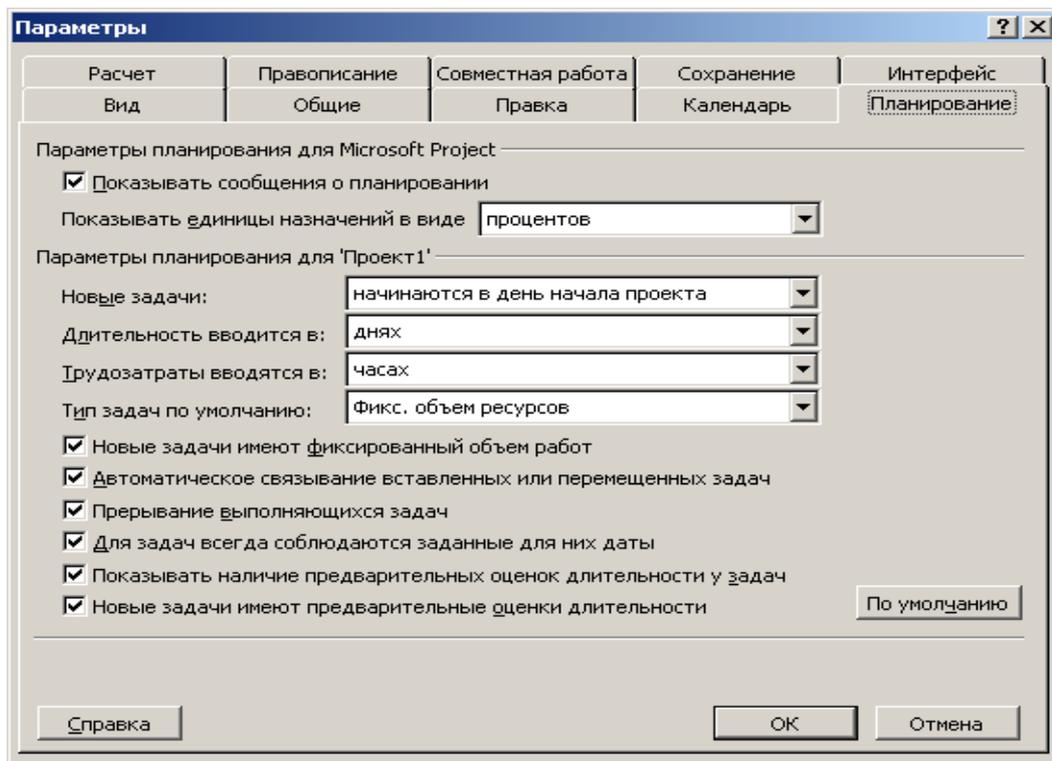


Рисунок 6. Закладка «Планирование» окна «Параметры»

Тип задач по умолчанию - определяет влияние изменения существующих сведений о назначениях — трудозатрат, единиц назначения или длительности — на расчет двух других полей назначений для данной задачи. По умолчанию он задает тип для всех новых задач.

Существуют следующие типы задач.

Фиксированная длительность. Этот тип задачи выбирается в случае, если требуется сохранить постоянной длительность задачи независимо от любых изменений единиц назначения или трудозатрат. В задачах с фиксированной длительностью выполняются следующие действия.

- При изменении единиц заново вычисляет трудозатраты.
- При изменении длительности в задаче с фиксированной длительностью заново вычисляются трудозатраты.
- При изменении количества трудозатрат заново вычисляются единицы.

Фиксированные единицы. Тип задачи Фиксированные единицы выбирается в случае, если требуется сохранить постоянным число единиц назначения независимо от каких-либо изменений длительности или трудозатрат. Этот параметр применяется по умолчанию. В задачах с фиксированными единицами выполняются следующие действия.

- При изменении единиц в задаче с фиксированными единицами заново вычисляется длительность.

- При изменении длительности заново вычисляются трудозатраты.

- При изменении количества трудозатрат заново вычисляется длительность.

Фиксированные трудозатраты. Этот тип задачи выбирается в случае, если требуется сохранить постоянным количество трудозатрат независимо от каких-либо изменений длительности или единиц назначения ресурсов для задачи. Поскольку задачи с фиксированными трудозатратами по определению являются задачами с фиксированным объемом работ, флажок *Фиксированный объем работ* устанавливается для них автоматически. В задачах с фиксированными трудозатратами выполняются следующие действия:

- при изменении единиц заново вычисляется длительность;

- при изменении длительности заново вычисляются единицы;
- при изменении количества трудозатрат для задач с фиксированными трудозатратами производится перерасчет длительности.

Флажок *Показывать сообщения о планировании (Show scheduling messages)* определяет, будет ли программное обеспечение сообщать о несоответствиях в плане проекта.

Упражнение № 2. Таблицы

Таблицы являются основным средством хранения данных.

В проектном файле все данные хранятся в виде двух таблиц, одна из которых содержит информацию о задачах, а вторая – о ресурсах проекта, то есть задействованных при выполнении проекта людях и материальных ценностях. Две этих внутренних таблицы состоят из множества полей, большинство из которых не используется. Отображаются только столбцы, включенные в определенные представления.

В программном обеспечении заложен набор предопределенных таблиц, каждая из которых содержит несколько полей одной из внутренних таблиц проектного файла. Списки предопределенных таблиц с информацией о задачах и ресурсах приведены в таблицах 1 и 2.

Таблица 1. Таблицы с информацией о задачах

<i>Таблица</i>	<i>Содержание</i>
Исходный план (Baseline)	Данные из базового плана проекта
Даты ограничений (Constraint Dates)	Ограничения задачи (даты ограничений и типы)
Затраты (Cost)	Стоимость задач и проекта
Задержка (Delay)	Информация для выравнивания загрузки ресурсов
Освоенный объем (Earned Value)	Общая таблица для сравнения запланированного и фактического объемов работ, запланированной и фактической стоимости
Индикаторы календарного плана освоенного объема (Earned Value Schedule Indicators)	Сравнение запланированного и фактического графиков выполнения проекта
Показатели затрат для освоенного объема (Earned Value Cost Indicators)	Сравнение запланированного и фактического проектного бюджета
Ввод (Enter)	Таблица для ввода общей информации о задаче
Экспорт (Export)	Информация для экспорта данных о задачах во внешний файл
Гиперссылка (Hyperlink)	Связанные с задачей ссылки, ведущие на внешние сайты
Ожидаемый сценарий (PA_Expected Case) Оптимистичный сценарий (PA_Optimistic Case) Пессимистичный сценарий (PA_Pessimistic Case)	Три таблицы для анализа проекта по методике PERT. Идентичны по структуре и содержат данные об ожидаемом, оптимистичном и пессимистичном планах проекта
Ввод для анализа PERT (PA_PERT Entry Sheet)	Таблица для ввода данных, используемых при проведении анализа плана по методике PERT
Сводные задачи (Rollup Table)	Служит для оптимизации отображения сводных задач
Schedule (календарный план)	Расписание начала и окончания выполнения задач, интервалы между задачами
Суммарные (Summary)	Общая информация о задачах проекта
Отслеживание (Tracking)	Информация о ходе выполнения проекта
Использование (Usage)	Задачи проекта, объемы работ, длительность, даты начала и

<i>Таблица</i>	<i>Содержание</i>
	окончания
Расхождение (Variance)	Данные об отклонении от плана при выполнении работ
Трудозатраты (Work)	Информация об объеме работы по проекту

Таблица. 2. Таблицы с информацией о ресурсах

<i>Таблица</i>	<i>Содержание</i>
Стоимость (Cost)	Стоимость ресурсов проекта
Приобретенная стоимость (Earned Value)	Сравнение запланированной и фактической стоимостей ресурса в проекте
Ввод (Entry)	Таблица ввода общей информации о ресурсе
Ввод материального ресурса (Entry – Material Resources)	Таблица для ввода общей информации о материальном ресурсе
Ввод нематериального ресурса (Entry – Work Resources)	Таблица для ввода общей информации о нематериальном ресурсе
Экспорт (Export)	Информация для экспорта данных о ресурсах во внешний файл
Гиперссылка (Hyperlink)	Связанные с ресурсом ссылки, ведущие на внешние сайты в Internet и Intranet
Сводная (Summary)	Общая информация о ресурсах проекта
Использование (Usage)	Информация о работе, на которую выделены ресурсы
Работа (Work)	Информация об объеме работы, на которую выделены ресурсы

Таблицы отображаются в *представлениях*, причем есть представления, в которых таблицы совмещены с диаграммой, например диаграммой Ганта (Gantt Chart). Существуют и представления, состоящие только из таблицы. Какая из таблиц загружается по умолчанию при загрузке программы, определяется в настройках.

Переключение между таблицами осуществляется с помощью команды Вид→Таблица (View→Table). В пункте меню Вид→Таблица перечислены наиболее часто используемые таблицы для текущего представления. Если в этом пункте меню нет нужной таблицы, то ее можно найти в пункте *Другие таблицы (More tables)*.

Упражнение № 3. Подготовка к составлению плана

Управление проектом заключается в составлении плана и отслеживании хода работ по нему. Проекты могут осуществляться в любой области деятельности. Так проектом может быть разработка информационной системы, выполнение курсового или дипломного проекта, постройка здания, проведение предвыборной компании и т.д.

Проект предпринимается для достижения определенного результата в определенные сроки и за определенные деньги. План проекта составляется для того, чтобы определить, с помощью каких работ будет достигаться результат проекта, какие люди и оборудование будут нужны для выполнения работ, в какое время эти люди и оборудование будут заняты работой по проекту. Поэтому проектный план содержит три основных элемента: *задачи (tasks)*, *ресурсы (resource)* и *назначения (assignment)*.

Задачей называется работа, осуществляемая в рамках проекта для достижения определенного результата. Поскольку проект обычно содержит много

задач, то для удобства отслеживания плана их объединяют в группы, или *фазы*. Совокупность фаз проекта называется его жизненным циклом.

Фаза проекта состоит из одной или нескольких задач, в результате выполнения которых достигается один или несколько основных результатов проекта. Если для достижения результатов задачи нужно выполнить только ее, то для достижения результата фазы нужно выполнить группу других задач. В этом заключается отличие фазы от задачи – ее результат суммирует результаты других задач. Поэтому фазы называются *суммарными задачами* (summary task). Фазы могут состоять как из задач, так и других фаз.

Проект разбивается на фазы для удобства контроля хода работы. По завершении проектной фазы обычно осуществляется анализ полученных результатов, чтобы с минимальными затратами определить и исправить ошибки.

Задачи, в результате выполнения которых достигаются промежуточные цели, называются *завершающими задачами*. Они называются *вехами* (milestone). Обычно результатом фазы является достижение некоторой промежуточной цели, поэтому вехой в плане проекта принято обозначать последнюю задачу фазы, в результате которой достигается ее результат. Если такой задачи нет, а фазовый результат достигается, например, одновременным завершением нескольких задач, то создается фиктивная завершающая задача, длительность которой устанавливается равной 0 дней, и на нее не выделяются исполнители. Она присутствует в плане исключительно для обозначения момента завершения фазы, что облегчает отслеживание плана проекта.

Длительность задачи - это период рабочего времени, необходимый для выполнения задачи. Длительность задачи может не соответствовать *трудозатратам* занимающегося задачей сотрудника. Длительность (duration) соответствует времени, через которое будет получен результат работы, а трудозатраты (work) – времени, затраченному сотрудниками на получение результата.

Задачи в плане проекта взаимосвязаны, например, часто одна задача не может начаться до тех пор, пока не будет закончена другая. На плане проекта *зависимости* (dependencies) обозначаются с помощью *связей* (links). Оба этих термина – *зависимость* и *связь* - используются с одним и тем же смыслом, обозначая логику, определяющую последовательность выполнения работ в плане проекта.

Под **ресурсами** понимаются сотрудники и оборудование, необходимые для выполнения проектных задач. Каждый сотрудник, участвующий в проекте, получает определенную *роль*, соответствующую его квалификации. При составлении списка ресурсов часто используется ролевое планирование. Например, сначала определяется, что для исполнения работ требуются три программиста и один менеджер, а затем, когда план проекта утвержден, подбираются конкретные сотрудники для этих ролей.

Важное свойство ресурсов – **стоимость** (cost) их использования в проекте. Существуют два типа стоимости ресурсов: *повременная ставка* (rate) и *стоимость за использование* (cost per use). Повременная ставка выражается в стоимости использования ресурса за единицу времени. Обычно почасовая ставка используется для учета стоимости не материальных ресурсов. Величина затрат на использование обозначает стоимость использования оборудования или сотрудника в задаче, которая не зависит от того, сколько времени задействован в задаче сотрудник или материальный ресурс. Общие затраты на использование такого ресурса определяются путем умножения стоимости использования на число задач, в которых он задействован. У ресурса может быть указана стоимость как одного из двух типов, так и обоих.

Назначения – это связь определенной задачи и ресурсов, необходимых для ее выполнения. На одну задачу может быть назначено несколько ресурсов, причем как материальных, так и нематериальных. Назначения объединяют в плане задачи и ресурсы, делая план целостным. Благодаря назначениям решается целый ряд задач планирования:

- определяются ответственные лица за исполнение задач;
- когда определены задачи, за которые отвечает ресурс, можно рассчитать общий объем времени, затрачиваемый им на проект, а значит и его стоимость;
- определив стоимость участия всех ресурсов в проекте, можно определить общую стоимость проекта;
- назначая ресурсы на задачи, можно сокращать срок выполнения работ, выделяя на них больше ресурсов, и тем самым, сокращая общую длительность проекта.

Большинство проектов имеют определенную дату окончания, бюджет и объем работ. Тройку «время», «деньги», «объем работ» часто называют **проектным треугольником** потому, что при внесении изменений в один из этих элементов, меняются оба другие. Хотя для проекта в равной степени важны все три элемента, один из них, как правило, имеет наибольшее влияние на другие в зависимости от выбранного приоритета. Например, если изменить план проекта, укоротив расписание, то либо возрастает стоимость проекта, либо уменьшается объем выполненных работ. Если изменить план проекта с целью уменьшения его бюджета, то может возрасти длительность выполнения проекта и уменьшиться объем работ. Если увеличить объем работ, то проект будет длиться дольше и стоить дороже. В общем случае изменения в плане зависят от специфики проекта. В некоторых случаях сокращение времени увеличивает стоимость, а в других – уменьшает ее.

Качество – четвертый элемент проектного треугольника. Изменения, вносимые в любую из сторон треугольника, практически всегда влияют на качество. Качество не является стороной треугольника – это результат действий со временем, стоимостью и объемом работ. Например, если существует лишнее время в расписании, то можно увеличить объем работ, добавив новые задачи и, возможно, увеличив длительность проекта. С этими дополнительными задачами и временем можно добиться более высокого качества выполнения проекта. С уменьшением объемов работ у проекта будет меньше шансов выйти на требуемый уровень качества, поэтому снижение расходов может привести к ухудшению качества проекта.

Упражнение № 4. Составление плана проекта

Составление плана проекта в общем виде заключается в описании задач проекта, доступных ресурсов и определении взаимосвязей между ними с помощью назначений. Однако, при составлении расписания работ, количество операций несколько увеличивается.

Планирование начинается с определения проекта, то есть описания его ключевых характеристик. Затем составляется список фаз и задач, а также список необходимых для их выполнения ресурсов. После этого в план вносятся дополнительная информация о задачах и ресурсах, которая будет использоваться при определении назначений и в дальнейшем при проведении работ по плану (**отслеживание работ**). Далее осуществляются назначения, после чего проект оптимизируется, если его длительность или бюджет окажутся больше ожидаемых.

Составление плана невозможно без задания ключевых параметров проекта, таких как его длительность, рабочее время и методика планирования.

Чтобы составить расписание (план) работ нужно создать файл нового проекта, щелкну по кнопке *Создать (New)* на панели инструментов. Если установлен соответствующий флажок в настройках программы, то откроется диалоговое окно *Сведения о проекте (Project Information)* (рисунок 7). Чтобы изменить параметры проекта в дальнейшем, это диалоговое окно можно вызвать командой *Проект→Сведения о проекте(Project→Project Information)*.

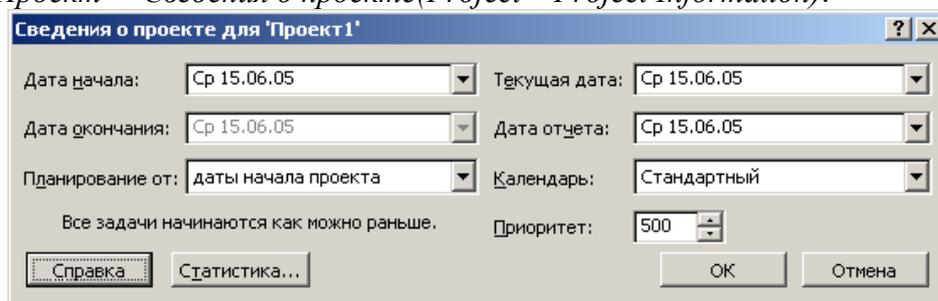


Рисунок 7. Окно сведений о проекте

Проект можно планировать двумя способами: от даты начала проекта или от даты окончания. Если у проекта нет жесткой даты окончания, то при планировании применяется первый способ: фиксируется дата, когда необходимо начать выполнение проекта, и в ходе составления плана определяется дата его завершения. Если же проект должен быть завершен к фиксированной дате, то используется противоположный способ: фиксируется дата окончания и в ходе составления плана определяется, когда выполнение проекта должно быть начато, чтобы все работы были выполнены в срок.

Способ планирования выбирается в раскрывающемся списке *Планирование от (Schedule from)*.

Даты начала и окончания проекта выбираются в списках *Дата начала (Start date)* или *Дата окончания (Finish date)*. Зафиксировать можно только одну из дат в соответствии с выбранным способом планирования.

Значения текущей даты и даты отчета выбираются в раскрывающихся списках *Текущая дата (Current date)* и *Дата отчета (Status date)*. По умолчанию текущая дата соответствует системной дате операционной системы, а дата отчета равна текущей дате и поэтому в окне сведений о проекте в качестве ее значения выбрано НД (NA). Любое из значений можно изменить независимо от другого. При отслеживании проекта и вводе сводной информации используется дата отчета, а не текущая дата.

По умолчанию программа считает, что все выполненные трудозатраты относятся ко времени до даты отчета, а все оставшиеся трудозатраты – ко времени после даты отчета. Если выполнение задачи запланировано после даты отчета, но вносятся данные о фактических трудозатратах по этой задаче, то программное обеспечение изменит выполненную часть задачи так, чтобы она закончилась к дате отчета, а выполнение оставшихся трудозатрат остается на будущее.

Чтобы определить рабочее время, в рамках которого будут выполняться работы в раскрывающемся списке *Календарь (Calendar)* нужно выбрать один из доступных календарей. Календарем называется набор параметров, определяющих перечень рабочих и нерабочих дней, а также рабочее время в каждом из рабочих дней. В стандартной поставке в этом списке присутствуют три пункта:

- 1) стандартный (Standard);
- 2) 24 часа (24 Hours);
- 3) ночная смена (Night shift).

Первый календарь соответствует стандартному расписанию с 8 часовым рабочим днем, второй – круглосуточному рабочему дню, а третий предполагает круглосуточный режим работы с перерывами.

Очень часто входящие в состав поставки календари не подходят для проекта. В этом случае можно изменить существующий календарь или создать новый. Существует возможность создавать как групповые (или базовые), так и личные календари. Последние отражают персональные расписания отдельных сотрудников. Поэтому при создании базового календаря в него следует вносить только настройки, общие для всех участников проекта или группы, к которой относится календарь. Специфические настройки заносятся в личный календарь каждого сотрудника.

В поле *Приоритет* указывается число в диапазоне от 0 до 1000, которое используется при выравнивании загрузки ресурсов между разными проектами. Чем больше число, тем выше приоритет проекта.

После нажатия кнопки *Ок* создается новый файл проекта.

Первым шагом в планировании является составление списка задач. Только после этого можно оценить сроки и стоимость выполнения каждой задачи.

Определение состава проектных работ начинается с определения этапов (или фаз) проекта. После того, как состав фаз и результаты определены, нужно определить последовательность фаз относительно друг друга и крайние сроки их выполнения. Затем нужно выяснить, из каких работ состоят фазы, в какой последовательности выполняются работы внутри фаз и в какие сроки необходимо уложиться при выполнении каждой работы.

Определять состав работ удобно поэтапно. Сначала создается *скелет плана работ*, состоящий из фаз, их результатов и нескольких основных задач. Затем в план добавляются остальные задачи, определяются их длительности и связи. Затем определяются ключевые даты проекта, которые включают крайние сроки достижения результатов проекта и некоторые другие ограничения по времени. Наконец, в план добавляется дополнительная информация о задачах.

Создадим новый проект с методикой планирования от даты начала. Используем стандартный календарь. В качестве даты проекта выберем предлагаемую по умолчанию.

План работ удобнее всего составлять в представлении *Диаграмма Ганта (Gantt Chart)*. Диаграмма Ганта [2] представляет собой хронограмму на полный набор работ, которая широко используется в настоящее время. Для добавления задачи в план проекта нужно установить курсор в таблицу слева от диаграммы и ввести название задачи в поле *Название задачи (Task name)*. После этого отрезок, символизирующий задачу, появится на диаграмме.

Добавление в план фазы не отличается от добавления задачи – любая задача автоматически становится фазой, как только у нее появляется вложенная задача, то есть задача, находящаяся на следующем уровне структуры плана. До тех пор пока у задачи нет вложенных задач, она не является фазой.

Чтобы поместить задачу на более низкий уровень структуры, нужно установить курсор в строку с задачей и на панели инструментов щелкнуть по кнопке *На уровень ниже* (\Rightarrow) в панели инструментов или нажать комбинацию клавиш $\text{Alt}+\text{Shift}+\rightarrow$. Для перемещения задачи на более высокий уровень структуры нужно щелкнуть на кнопке *На уровень выше* (\Leftarrow) панели инструментов или использовать комбинацию клавиш $\text{Alt}+\text{Shift}+\leftarrow$.

Пример создания скелетного плана приведен на рисунке 8.

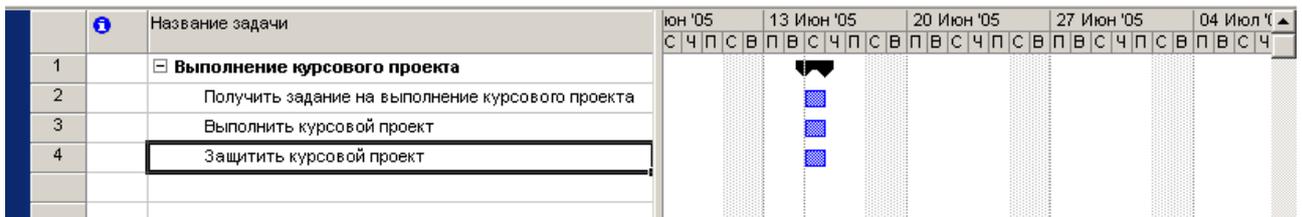


Рисунок 8. Пример создания скелетного плана

Из рисунка 8 видна разница графического отображения фазы «Выполнение курсового проекта» от задачи «Получит задание на выполнение курсового проекта».

Результаты фаз вводятся в виде завершающих задач, и эти задачи могут не обозначать реальной деятельности. Например, результатом фазы «Выполнение курсового проекта» является сданный курсовой проект. Для того, чтобы указать тот факт, что данная задача является завершающей, ее длительность устанавливается равной 0 (рисунок 9).

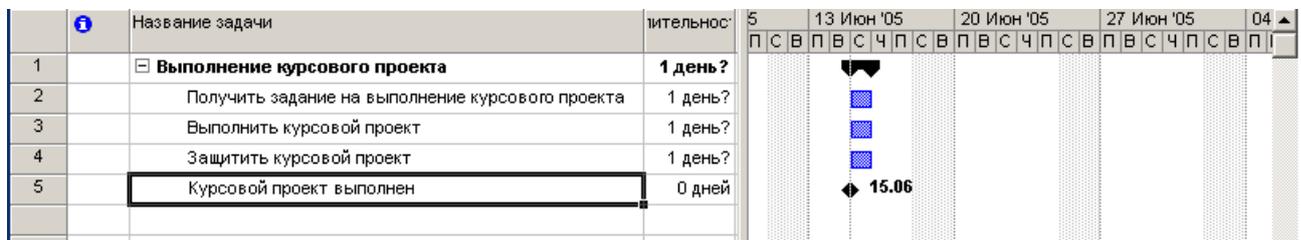


Рисунок 9. Добавление завершающей задачи

Добавим теперь завершающие работы к фазам «Получить задание на выполнение курсового проекта», «Выполнить курсовой проект», «Защитить курсовой проект». Сейчас это не фазы, но они станут фазами, если к этим работам добавить подчиненные работы (рисунок 10).

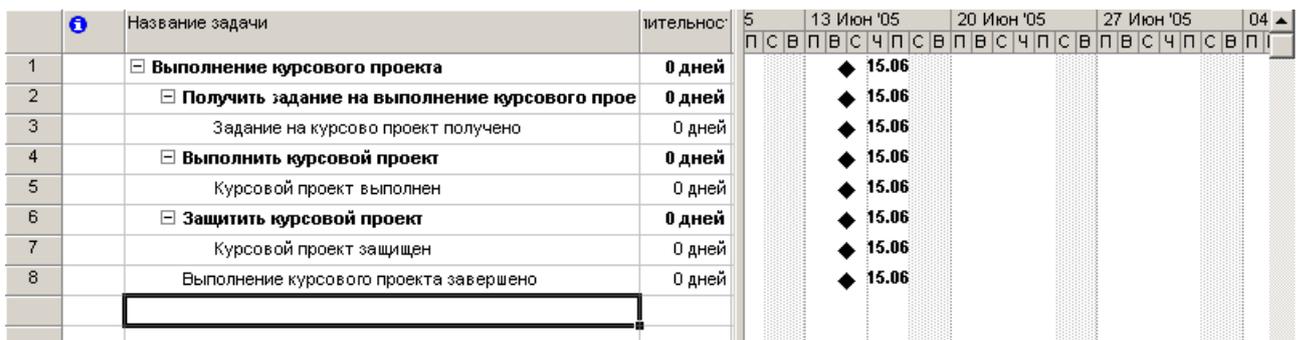


Рисунок 10. Добавление завершающей задачи к каждой фазе

Поскольку сейчас каждая фаза содержит только завершающую задачу, то есть задачу длительности 0, а длительность выполнения фазы определяется длительностями входящих в фаза задач, то длительность каждой фазы равно 0. Поэтому фазы помечены как завершающие задачи. Добавим в фазы задачи, решением которых достигается цель фазы (рисунок 11).

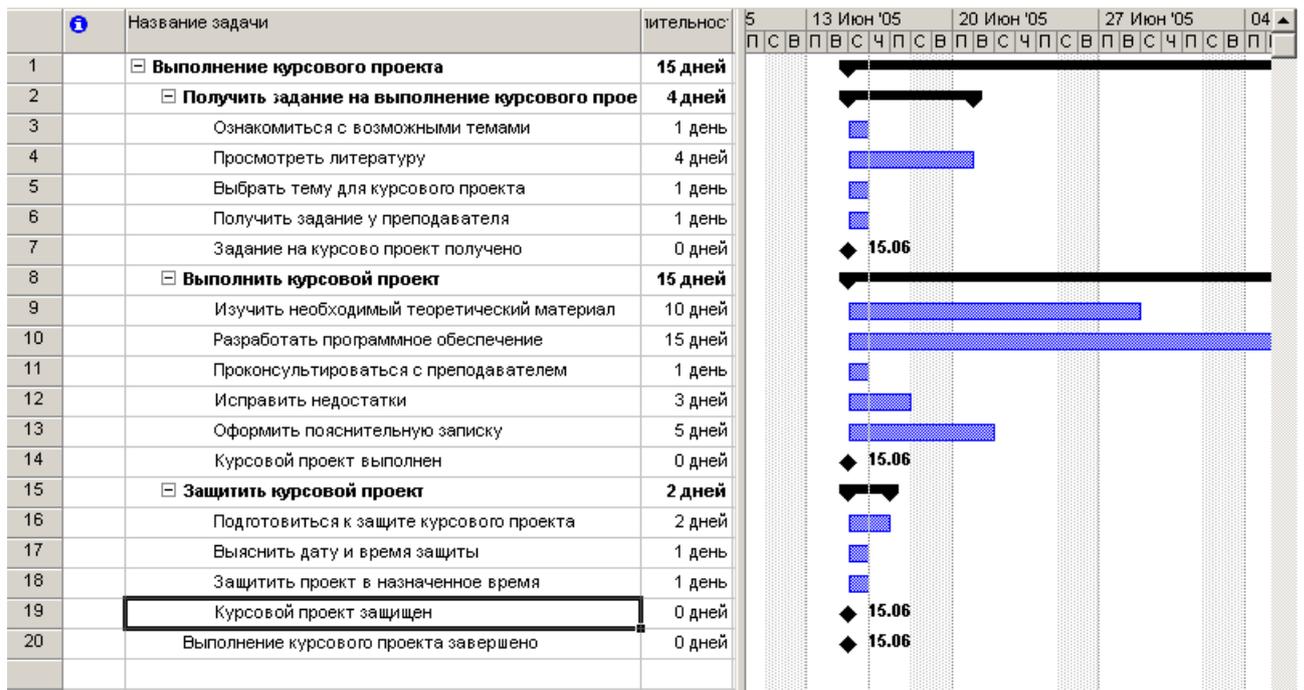


Рисунок 12. Задачи с добавленными длительностями

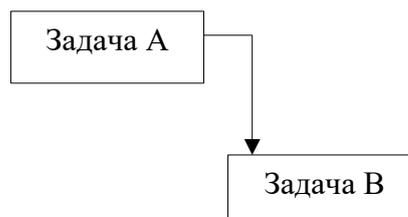
После определения длительностей задач, можно переходить к определению зависимостей между задачами. Например, разработка программного обеспечения по проекту невозможна до тех пор, пока не будет определена тема работы и не будет изучен соответствующий теоретический материал.

Задача, влияющая на другую задачу, называется *предшественником (Predecessor)*, а задача, зависящая от другой, называется *последователем (Successor)*. Например, задача «Изучить необходимый теоретический материал» является предшественником для задачи «Разработать необходимое программное обеспечение», а задач «Разработать необходимое программное обеспечение» последователь для задачи «Изучить необходимый теоретический материал».

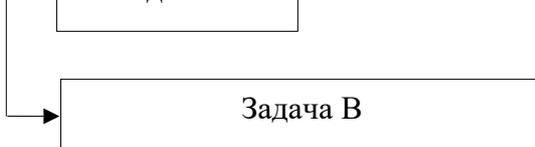
Каждая задача может иметь неограниченное количество предшествующих и последующих задач. Связи могут объединять и фазы. Все принципы организации связей между задачами применимы и к фазам.

Имеется четыре типа связей между задачами.

1) Связь типа «Окончание – начало» (*Finish to Start*), или сокращенно *ОН (FS)*. Это наиболее распространенный тип связи между задачами, при котором задача В не может начаться раньше, чем закончиться задача А. Графически этот тип связи описывается следующим образом:

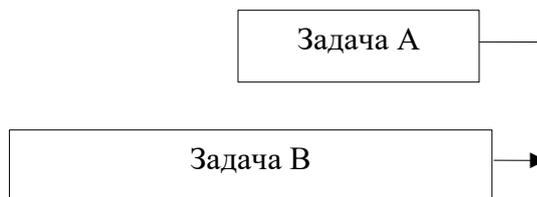


2) Связь типа «Начало – Начало» (*Start to Start*), или сокращенно *НН (SS)*, обозначает зависимость, при которой задача В не может начаться до тех пор, пока не началась задача А. С помощью такой связи обычно объединяются задачи,

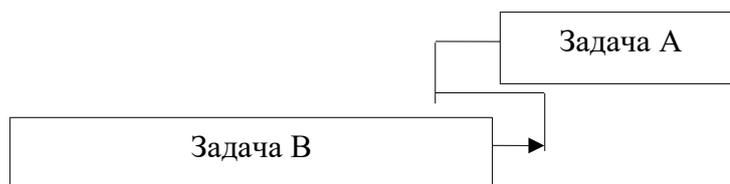


которые должны выполняться почти параллельно. Графически этот тип связи описывается следующим образом:

3) Связь типа «Окончание-Окончание» (*Finish to Finish*), или сокращенно *ОО (FF)* обозначает зависимость, при которой задача В не может закончиться до тех пор, пока не закончилась задача А. Обычно такой связью объединяются задачи, которые должны выполняться почти одновременно, но при этом одна не может закончиться, пока не завершена другая. Например, сдача программы идет одновременно с исправлением ошибок, и пока исправление ошибок не завершено, сдача программы не может завершиться.



4) Связь типа «Начало - Окончание» (*Start to Finish*), или сокращенно *НО (SF)*. Обычно такая связь используется в том случае, когда А является задачей с фиксированной датой начала, а задача В не может закончиться до тех пор, пока не началась задача А.



Связь создается перетаскиванием мышью одного отрезка диаграммы Ганта на другой, при этом тип связи по умолчанию определяется как ОН (Окончание - Начало). Предшествующей считается задача, с которой началось перетаскивание, а последующей та, на которой перетаскивание закончилось (на последующую задачу указывает стрелка в конце связи) (рисунок 13).

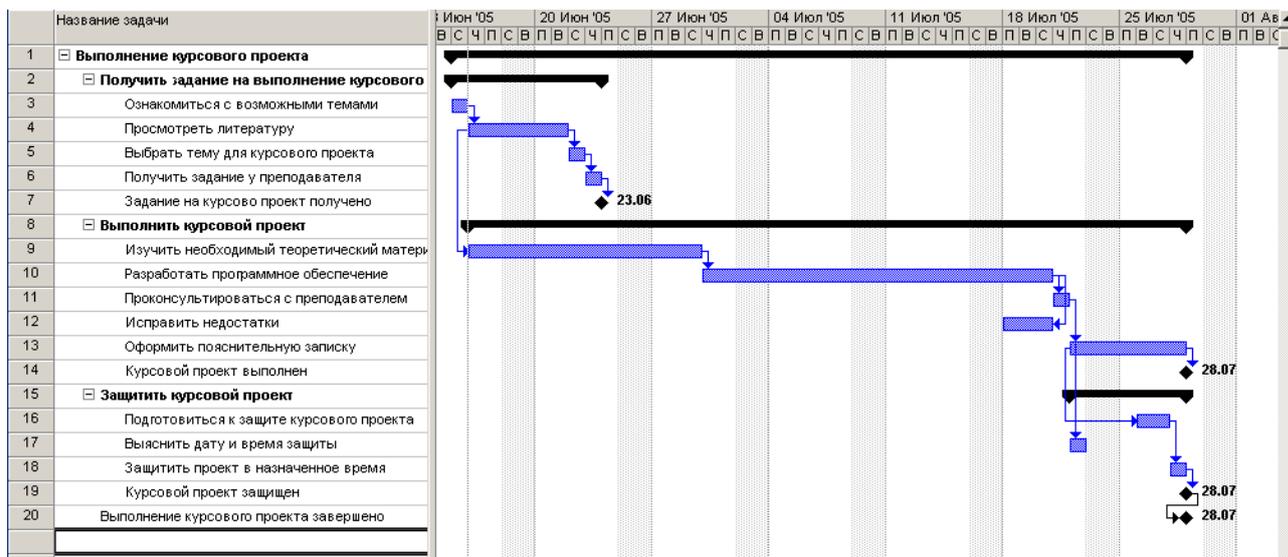


Рисунок 13. Диаграмма Ганта со связями между задачами

Для удаления связи или изменения ее типа необходимо дважды щелкнуть на ней и произвести соответствующие операции в открывшемся диалоговом окне (рисунок 14).

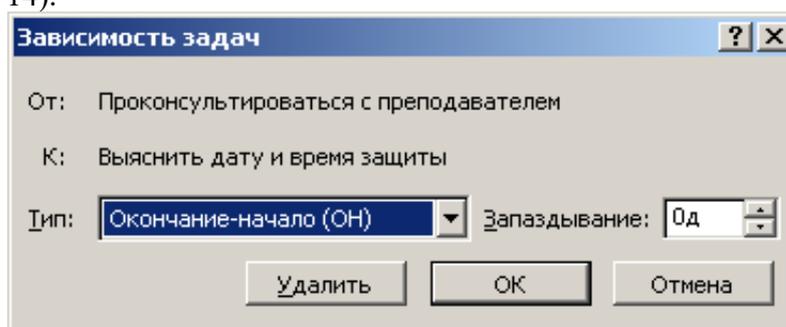


Рисунок 14.

Окно настройки зависимости задач

В раскрывающемся списке *Тип (Type)* можно выбрать тип связи, в поле со счетчиком *Запаздывание (Lag)* указать временной интервал между связанными задачами. Кнопка *Удалить (Delete)* позволяет удалить связь.

Упражнение № 5. Форматирование диаграмм Ганта

Диаграмма Ганта (Gant Chart) названа в честь Генри Ганта (1861-1919). Диаграмма Ганта является основным средством визуализации плана проекта. Все элементы диаграммы Ганта являются настраиваемыми отрезками, каждый из которых может состоять из трех элементов: точки начала, точки окончания и промежуточной части (при этом любой их этих элементов может отсутствовать). При стандартной настройке отрезки, обозначающие фазы, состоят из трех элементов, отрезки, обозначающие задачи – только из промежуточной части, а завершающие задачи только из начальной точки. Длина отрезков, обозначающих фазы и задачи пропорциональна их длительности.

На диаграмме Ганта рядом с отрезками может отображаться дополнительная информация. Состав этой информации определяется настройками программы. Входит несколько заранее настроенных версий диаграммы Ганта, список которых приведен в таблице 3.

Таблица 3. Предопределенные версии диаграмм Ганта

<i>Название диаграммы</i>	<i>Описание</i>
Подробная диаграмма Ганта (Detail Gantt)	Диаграмма используется при оптимизации плана проекта, когда требуется равномерно распределить нагрузку между ресурсами. На ней отображаются возможные периоды времени, на которые исполнение задачи можно отложить, не сдвигая срока окончания проекта
Диаграмма Ганта с выравниванием (Leveling Gantt)	Диаграмма используется для выравнивания нагрузки ресурсов. На ней отображаются все изменения, осуществленные в процессе выравнивания
Диаграмма Ганта с отслеживанием (Tracking Gantt)	Диаграмма используется для сравнения запланированных сроков выполнения проекта и реальных сроков исполнения работ. Для каждой задачи и фазы отображаются запланированный и реальный сроки исполнения
Диаграмма Ганта с несколькими планами (Multiple Baseline Gantt)	Диаграмма используется для сравнения трех первых базовых планов проекта
Диаграмма Ганта с ожидаемым планом проекта (PA_Expected Gantt), Диаграмма Ганта с оптимистичным планом проекта (PA_Optimistic Gantt), Диаграмма Ганта с пессимистичным планом проекта (PA_Pessimistic Gantt)	Диаграммы предназначены для анализа плана работ по методу PERT (PERT Analysis, или сокращенно PA). Поскольку метод заключается в построении трех планов (реалистичного, оптимистичного и пессимистичного) и их дальнейшем анализе, то для работы с ним используются три диаграммы

Чтобы воспользоваться предопределенной версией диаграммы Ганта, необходимо выбрать ее название в пункте меню *Вид → Другие представления...*

В том случае, если необходимо изменить вид стандартной диаграммы Ганта или ее версии, можно использовать средства форматирования диаграмм, которые позволяют:

- изменять форму и цвет составляющих диаграмму отрезков;
- определять, какая проектная информация отображается на диаграмме рядом с отрезками;
- отображать дополнительную графическую информацию (например, отклонение от базового плана);
- форматировать шкалу времени, уменьшая или увеличивая масштаб отображения плана.

Чтобы изменить внешний вид отрезков диаграммы Ганта необходимо щелкнуть правой клавишей мыши над необходимым отрезком, в контекстном меню выбрать пункт *Форматировать отрезок...* и изменить необходимые значения настроек.

Существует также возможность группового форматирования элементов диаграммы, которое позволяет полностью настраивать вид диаграммы, определяя, какая именно информация из проектного файла и каким образом отображается на диаграмме. Для этого в диалоговом окне, вызываемом с помощью команды *Формат → Стили отрезков* (рисунок 15).

Окно состоит из двух основных блоков. Верхний блок содержит таблицу, в которой определяются отражаемые на диаграмме типы отрезков. Нижний блок содержит две вкладки с параметрами, определяющими внешний вид типов отрезков, указанных в верхнем блоке.

В столбцах таблицы верхнего блока задаются основные свойства типов отрезков диаграмм.

В столбце *Название (Name)* определяется название типа отрезка, которое будет отображаться во всплывающей подсказке при наведении указателя мыши на отрезок.

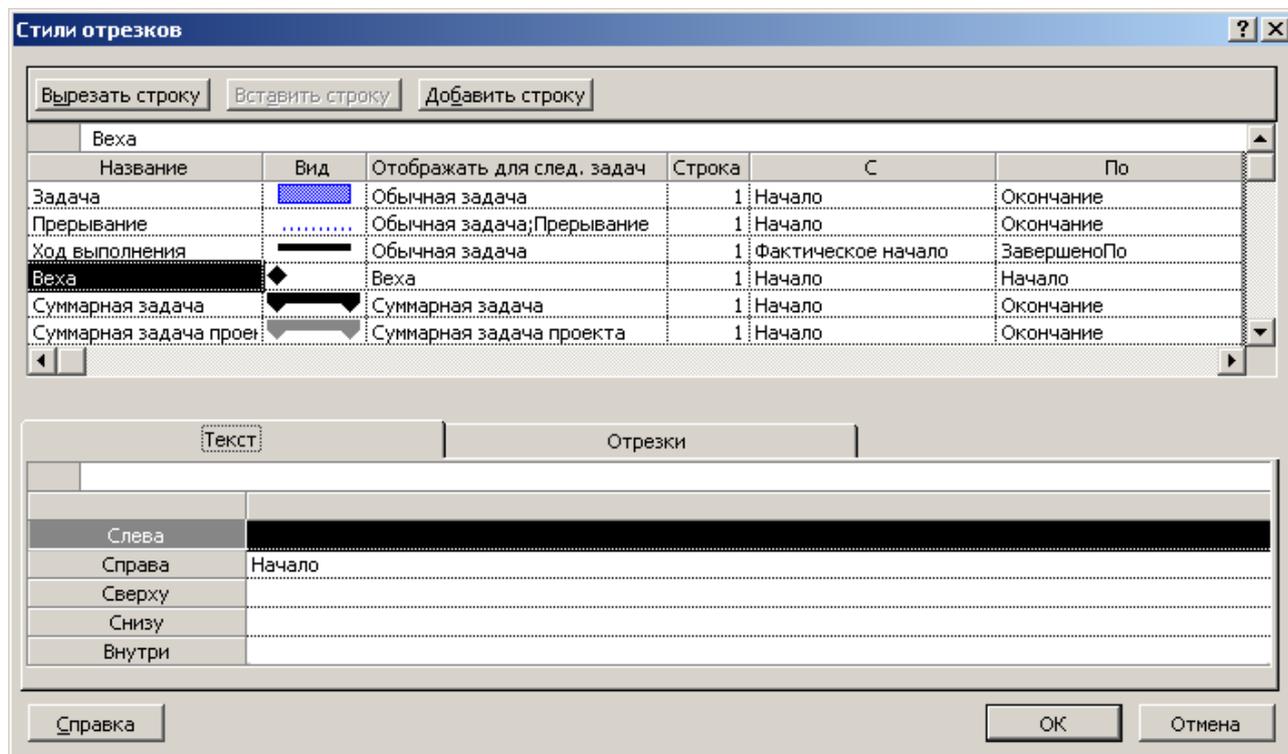


Рисунок 15. Окно «Стили отрезков»

Поле *Вид (Appearance)* отражает внешний вид элемента диаграммы в соответствии с настройками в нижнем блоке таблицы. Для того, чтобы это поле изменилось, нужно установить курсор в строку с нужным типом отрезков и настроить параметры отображения на вкладках нижней части окна.

Поле *Отображать для след. задач (Show for ...Tasks)* определяет, какие задачи отбирать для отображения данным стилем. Отбор происходит на основании полей типа *Флаг (Flag)*, которые могут содержать только значения *Да (Yes)* и *Нет (No)*. Чтобы определить тип задач, нужно выбрать одно или несколько таких полей в раскрывающемся списке.

В тех случаях, когда необходимо вывести на диаграмме несколько типов фигур для одного типа задач, но не желательно, чтобы они наслаивались друг на друга, можно воспользоваться полем *Строка (Row)*. Номером в поле *Строка* определяется порядок отображения фигур: первым отображается отрезок номер 1, вторым – 2 и т.д.

В полях *С (From)* и *По (To)* определяются начальные и конечные точки отрезка на диаграмме.

Упражнение № 6. Сетевой график

Хотя подавляющее большинство операций по планированию и анализу проекта выполняется с использованием диаграммы Ганта, для работы с сетевым графиком также предусмотрен богатый набор функций. Наибольший эффект от их

применения можно получить на этапе разработки структуры нового проекта «с нуля», а также при анализе проекта с точки зрения возможных рисков.

В отличие от классического сетевого графика, на котором работам соответствуют дуги, а события – вершины, в программном обеспечении используется другой вариант графика, при котором работам соответствуют вершины, а дуги переходам от одной работы к другой.

Есть три вида графиков, которые можно отнести к сетевым: собственно *сетевой график (Network Diagram)*, *сетевой график с описанием (Descriptive Network Diagram)* и *схема данных (Relationship Diagram)*.

Отличие сетевого графика с описанием от обычного сетевого графика заключается только в повышенной информативности блоков: по умолчанию в них отображается больше информации. Принцип построения и внешний вид схемы данных отличается от первых двух графиков. Разница состоит в том, что на первых двух графиках можно сразу просматривать информацию обо всем плане проекта, а на схеме данных – только об одной задаче этого плана.

Блоки сетевого графика могут различаться цветом и формой в зависимости от типа задачи (обычная задача, завершающая задача или фаза) и ее состояния (исполняется, не выполняется, завершена). На блоке может быть указана любая дополнительная информация: даты начала и окончания, длительность, задействованные ресурсы. По умолчанию фазы обозначаются параллелограммами, задачи – прямоугольниками, завершающие задачи – шестиугольниками. Начатые задачи перечеркиваются одной линией, а завершенные двумя. Стрелки соответствуют связям между задачами.

Включен большой набор средств для форматирования сетевых графиков. Принципы форматирования сетевого графика не отличаются от принципов форматирования диаграмм Ганта: можно отформатировать отдельный элемент графика, группу элементов и настроить дополнительные параметры группы.

Чтобы отформатировать блок, нужно щелкнуть на нем правой клавишей мыши и выбрать пункт контекстного меню *Формат рамки (Format Box)*.

Возможности форматирования групп блоков сетевого графика зависят от того, какие типы задач эти блоки символизируют на графике. Групповое форматирование блоков осуществляется в диалоговом окне, определяющем свойства групп блоков – пункт меню *Формат – Стили рамок*.

Сетевой график, как и диаграмма Ганта, позволяет редактировать план проекта. Двойной щелчок на блоке позволяет редактировать свойства задачи. Для редактирования отдельного свойства задачи, отображенного на блоке, нужно щелчком мыши установить курсор в соответствующую ячейку блока и затем отредактировать ее.

Для создания задачи (блока) нужно нажать кнопку мыши на свободном месте графика, растянуть на нем прямоугольную рамку, определяющую размер будущего блока, и отпустить кнопку мыши. Создавать задачу на графике не очень удобно, потому что не всегда можно точно предсказать, к какой фазе проекта будет отнесена новая задача. Изменение уровня вложенности задачи производится с помощью комбинаций клавиш *Shift+Alt+←* и *Shift+Alt+→*. Удалить задачу можно нажатием клавиши *Delete*. Связи между задачами создаются перетаскиванием задач друг на друга с помощью мыши, как и в диаграмме Ганта.

Сетевой график удобно настраивать с помощью панели инструментов *Сетевой график (Network Diagram)*, которая вызывается с помощью окна *Сервис – Настройка – Панели инструментов*.

Схема данных – это особый вид сетевого графика, предназначенный только для анализа связей между задачами проекта. Схема данных разделена на страницы, которые можно пролистывать с помощью полосы прокрутки. На одной странице

можно просматривать информацию только об одной задаче. Страницы с задачами расположены в порядке возрастания номеров задач. Диаграмма не позволяет редактировать блоки или проектные данные – их можно только просматривать. Использовать эту диаграмму удобно для анализа плана проекта с множеством зависимостей, поскольку с ее помощью легко сфокусироваться на задачах, связанных с выбранной.

Практическая работа №4 Создание собственного проекта (по вариантам) Порядок выполнения работы

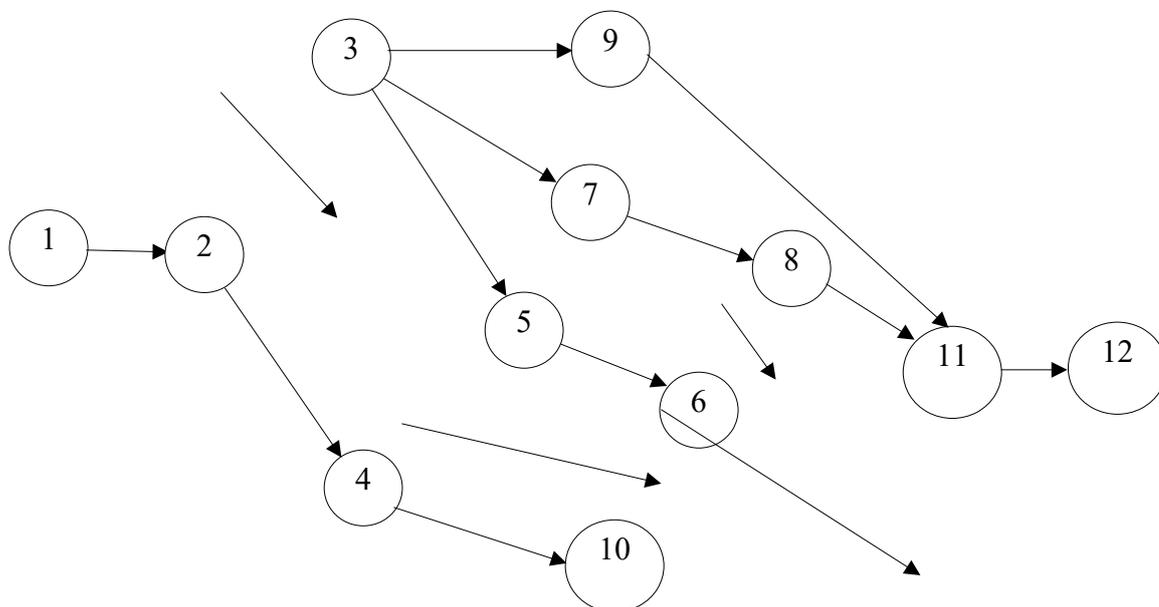
- 1) Повторите теоретический материал практических работ №1-3.
- 2) Выполните задание, соответствующее номеру в журнале.
- 3) Подготовьте ответы на контрольные вопросы в письменном виде.

Варианты заданий

Создайте диаграмму Ганта для перечисленных ниже работ. Разбиение на фазы проведите самостоятельно. Планирование производить по методике планирования от даты начала проекта. Отформатируйте диаграмму. Сохраните созданный проектный файл для дальнейшего использования.

Вариант 1

График разработки и внедрения программного комплекса [3] приведен на рисунке.



Работы:

- 1) Предварительное определение перечня и структуры выдаваемых документов, информационных массивов и характеров их использования (40 дней).
- 2) Разработка общей схемы решения задачи, утверждение перечня и форм выдаваемых документов, выдача задания на программирование (30 дней).

- 3) Определение структур данных и способов кодирования информации (35 дней).
- 4) Обеспечение формирования исходных данных (20 дней).
- 5) Обеспечение формирования нормативных массивов (12 дней).
- 6) Обеспечение формирования базовых массивов (30 дней).
- 7) Разработка программного обеспечения (100 дней).
- 8) Отладка программ (30 дней).
- 9) Техническое обеспечение решения задачи (12 дней).
- 10) Организационное обеспечение решения задачи (20 дней).
- 11) Опытно-промышленная проверка (30 дней).
- 12) Корректировка по результатам проверки (60 дней).

Вариант 2

В соответствии с [4] выделяют три основных типа документов, составляющих централизованный документооборот: входящие документы – документы поступившие из внешних организаций, исходящие документы – документы, создаваемые структурными подразделениями для отправки во внешние организации и внутренние документы, издаваемые руководителем или структурными подразделениями для рассылки внутри службы.

Входящие документы подвергаются следующим операциям:

- экспедиционной обработке (3 часа);
- регистрации (2 мин.);
- рассмотрению руководством и выдаче резолюции (4 часа);
- ознакомлению в структурном подразделении или исполнению (3 дня);
- контролю за исполнением, включающему постановку на контроль документов (10 мин.);

- отслеживание хода исполнения (30 мин.);
- составлению делопроизводственных отчетов для руководства (1 час);
- снятию документов с контроля (2 мин.);
- передача документов в архив (10 мин.)

Вариант 3

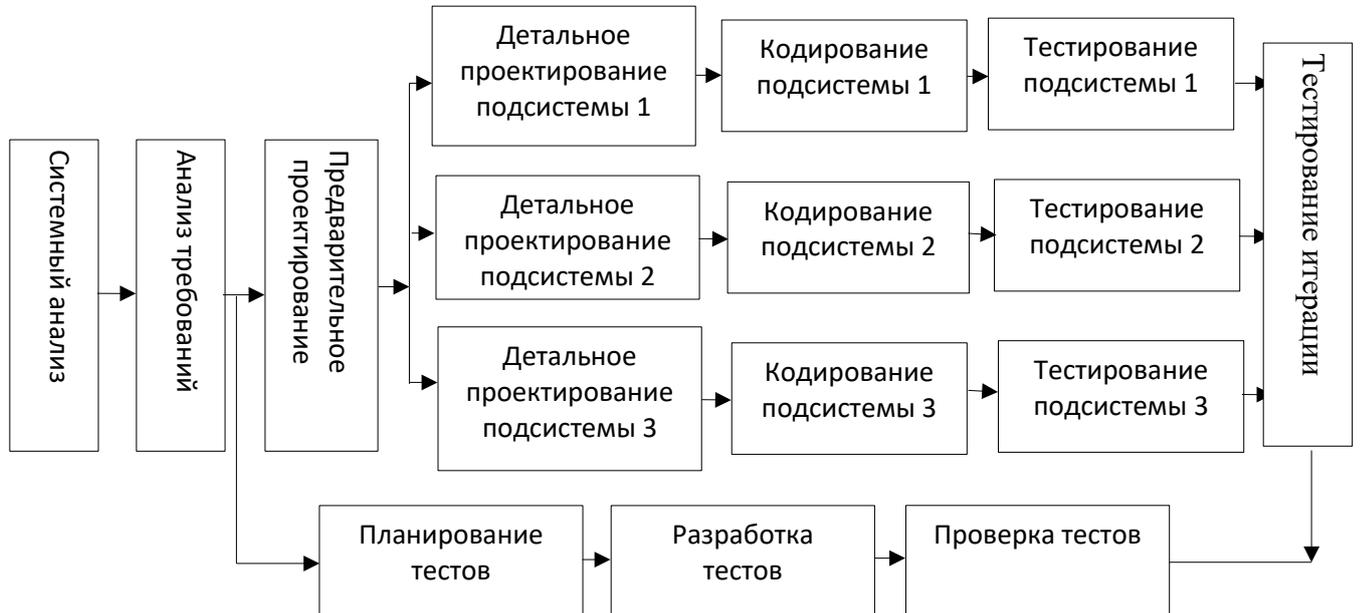
В [5] рассмотрен следующий пример формулировки требований на начальных итерациях разработки программного проекта:

Дисциплина	Артефакт	Комментарии по поводу формулировки требований				
		Начало 1 неделя	Развитие 1 4 недели	Развитие 2 4 недели	Развитие 3 3 недели	Развитие 4 3 недели
Требования	Модель прецедентов	Двухдневный семинар по формулировке требований. Только 10% прецедентов описываются подробно	В конце этой итерации проводится двухдневный семинар по формулировке требований. При этом учитываются результаты выполненной работы, 30 % прецедентов описываются подробно	В конце этой итерации проводится двухдневный семинар по формулировке требований. При этом учитываются результаты выполненной работы, 50 % прецедентов описываются подробно	То же самое, 70% прецедентов описываются подробно	80 – 90% прецедентов описываются подробно. Только небольшая часть прецедентов реализуется на стадии развития, реализация остальных переносится на стадию конструирования
Проектирование	Модель проектирования	Проектирование для небольшого числа архитектурно важных требований с высокой степенью риска	Проектирование для небольшого числа архитектурно важных требований с высокой степенью риска	То же самое	То же самое	То же самое. Все архитектурно важные элементы с высокой степенью риска должны быть спроектированы к этому моменту
Реализация	Модель реализации (код)		Реализация проектного решения	То же самое. 5% окончательной системы должно быть построено	То же самое. 10% окончательной системы должно быть построено	То же самое. 15% окончательно системы должно быть построено
Управление проектом	План разработки	Достаточно общая оценка требуемых затрат и ресурсов	Оценки конкретизируются	Оценки конкретизируются	Оценки конкретизируются	Общая длительность проекта, основные этапы

Количество прецедентов в проекте – 16. Общее время разработки системы – 30 недель.

Вариант 4

Основной задачей при планировании программного проекта является определение структуры распределения работ [6].



Первыми выполняемыми задачами являются системный анализ и анализ требований к программному продукту. Они закладывают фундамент для последующих параллельных задач.

Системный анализ проводится с целью:

- выяснения потребностей заказчика (3 недели);
- оценки выполнимости системы (1 неделя);
- выполнения экономического анализа (1 неделя);
- выполнения технического анализа (1 неделя);
- распределения функций по аппаратуре, программам, сотрудникам (2 неделя);
- определения стоимости и ограничений планирования (1 неделя);
- создания системной спецификации (2 недели).

Анализ требований дает возможность:

- определить функции и характеристики программного продукта (1 неделя);
- определить интерфейс программного продукта с другими системными элементами (2 недели);
- определить программные ограничения программного продукта (1 неделя);
- построить модели функционирования продукта (2 недели).

Результаты анализа требований сводятся в спецификацию требований к программному продукту.

Время выполнения других работ:

- предварительное проектирование - 2 недели;
- детальное проектирование подсистемы 1 – 3 недели, подсистемы 2 – 1 неделя, подсистемы 3 – 2 недели;
- планирование тестов – 2 недели;
- разработка тестов – 3 недели;
- проверка тестов – 1 неделя;
- тестирование итерации – 1 неделя;

- создание спецификации требований – 1 неделя.

Вариант 5

В последнее время неуклонно возрастает интерес к технологиям поддержки принятия решений [7].

В общем виде схема поддержки принятия решения включает следующие блоки: выявление проблем, формулировка задач, формализация задач, анализ ресурсов, сбор данных и их анализ, синтез решения и формирование альтернатив, анализ альтернатив и выработка рекомендаций. Сведения об обобщенном процессе принятия некоторого решения приведены в таблице. Задачи, которые могут быть решены параллельно, определить самостоятельно.

Выявление проблем		
Мониторинг внешней среды		1 неделя
Выявление тенденций поведения рынка и внешней среды		1 неделя
Мониторинг объекта управления		2 недели
Анализ поведения продукции на рынке		1 неделя
Получение прогнозных оценок		1 неделя
Выявление во внешней среде прогнозных ситуаций, не позволяющих принимать решение ранее известными методами		1 неделя
Выявление истощения ресурсов, используемых для принятия решений или производства		1 неделя
Выявление снижения деятельности управляемого объекта или методов управления		2 недели
Выявление появления новых средств производства, управления или информационных продуктов		1 неделя
Выявление проблемы		2 недели
Оценка новизны проблемы		1 неделя
Установление взаимосвязи с другими проблемами		1 неделя
Оценка полноты и достоверности информации по данной проблеме		1 неделя
Ведение базы данных информационных ресурсов по проблеме		10 недель
Формулировка задач		
Определение разрешимости проблемы		1 неделя
Концептуальная разработка вариантов решения проблемы		1 неделя
Декомпозиция выявленной проблемы на отдельные задачи		1 неделя
Формулировка условий		2 недели
Формулировка целей		2 недели
Логический анализ условий, целей и задач		1 неделя
Постановка задач исполнителям		1 неделя
Формализация		
Классификация сформулированных целей и задач		1 неделя
Выбор системы формализации		1 неделя
Формализация условий и целей		2 недели
Построение инфологической модели		1 неделя
Формальный логический анализ условий		1 неделя
Поиск задач-аналогов		2 недели
Поиск регламентных решений задач		1 неделя
Декомпозиция задачи		1 неделя
Детализация входных и выходных данных		4 недели
Детализация целей		2 недели
Детализация условий		2 недели
Составление блок-схемы и ее формального описания		4 недели

Анализ ресурсов	
Анализ методов полного решения	2 недели
Анализ методов частичного решения	2 недели
Анализ информационных ресурсов для решения	1 неделя
Анализ интеллектуальных ресурсов для решения	1 неделя
Анализ организационных решений для применения решения	1 неделя
Анализ технологических ресурсов для применения решения	1 неделя
Сбор данных и их анализ	
Информационный поиск необходимых данных	4 недели
Перевод собранных данных в электронную форму	4 недели
Организация и ведение базы данных	10 недель
Анализ полученных данных	1 неделя
Анализ существующих моделей и информационных потоков	2 недели
Синтез решений и формирование альтернатив	
Структуризация задачи	1 неделя
Структуризация решений	1 неделя
Выделение линейной последовательности действий	1 неделя
Выделение рекурсивных последовательностей	1 неделя
Выделение циклических последовательностей	1 неделя
Выделение параллельных операций	1 неделя
Выделение временной последовательности обработки	1 неделя
Моделирование решений	4 недели
Формальный логический и структурный анализ	1 неделя
Контроль за выполнением решения	постоянно
Получение прогнозных оценок	2 недели
Формирование альтернатив	2 недели
Ведение банка данных моделей и решений	4 недели
Анализ альтернатив	
Классификационный анализ альтернатив	1 неделя
Качественный анализ альтернатив	2 недели
Параметрический анализ альтернатив	2 недели
Некритериальный анализ	1 неделя
Прогнозирование состояний рынка или среды	2 недели
Рекомендации по применению альтернатив для разных состояний среды или рынка	1 неделя

Вариант 6

В соответствии с [4] выделяют три основных типа документов, составляющих централизованный документооборот: входящие документы – документы, поступившие из внешних организаций, исходящие документы – документы, создаваемое структурными подразделениями для отправки во внешние организации и внутренние документы, издаваемые руководителем или структурными подразделениями для рассылки внутри службы.

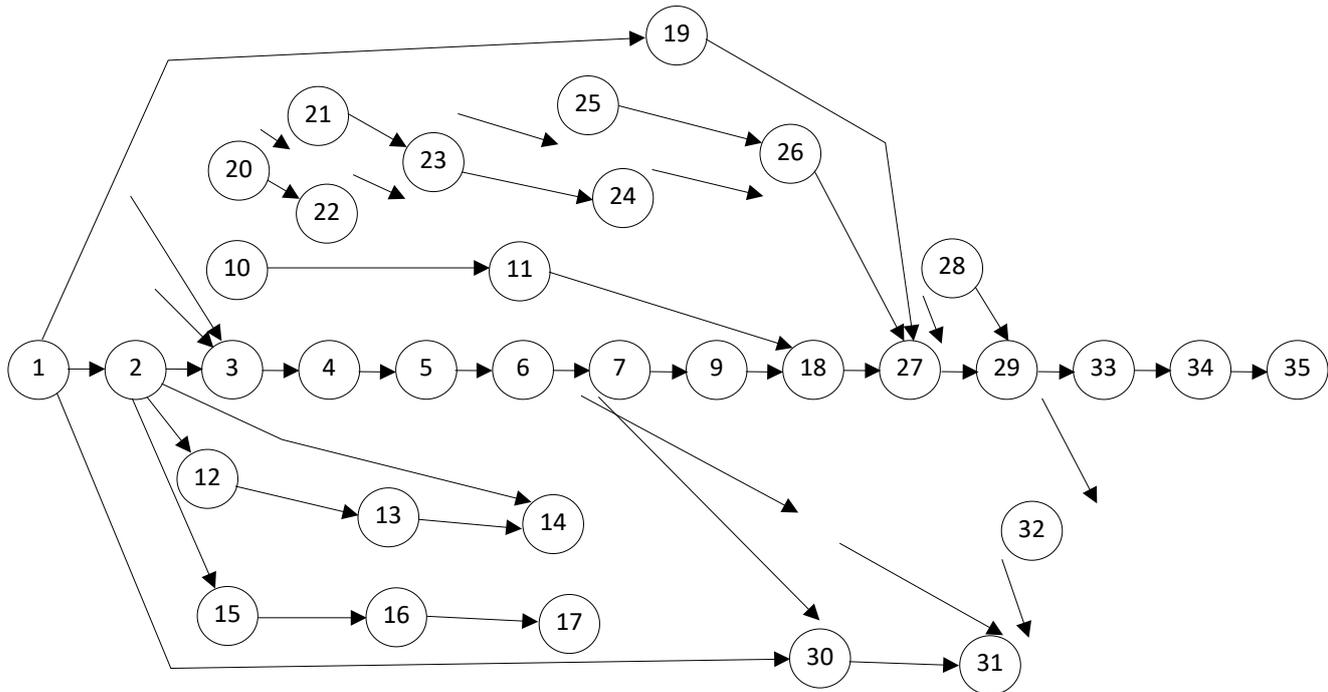
Основные процедуры обработки исходящих документов:

- разработка проекта документа в структурном подразделении, включающая сбор необходимых сведений (2 дня), оформление документа (1 час), утверждение документа руководителем подразделения (2 часа);
- согласование проекта документа в структурных подразделениях организации, которая включает в себя согласование с экономической службой (2 дня); согласование с юридической службой (1 день), согласование с кадровой службой (1 день), согласование с хозяйственной службой (1 день);
- утверждение документа руководством (1 день);
- регистрация документа (10 мин.);

- экспедиционная обработка документа (4 часа).

Вариант 7

На сетевом графике ввода в производство и производства нового станка, представленном на рисунке [1], работы обозначены стрелками, а узлы указывают на очередность работ. Узел означает событие – момент завершения всех работ стрелок, входящих в этот узел. Пояснения к графику и продолжительности работ сведены в таблицу.



Наименование работы	Обозначение работ на графике	Продолжительность (дни)
Конструирование изделия	1-2	60
Корректировка чертежей	2-10	10
Распределение работ по цехам	2-3	2
Составление технологии на механическую обработку	3-4	10
Нормирование механической обработки	11-18	5
Выдача заказ-наряда на конструирование оснастки	4-5	2
Конструирование оснастки	5-6	20
То же	6-7	20
Составление технологии на детали оснастки	6-8	10
То же	7-9	10
Изготовление оснастки	8-9	25
То же	9-18	25
Заказ металла	12-13	3
Составление ведомости на заказ металла	2-12	1
Обеспечение завода металлом	13-14	10
Составление технологии на заготовки	2-14	10
Изготовление заготовок	14-18	25
Заказ литья и поковок	2-15	5
Обеспечение завода литьем, поковками	15-16	15
Старение литья	16-17	8

Окраска литья	17-18	3
Механическая обработка деталей	18-27	14
Заказ материалов и комплектующих изделий для сборки	2-19	2
Обеспечение сборки материалами и комплектующими изделиями	19-27	9
Составление технологии на сборку	2-20	7
То же	21-23	7
Нормирование сборочных работ	20-27	6
Выдача заказ-нарядов на оснастку для сборки	20-22	2
Конструирование оснастки для сборки	23-25	19
Составление технологии на детали оснастки для сборки	23-24	10
То же	25-26	10
Составление технологии на детали оснастки для сборки	23-24	17
То же	25-26	17
Изготовление оснастки для сборки	24-26	18
То же	26-27	18
Узловая сборка под механическую обработку	27-28	4
Узловая сборка	27-29	4
Дополнительная механообработка собранных узлов	28-29	2
Общий монтаж станка	29-33	5
Конструирование наладок по спецстанкам	2-30	17
Заказ литья и поковок для наладок	30-31	2
Обеспечение литьем и поковками для наладок	31-33	12
Составление технологии на детали наладок	30-33	10
Изготовление наладок	32-33	17
Наладка и сдача станка	33-34	10
Окраска станка	34-35	1

Вариант 8

В соответствии с [4] выделяют три основных типа документов, составляющих централизованный документооборот: входящие документы – документы поступившие из внешних организаций, исходящие документы – документы, создаваемое структурными подразделениями для отправки во внешние организации и внутренние документы, издаваемые руководителем или структурными подразделениями для рассылки внутри службы.

При создании внутренних документов делопроизводители выполняют следующие операции:

- принятие решения и выдача руководителем задания на разработку внутреннего документа (1 день);
- подготовка проекта внутреннего документа в подразделении, состоящего из: сбора необходимых сведений (1 день), подготовка текста документа секретарем (1 день), оформление документа (2 часа);
- обеспечение согласования документа, включающая: согласование с заинтересованными лицами (1 день), согласование со структурными подразделениями (2 дня);
- утверждение документа (4 часа);
- регистрация документа (10 мин.);

- доставка документа до подразделений, которая осуществляется 3 сотрудниками, которым необходимо доставить документ в 14 подразделений (20 минут для доставки в одно подразделение);
- контроль исполнения документа (с момента окончания доставки в течении 15 дней).

Контрольные вопросы

- 1) Как осуществляется настройка программы.
- 2) Как хранятся и отображаются данные (что такое «внутренняя» и «внешняя» таблицы).
- 3) Какие стандартные таблицы входят в состав программного обеспечения.
- 4) Что такое диаграмма Ганта.
- 5) Как производится групповое форматирование элементов диаграммы Ганта.
- 6) Назовите predetermined версии диаграмм Ганта и опишите их назначение.
- 7) Как настроить формат отрезка на диаграмме Ганта.
- 8) Как составляется скелетный план проекта.
- 9) Что такое веха. Как создаются вехи.
- 10) Какие типы связей могут быть определены между задачами.
- 11) Что такое фаза и как создаются фазы.

Практическая работа №5

Настройка ресурсов проекта

Ввести начало проекта – 1 июня 2015 года – меню Проект... Сведения о проекте...

Чтобы **изменить рабочие часы базового календаря проекта**: Сервис – Изменить рабочее время, в этом же окне можно задать нерабочие дни.

Сначала набрасывается черновой план проекта, состоящий из 5-15 основных этапов. Ставятся вехи **НАЧАЛО** и **ОКОНЧАНИЕ** проекта. Первая строка – заголовок проекта.

Оценить длительность этапов работ, используя опыт реализации подобных проектов и консультации экспертов, участвовавших в выполнении подобных задач. Для быстрого ввода длительности используйте сокращения – 2н, 1мес...

Ввести список задач, вех и длительности задач как в примере:



После расположения задач в логической последовательности необходимо создать структуру, представляющую иерархию выполняемых задач. Задача,

расположенная на самом верхнем уровне структуры расписания, называется **суммарной задачей**. Задачи более низкого уровня называются **подзадачами**, которые в свою очередь могут быть декомпозированы на более мелкие управляемые элементы – пакеты работ и работы. Программное обеспечение поддерживает до девяти уровней вложенности задач.

Для перемещения задач на один уровень ниже необходимо выделить эти задачи и нажать кнопку на панели инструментов «Форматирование» - «**На уровень ниже**». Выделенные задачи становятся подзадачами, а вышестоящая задача становится для них суммарной.

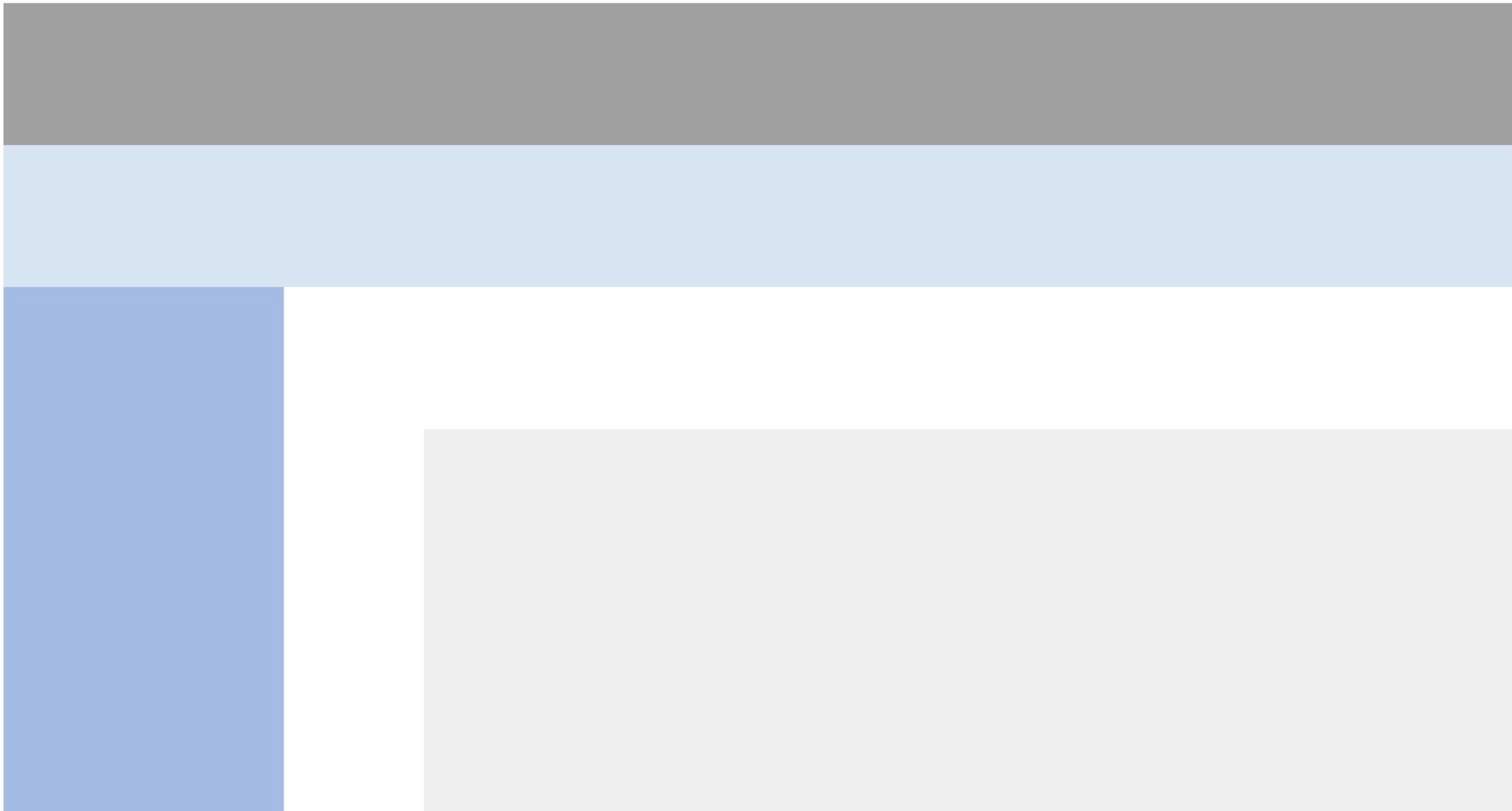


После того, как предварительный план проекта создан, расшифровывают каждый этап отдельно с подключением учета денежных средств. На практике применяется метод «набегающей волны» т.е. те этапы, которые планируются в ближайшем будущем, раскрываются очень подробно и точно, те, которые находятся в отдаленном будущем, имеют пока приблизительную расшифровку. Это удобно, потому что после начала исполнения плана могут возникнуть дополнительные факторы, влияющие на выполнение последующих этапов.

Номер	Структура	Название	Длительность	Номер предшественника
1	1	Ремонт офиса	-	
2	1.1	Начало работ	0 дней	
3	1.2	Подготовительные работы	-	
4	1.2.1	Вынос мебели	2 дней	2
5	1.2.2	Демонтаж окон	2 дней	4
6	1.2.3	Демонтаж сантехники	1 день	5
7	1.2.4	Демонтаж потолков	2 дней	6
8	1.2.5	снятие старых обоев	1 день	7
9	1.2.6	Снятие линолеума	0,5 дней	8
10	1.2.7	Вывоз мусора	0,5 дней	9
11	1.2.8	Подготовительный этап закончен	0 дней	10
12	1.3	Малярные работы	-	

Номер	Структура	Название	Длительность	Номер предшественника
13	1.3.1	Выравнивание потолков	3 дней	21
14	1.3.2	Выравнивание стен	4 дней	13
15	1.3.3	Грунтовка	1 день	14
16	1.3.4	Малярные работы закончены	0 дней	15
17	1.4	Монтажные работы	-	
18	1.4.1	Замена окон	2 дней	5
19	1.4.2	Сантехнические работы	2 дней	6
20	1.4.3	Электрика	1 день	11
21	1.4.4	Монтаж закончен	0 дней	20;19;18
22	1.5	Отделочные работы	-	
23	1.5.1	Покраска потолка	2 дней	16
24	1.5.2	Покраска стен	4 дней	23
25	1.5.3	Линолеум	1 день	24
26	1.5.4	Отделочные работы закончены	0 дней	25
27	1.6	Покупка мебели и оргтехники	-	
28	1.6.1	Покупка мебели	1 день	
29	1.6.2	Сборка мебели	1 день	28;26
30	1.6.3	Расстановка мебели и техники	1 день	29
31	1.7	Завершение проекта	0 дней	30

Под этапом «Подготовительные работы» добавить 8 строк, используя на клавиатуре клавишу **Insert**. Занести название и длительность подзадач. Переместить их на уровень ниже. Определить логические последовательности между подзадачами. То же самое проделать с остальными этапами.

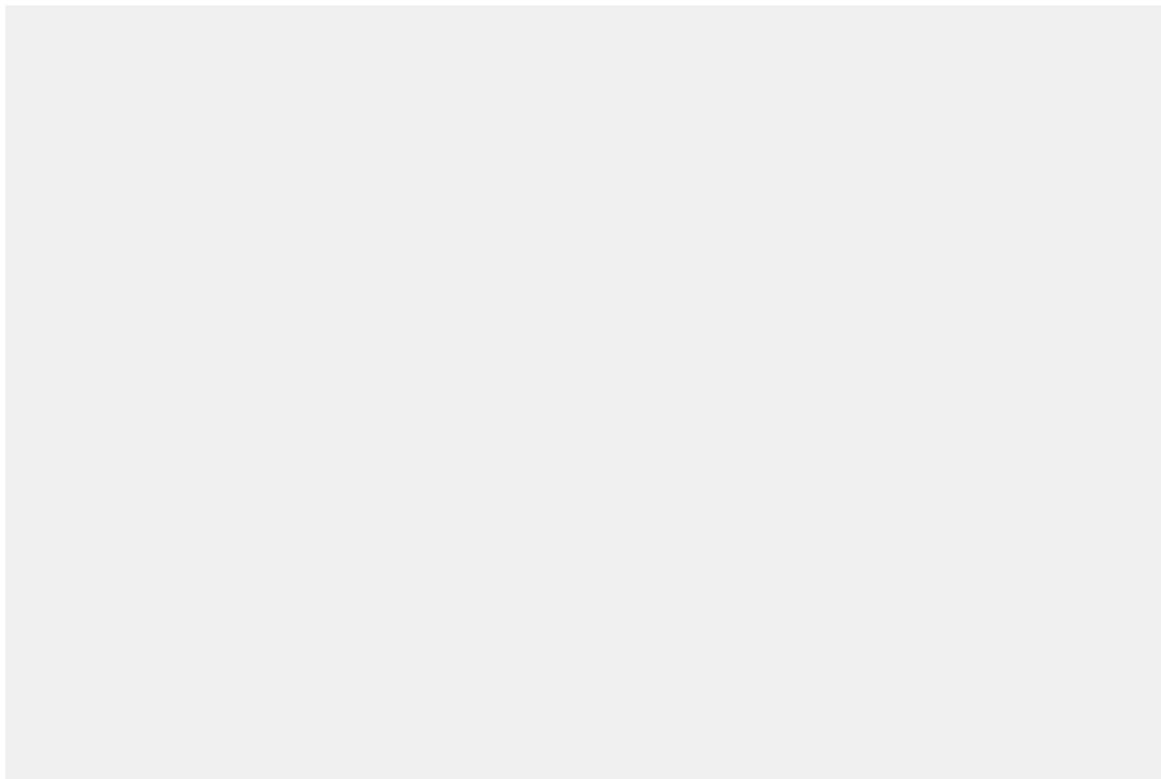


Удобнее работать, когда суммарные задачи не имеют связей, а только показывают общую длительность этапа. Связи обычно ставят между последней задачей предыдущего этапа и первой соответствующего.

Учет затрат на реализацию проекта

Для работы с разными типами данных удобно пользоваться соответствующими таблицами: **Вид – Таблица – ...**(выбрать таблицу с необходимыми данными). Внесение основных данных о задачах (длительность, начало, окончание и др.) производилось в **Таблице Ввод**. Для работы с затратами удобно пользоваться одноименной таблицей **Вид...Таблица...Затраты**. В том случае, если необходимо на экране отображать и основные данные о задачах, то в Таблицу Ввод можно вставить столбцы: Фиксированные затраты, Затраты и Начисление затрат.

Воспользовавшись одним из этих способов внести суммы затрат в столбец Фиксированные затраты.



Задача №1 – название проекта является суммарной для всего проекта, поэтому в столбце Затраты будет отображаться сумма затрат для всего проекта.

Ресурсы

После того, как определен состав задач, нужно определить, кто эти задачи будет выполнять, и какое оборудование будет использоваться. Для этого нужно ввести в план проекта список ресурсов и информацию о них, а затем распределить эти ресурсы между задачами.

Список людей и оборудования

Чтобы ввести список ресурсов, необходимо открыть представление **Лист ресурсов** из пункта меню **Вид**.

Внести список ресурсов как в примере:

Название ресурса	Тип	Единицы измерения материала	Краткс назва	Группа	Макс. единиц	Стандартная ставка	Ставка сверхурочных	Затраты на использ.	Начисление	Базовый календарь
менеджер	Трудовой		м		100%	100,00р./час	0,00р./час	0,00р.	Пропорциональное	Стандартный
строитель	Трудовой		с		200%	40,00р./час	0,00р./час	0,00р.	Пропорциональное	Стандартный
Малар	Трудовой		М		200%	30,00р./час	0,00р./час	0,00р.	Пропорциональное	Стандартный
Краска	Материальный	банка	К			100,00р.		0,00р.		
Сантехник	Трудовой		С		200%	35,00р./час	0,00р./час	0,00р.	По окончании	Стандартный
Комплект сантехники	Материальный	комплект	К			4 200,00р.		0,00р.	В начале	
Электрик	Трудовой		Э		200%	35,00р./час	0,00р./час	0,00р.	По окончании	Стандартный
Электрооборудование	Материальный		Э			3 800,00р.		0,00р.	В начале	
ООО "Окна"	Трудовой		О		100%	0,00р./час	0,00р./час	0,00р.	В начале	Стандартный
Линолеум	Материальный	пог.м.	Л			300,00р.		0,00р.	В начале	
Сухие смеси	Материальный	кг	С			100,00р.		0,00р.	Пропорциональное	
Грунтовка	Материальный	л	Г			10,00р.		0,00р.	Пропорциональное	

Программа поддерживает 3 типа ресурсов – Трудовые, Материальные и Затраты. Трудовые ресурсы используются для управления нагрузкой работников и учета трудоемкости проекта. Это основной вид ресурса и он вводится по умолчанию. За единицу измерения усилий, затраченных такими ресурсами при выполнении задачи, принято время. Материальные ресурсы используются для управления закупками и использованием оборудования и материалами. Тип ресурса затраты нужен для более подробного описания затрат.

Столбец **Единица измерения** активен только для материальных ресурсов. В столбце **Группа** отображается имя группы, к которой принадлежит ресурс.

Столбец **Максимальная единица** показывает максимальный объем ресурса, доступный для выполнения любых задач в течение данного проекта (50% - на полставки, 200% - 2 человека). Можно настроить программу так, чтобы максимальная единица показывалась не в процентах, а в числовых значениях: **Сервис...Параметры.... Закладка Планирование... Показывать единицы назначений...** и выбрать нужный пункт.

Стандартная ставка: для материального ресурса – стоимость за единицу, для трудового – стоимость за час.

Ставка сверхурочных: стоимость часа работы трудового ресурса, используется в том случае, если трудовой ресурс занят больше, чем позволяет его максимальная единица.

Затраты на использование: дополнительные одноразовые затраты, применяемые при каждом назначении ресурса.

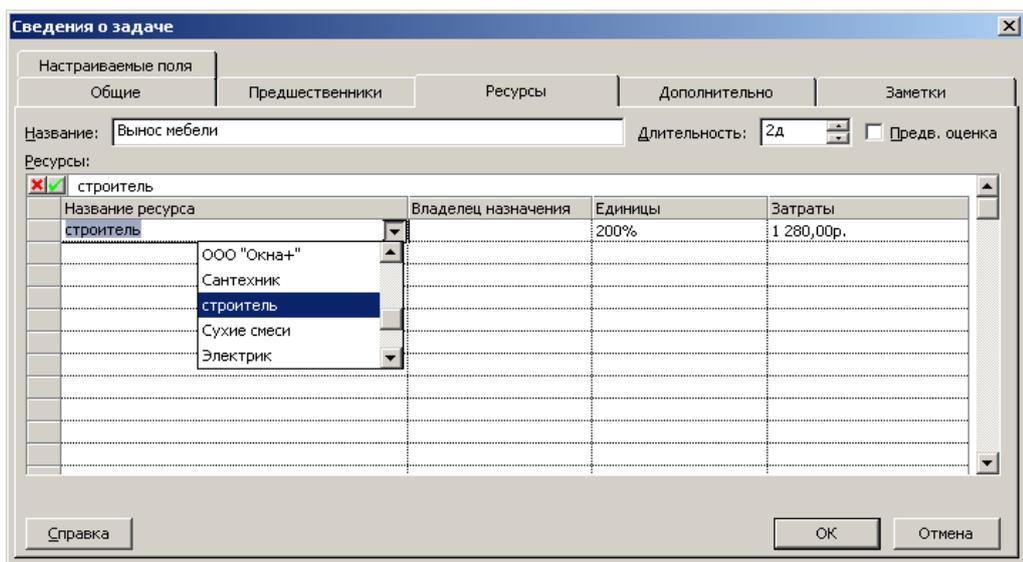
Начисление затрат на ресурсы может производиться в конце, в начале задачи или пропорционально ее выполнению.

Календарь: По умолчанию в качестве базового календаря применяется стандартный календарь проекта, если он не подходит для данного ресурса можно выбрать другие календари из предложенных или создать для отдельных ресурсов свои календари.

Для просмотра всех сведений о ресурсе необходимо выполнить по названию ресурса двойной щелчок мышью. Закладка **Общие** содержит данные о доступности ресурса. Закладка **Затраты** используется в том случае, если выполнение разных работ ресурсом оплачивается по-разному.

Назначение ресурсов задачам

Для назначения ресурсов задачам необходимо вернуться в представление **Диаграмма Ганта** из пункта меню **Вид**. Ресурсы удобно назначать открыв задачу двойным щелчком по названию.

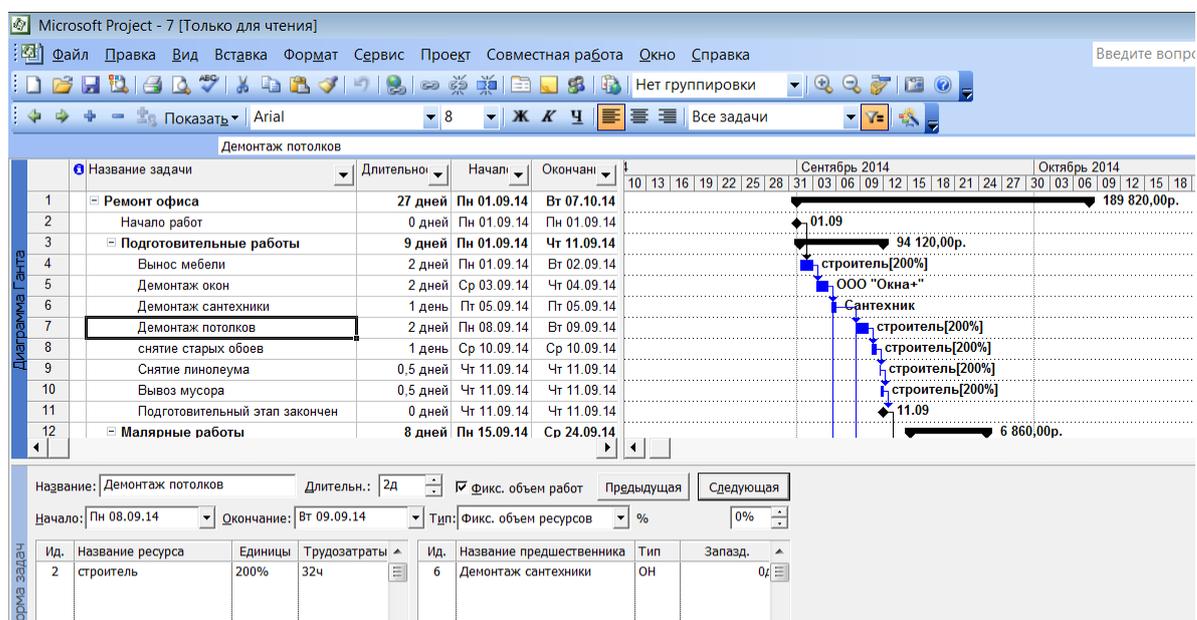


В диалоговом окне **Сведения о задаче** выбрать закладку **Ресурсы**. В столбце **Название ресурсов** в открывающемся списке выбрать необходимый для выполнения задачи ресурс. В столбце **Единицы** обозначить его количество.

Определение трудоемкости

При назначении ресурсов на задачи по умолчанию устанавливается, что работник тратит на выполнение задачи все свое рабочее время. Соответственно при назначении одного работника на несколько параллельных задач получается его перегрузка. Но часть работ выполняется в фоновом режиме в течение нескольких дней или недель. Необходимо определить процент рабочего времени, необходимого для выполнения данной задачи.

Для корректировки загрузки ресурса выберите команду **Разделить** из пункта меню **Вид**.

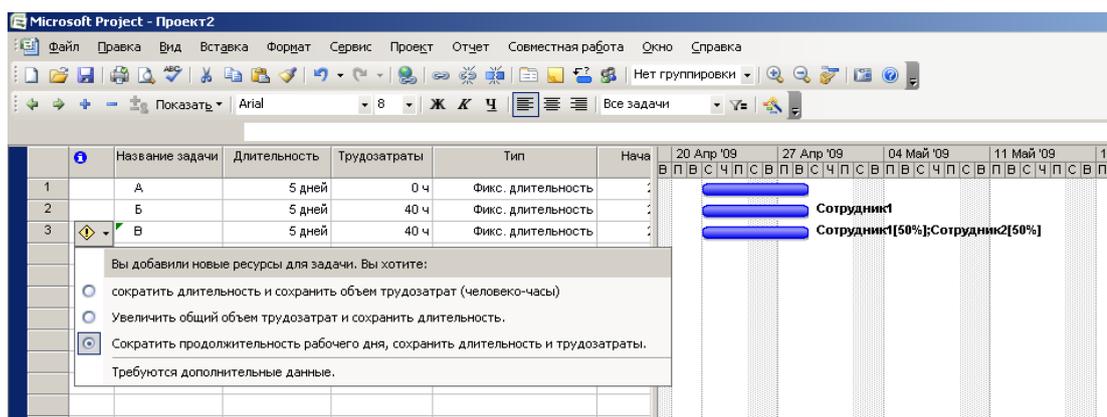


Задачи в плане проекта могут быть трех типов: **Фиксированная длительность**, **Фиксированные трудозатраты** или **Фиксированный объем ресурсов**. Тип задачи определяет, как редактирование одного из свойств задачи — длительности, трудозатрат или назначений — будет влиять на два других свойства.

От того, какой тип задачи выбран, зависит, значение какого из трех свойств фиксируется. Например, если вы определите тип задачи Фиксированная длительность, то изменение трудозатрат или числа назначенных на исполнение задачи сотрудников не изменит ее длительность.

Тип: Фиксированная длительность

Если вы создадите задачу с фиксированной длительностью и определите ее длительность в 5 дней, а затем назначите на исполнение сотрудника, то трудозатраты автоматически будут установлены в 40 часов. После того как вы добавите на задачу еще одного сотрудника, то программное обеспечение посчитает, что Вы хотите разгрузить первого и поделит трудозатраты 40 часов между двумя сотрудниками, а длительность по-прежнему будет равняться 5 дням. Если необходимо внести коррективы, воспользуйтесь кнопкой, которая позволяет изменить параметры загрузки ресурсов.



Если же увеличить длительность задачи до 6 дней, то трудозатраты каждого из задействованных в ней сотрудников возрастут, и суммарно составят уже 48 часов.

Тип Фиксированные трудозатраты

Тип Фиксированные трудозатраты используется в плане проекта для обозначения задач, для выполнения которых нужны определенные трудозатраты. Длительность исполнения таких задач напрямую зависит от числа выделенных на исполнение сотрудников. Чем больше сотрудников, тем меньший объем работы приходится на каждого из них и тем быстрее задача будет выполнена.

Например, если добавить в план задачу с длительностью в 5 дней, установить ее тип Фиксированные трудозатраты и назначить сотрудника с загрузкой 100%, то зафиксирует объем трудозатрат в 40 часов. Если теперь назначить на задачу еще одного сотрудника, то программа разделит зафиксированные трудозатраты в 40 часов между двумя из них, и, соответственно, время, за которое они вдвоем выполнят задачу (то есть ее длительность), уменьшится до 2,5 дня.

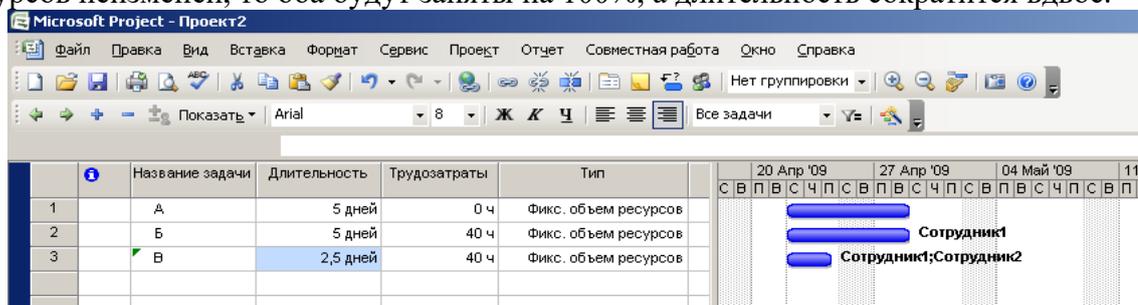


Если же у задачи с фиксированными трудозатратами увеличить длительность, то есть дать назначенным ресурсам больше времени на выполнение работы, то программа уменьшит их среднюю ежедневную нагрузку.

Тип Фиксированный объем ресурсов

Тип Фиксированный объем ресурсов используется в плане проекта для обозначения задач, длительность и объем работ которых полностью зависят от выделенных на них ресурсов.

Например, если добавить в план задачу длительностью в 5 дней и выделить на нее сотрудника, то трудозатраты будут определены в 40 часов. Если теперь назначить на задачу еще одного сотрудника, то трудозатраты останутся 40 часов, но так как объем ресурсов неизменен, то оба будут заняты на 100%, а длительность сократится вдвое.



Взаимосвязь свойств для задач разных типов

Тип задачи	Изменение объема ресурсов	Изменение длительности приводит к пересчету	Изменение работы
Фиксированный объем ресурсов	длительности	трудозатрат	длительности
Фиксированные трудозатраты	длительности	объема ресурсов	длительности
Фиксированная длительность	трудозатрат	трудозатрат	объема ресурсов

Фиксированный объем работ

В дополнение к указанию типа задачи можно использовать признак фиксированного объема работ. Этот признак можно добавить, установив флажок Фиксированный объем работ рядом со списком типов задач, если задача не относится к типу Фиксированные трудозатраты.

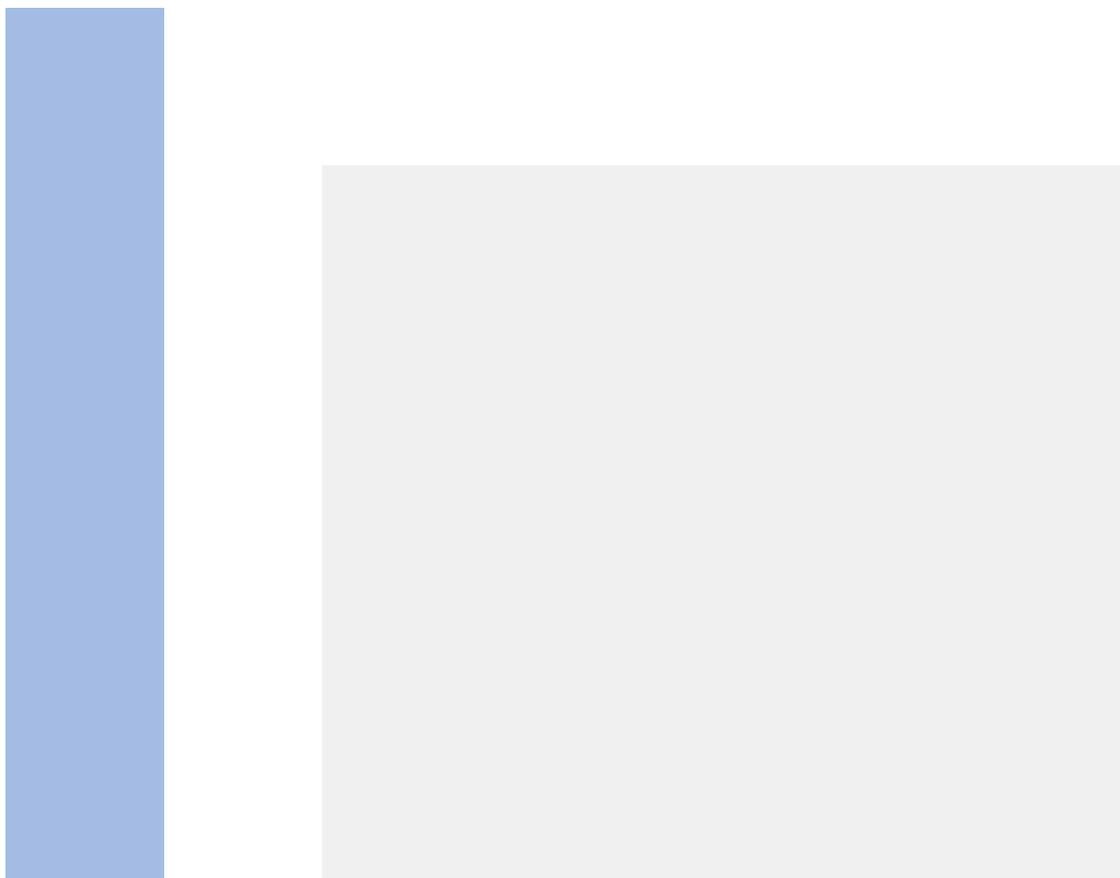
Если этот признак включен, то назначение ресурсов или удаление назначений приводит к изменению длительности или процента загрузки ресурсов, но не трудозатрат, необходимых для выполнения задачи. Таким образом, использование этого признака позволяет частично зафиксировать трудозатраты одновременно с одним из двух других свойств задачи: длительностью или объемом ресурсов.

Фиксация объема работ не учитывается при первом назначении ресурсов на задачу и влияет на логику работы только после первого назначения. Кроме того, признак фиксации объема работ не учитывается, когда вы изменяете длительность, трудозатраты или единицы уже назначенных ресурсов.

Например, когда на задачу фиксированной длительности 5 дней назначался второй сотрудник, трудозатраты увеличивались с 40 часов до 80. Если же задача будет фиксированной длительности и фиксированного объема работ, то добавление второго сотрудника не повлияет на трудозатраты, а приведет к понижению загрузки первого сотрудника до 50%, и второй сотрудник будет также задействован на 50%.

Если же добавить второй ресурс к задаче с фиксированными ресурсами, то трудозатраты вырастают с 40 до 80 часов. Если в аналогичной ситуации задача будет помечена как задача с фиксированным объемом работ, то при добавлении ресурса трудозатраты сохранятся, а длительность задачи уменьшится, поскольку участие второго ресурса уменьшает время, необходимое на выполнение объема работ.

Назначьте ресурсы задачам как в примере:



Практическая работа №6

Анализ и оптимизация плана проекта. Формы и отчеты проектов

Продолжаем работать с проектом «Ремонт офиса», созданного в практической работе №5.

Анализ и оптимизация плана проекта

План нужно проанализировать в нескольких аспектах. Во-первых, необходимо убедиться в соответствии расписания потребностям: ведь в процессе определения назначений длительности задач могли измениться. Во-вторых, необходимо проверить соответствие загрузки ресурсов: в процессе выделения ресурсов мы могли перегрузить некоторых из них. В-третьих, нужно проверить соответствие общей стоимости проекта, определившейся после создания назначений, нашим ожиданиям: в процессе назначения ресурсов мы могли назначить на задачи слишком много дорогостоящих ресурсов и тем самым превысить ожидаемую стоимость. И наконец, нужно оценить риски выполнения проекта: насколько велика вероятность не уложиться в расписание, не выполнить все

поставленные задачи и не уложиться в бюджет. Если в процессе анализа обнаруживаются проблемы, необходимо избавляться от них, оптимизируя план соответствующим образом.

Анализ не обязательно проводить в той последовательности, в которой его аспекты были перечислены. Обычно анализ начинается с аспекта, наиболее актуального для проекта. Мы рекомендуем начинать его с проверки загрузки ресурсов, поскольку риск срыва работ по плану, в котором изначально заложена работа сотрудников сверх нормы, очень высок.

Анализ и выравнивание загрузки ресурсов

Чтобы определить равномерность загрузки ресурсов, нужно открыть представление Лист ресурсов. В нем все ресурсы, загрузка которых превышает их доступность, выделены красным цветом, а в колонке Индикаторы рядом с их названиями отображается специальный значок.

Превышение доступности ресурса заключается в том, что для выполнения назначенной работы ресурсу требуется больше времени, чем у него есть. Существует несколько причин, способных привести к этому. Самой распространенной среди них является назначение ресурса на задачи, исполнение которых полностью или частично осуществляется одновременно. Другим вариантом может быть увеличение объема работ задачи, приведшее к превышению допустимого уровня загрузки ресурса. Наконец, назначение ресурса из-за изменений в плане может приходиться на дни, когда ресурс недоступен.

Выровнять загрузку ресурсов можно несколькими способами. Во-первых, уменьшив объем работы перегруженных ресурсов, сократив некоторые задачи или назначив других сотрудников на их выполнение. Во-вторых, избавившись от пересечения задач, вставив в расписание перерывы в задачах или назначениях либо изменив даты их начала и окончания. Наконец, учтя работу, выполняемую ресурсом сверх нормы, как сверхурочную.

Для выравнивания загрузки ресурсов можно воспользоваться автоматизированными средствами, а можно перераспределить загрузку вручную. Как правило, используются оба способа, поскольку команда автоматизированного выравнивания использует только второй из перечисленных методов выравнивания и поэтому обычно не может выровнять загрузку всех ресурсов.

Автоматическое выравнивание загрузки ресурсов

Диалоговое окно выравнивания загрузки ресурсов открывается с помощью команды меню Сервис > Выравнивание загрузки ресурсов. В разделе Вычисления для выравнивания определяются общие параметры выравнивания загрузки. Переключатели Выполнять автоматически и Выполнять вручную определяют, как будет осуществляться выравнивание: непосредственно при создании назначений (первый вариант) или при нажатии кнопки Выровнять в этом диалоговом окне (второй).

Ручное выравнивание ресурсов

Ручное выравнивание ресурсов осуществляется в два этапа. Сначала нужно найти те задачи, назначение на которые перегружает ресурсы. Затем нужно определить, как избавиться от перегрузки, поскольку вариантов довольно много. Можно перенести задачу,

прервать ее или изменить ее длительность. Можно уменьшить объем работы для ресурса или удалить назначение, причем как выделив на задачу другого сотрудника взамен перегруженного, так и не сделав этого. В таком случае трудозатраты задачи уменьшатся. Наконец, можно сохранить перегрузку, перенеся избыточные трудозатраты ресурса в сверхурочные.

Поиск перегружающих задач

Для поиска задач, участие в которых перегружает ресурсы, стоит воспользоваться представлением Использование ресурсов. В представлении нужно применить фильтр Превышение доступности ресурсов, чтобы отобразить только перегруженные ресурсы.

На диаграмме в строке напротив названия ресурса указывается его суммарная загрузка за минимальный отрезок на шкале времени. Данные за те дни, когда суммарная загрузка превышает допустимую, выделены красным. Поэтому, пролистывая диаграмму, можно быстро найти дни, когда ресурс перегружен. В столбце, относящемся к этому дню, отображается распределение загрузки между задачами, и данные задачи, перегружающей ресурс, тоже выделены красным.

Представления. Формы. Отчеты

Сетевой график и критический путь

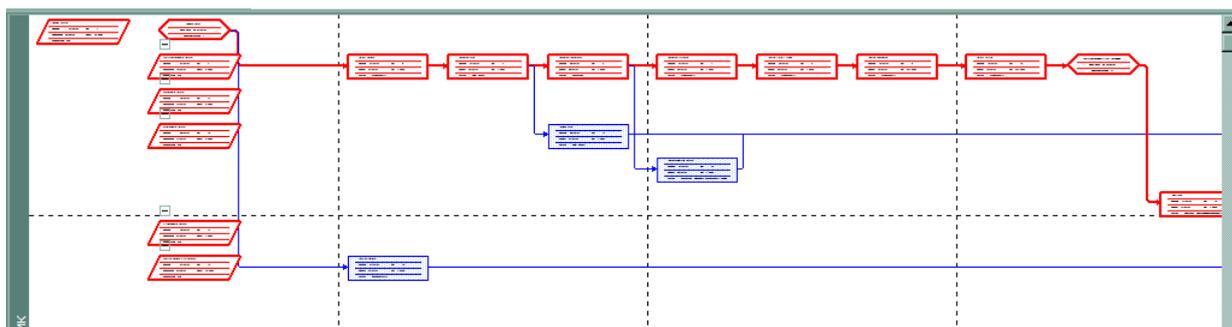
Красным цветом обозначены критические работы – если их длительность возрастет, возрастет длительность всего проекта.

Для сокращения критического пути можно:

- Уменьшать длительности отдельных задач (прогрессивные технологии, увеличение количества ресурсов)
- Выполнять задачи параллельно

Параллелограммами обозначены суммарные задачи – только они могут не иметь последователя или предшественника.

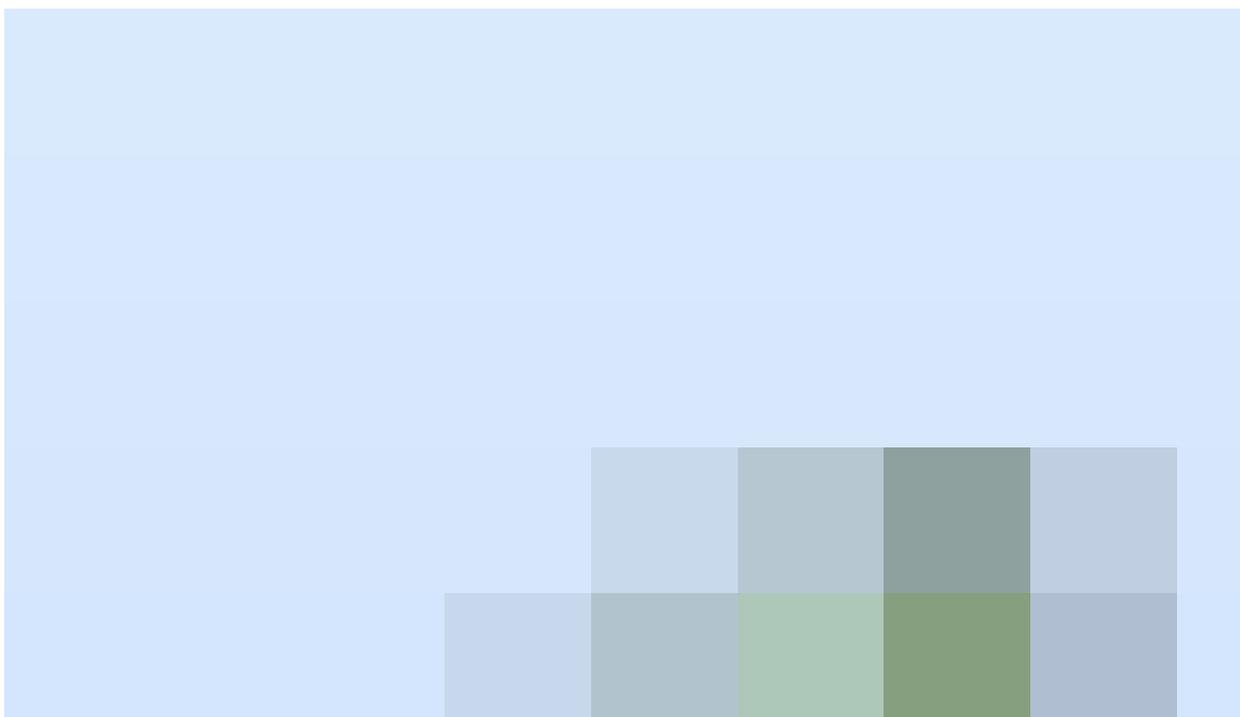
Открыть сетевой график настроить максимально маленький масштаб



Использование задач

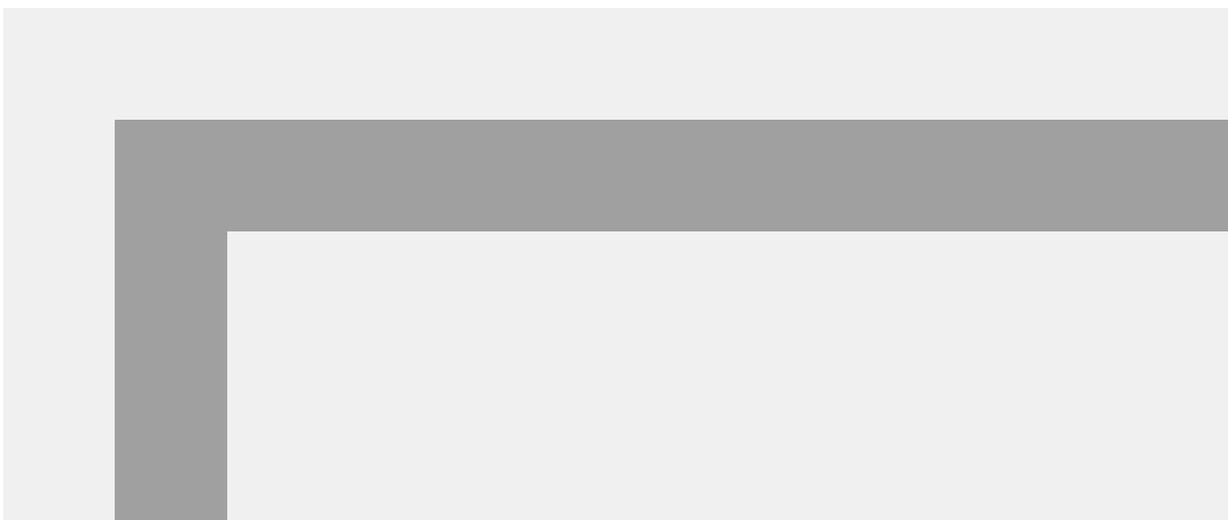
Получить информацию о предварительном бюджете. Вид... Использование задач.

Настроить распечатку затрат, как в примере.



Представление позволяет получать повременные данные по затратам и трудозатратам.
Календарь

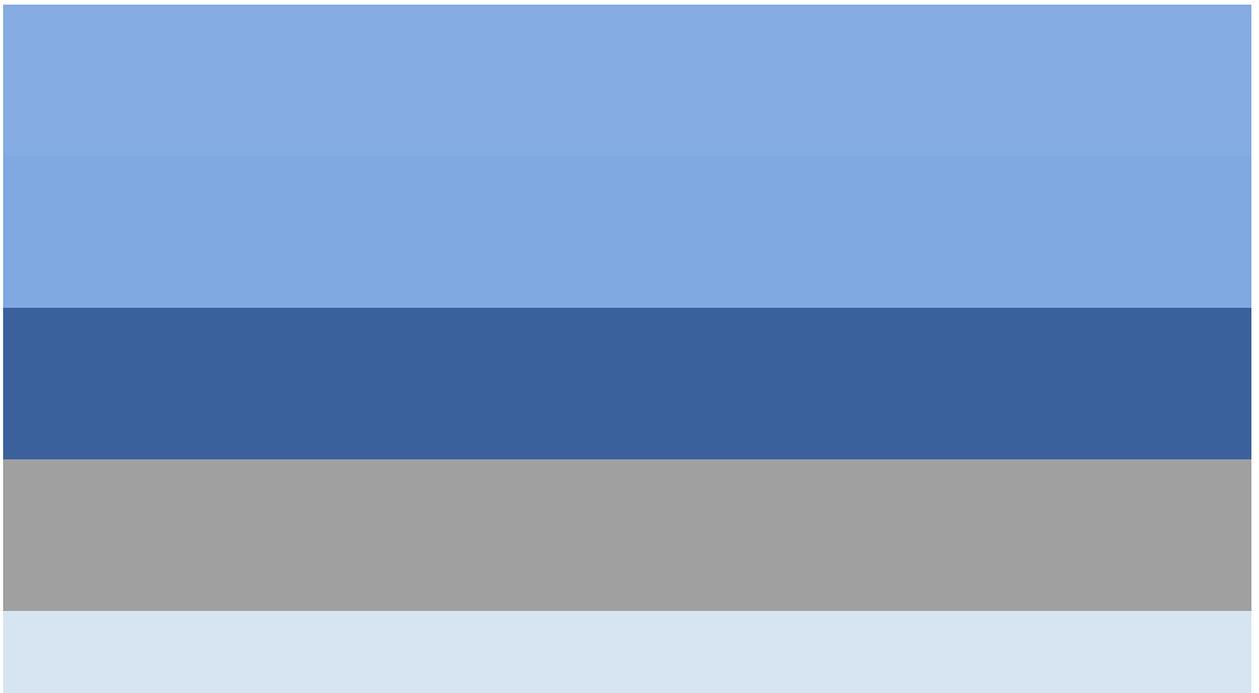
Открыть Календарь, настроить масштаб 6 недель на листе



Использование ресурсов

Красным отмечены работники и трудозатраты, которые имеют перегрузку. Важно легко уметь изменять масштаб и шкалу времени.

Открыть Использование ресурсов, настроить масштаб как в примере



Повременные данные по затратам и трудозатратам проекта по каждому ресурсу.

График ресурсов

В левой части представления выбирается ресурс, в правой части можно увидеть его занятость. Если нагрузка ресурса не превышает его максимальную единицу, то столбцы представления закрашены синим цветом, в противном случае – красным.

Открыть График ресурсов, настроить масштаб и шкалу времени как в примере



Комбинированные экранные формы

Для того, чтобы иметь более детальную информацию для принятия решения о загрузке работников удобно не только видеть дни перегрузки, но и какими конкретно работами человек в этот день занят.

Для этого нужно разделить окно на две части, в верхней открыть График ресурсов, в нижней диаграмму Ганта.

Настроить комбинированную экранную форму, как в примере



Настройка диаграммы Ганта

Диаграмма Ганта – самое используемое представление. Ее можно настроить так, как удобно пользователю. Меню Формат... Мастер диаграмм Ганта.

Настроить диаграмму так, чтобы тонкой линией был обозначен резерв времени для каждой задачи, справа печатались названия ресурсов, а у суммарных задач – затраты.

Отчеты

Отчет – это формат представления проектных данных, предназначенный для распечатки. В программу входит набор предопределенных отчетов, которые можно использовать в готовом виде или настроить меню Вид...Отчеты (для 2003 версии программы или Отчет...Отчеты для 2007).

Настроить следующие отчеты:

- Отчет, из которого видно, сколько денег требует каждая задача в каждый из дней проекта и итоговая ежедневная сумма по проекту в целом.
- Задание работникам, из которого видно в какой день на каких задачах и сколько часов он должен работать. (Отчет можно разрезать на полоски и раздать работникам)

Снятие копии с экрана

Для наглядного представления данных программы можно ту информацию, которая представлена на экране, сохранить в формате рисунка (кнопка на панели инструментов «Копировать рисунок») отправить по E-mail или перенести в Word, снабдить пояснениями и распечатать.

Фильтрация

При работе с проектами, использующими большое число задач и ресурсов, часто требуется отображать в таблицах не все доступные данные, а только некоторую их часть.

В том случае, если используется иерархическая структура проекта и часть задач нижнего уровня скрыта, фильтр будет использовать только открытые задачи. Поэтому, если требуется отфильтровать задачи всего проекта полностью, необходимо сначала использовать кнопку панели инструментов **Показать... Все подзадачи**. Самый простой способ настроить фильтрацию – использовать меню **Проект...Фильтр... Автофильтр**

Используя Автофильтр настроить выборку только тех задач, которые используют ресурс Менеджер



Группировка

Как правило, задачи проекта уже сгруппированы по этапам проекта при использовании иерархической структуры. Но очень часто в процессе работы требуется сгруппировать задачи или ресурсы иным способом. Меню Проект... Группировка... Настройка группировки... Например для контроля исполнения работ удобно настроить знать какие работы должны закончиться в данную неделю.

Сгруппировать задачи проекта по неделям по дате окончания



Базовый план проекта

Предварительный план проекта составлен. После того, как он утвержден руководством, его следует зафиксировать для дальнейшей сверки.

Сформировать базовый плана проекта - **Сервис... Отслеживание... Задать базовый план**

В представлении Диаграмма Ганта с отслеживанием задачи должны разбиться на 2 части: базовый план и фактическое состояние. Отслеживание удобнее вести с помощью этого представления. После того, как началось выполнение запланированных работ, руководитель проекта должен внимательно следить за ходом работ, чтобы вовремя заметить несоответствие между планом и фактическим выполнением работ. Для этого необходимо собирать данные о ходе выполнения работ, заносить эти данные в программу и сравнивать их с базовым планом.

Отслеживание проекта

После того, как проект начался, истек первый отчетный период и получен первый отчет, необходимо внести информацию о ходе выполнения и результатах задач.

Задать дату отчета – **Проект... Сведения о проекте...Дата отчета** – начало второй недели проекта.

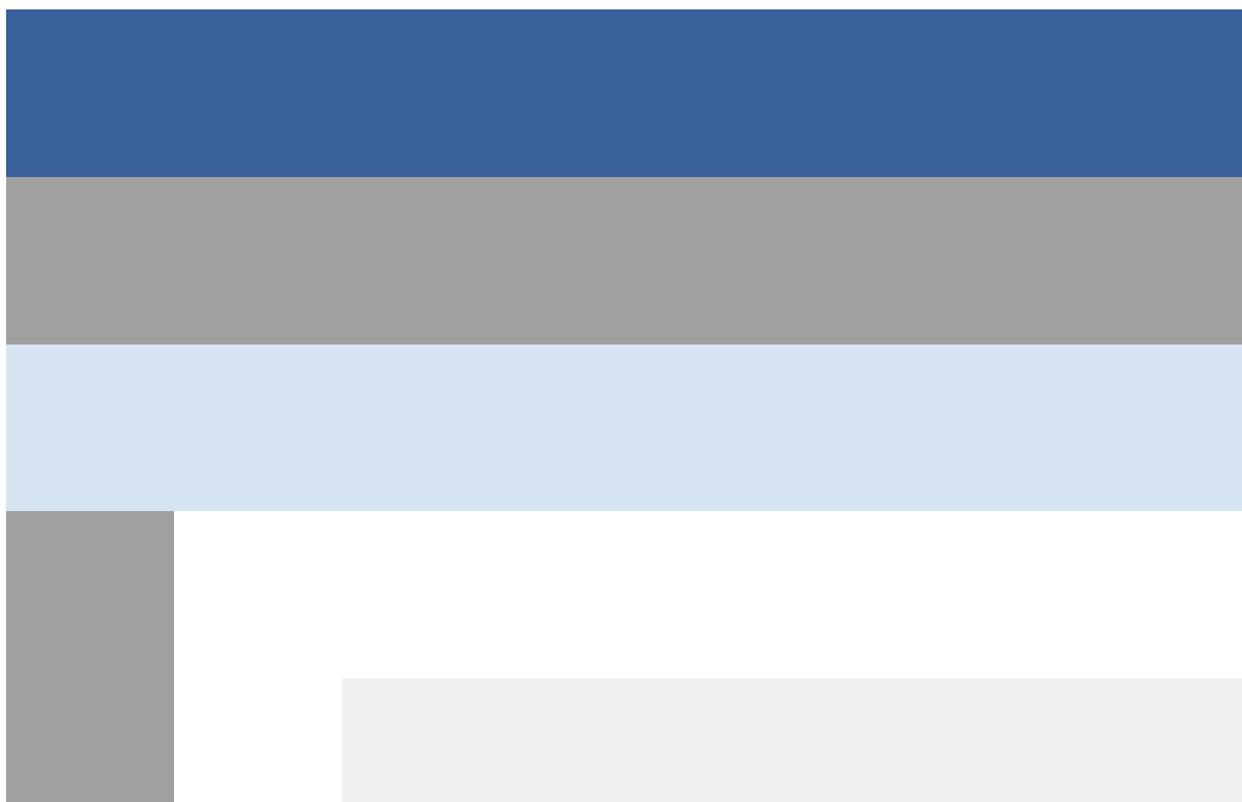
Ввод выполнения задач

Если необходимо отметить группу задач, которые выполнены по плану, их выделяют, **Сервис...Отслеживание... Обновить задачи/проект**. При такой работе удобно пользоваться группировками По началу и По окончанию. При неплановом выполнении задач вставляют дополнительные столбцы: Фактическая дата начала, Фактическая дата завершения, % выполнения и заполняют их.

Отслеживание можно вести с помощью специальной таблицы: **Вид...Таблица... Отслеживание**.

Удобно пользоваться панелью инструментов **Отслеживание: Вид...Панели инструментов...Отслеживание**.

Заполнить столбец % завершения как в примере, изменить фактическую дату начала некоторых задач и оценить результат на диаграмме Ганта.



Вывод линий хода выполнения задач

Линии хода выполнения задач наглядно показывают те задачи, которые задерживаются и те, которые опережают график. **Сервис... Отслеживание... Линии хода выполнения... Всегда показывать линии выполнения... На дату отчета**.

Выберите команду **Показывать линии хода выполнения на основе ...Базового плана**.

Настроить линию хода выполнения проекта как в примере

После того как проект завершен, можно оценить полученные результаты: **Вид... Таблица...Отклонения**. Откройте таблицу и найдите задачи, из-за которых произошло отклонение даты окончания всего проекта. В эту же таблицу можно добавить столбец Отклонение по стоимости и увидеть задачи, которые привели к увеличению стоимости проекта.

Ввод фактических данных позволит видеть пользователю отклонения фактических данных от базовых значений, если эти отклонения превышают допустимые, использовать данную модель для апробирования корректирующих воздействий.

Корректирующие воздействия

Если проект начинает отставать от графика возможны три альтернативных решения проблемы:

1. Оценка объема оставшихся работ и принятии решения о том, можно ли потерянное время наверстать на последующих шагах.

2. Если это неосуществимо, рассматривается возможность дополнительного вознаграждения для завершения проекта вовремя. Дополнительное вознаграждение может быть оправдано, если будет выполнено сравнение этих расходов с потенциальными потерями из-за превышения сроков проекта.

3. Наконец, следует рассмотреть возможность привлечения дополнительных ресурсов, это так же приводит к увеличению затрат на проект, и так же может компенсировать срывы сроков.

Если затраты по проекту начинают превышать бюджет, рассматривается оставшаяся работа и возможность компенсации перерасхода за счет будущих работ. Если это невозможно, рассматривается возможность сокращения масштабов проекта или получения дополнительного финансирования от заказчика.

Практическая работа №7

Формирование бизнес-цели проекта. Разработка устава проекта.

Цель работы: Изучение основных понятий: бизнес-цель, устав проекта, технико-экономическое обоснование проекта. Формирование практических умений и навыков формулирования цели организации; разработки устава проекта, технического задания ИС.

В результате выполнения практической работы студент должен:

знать:

- решения задач обработки информации;
- основные процессы управления проектом разработки.

уметь:

- создавать проект по разработке приложения и формулировать его задачи;
- выполнять управление проектом с использованием инструментальных средств.

Задание для практической работы:

1. Изучить теоретическую часть
2. Выполнить практическую часть
3. Ответить на контрольные вопросы
4. Оформить отчет.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

Бизнес-цель - это описание фактора, побуждающего к выполнению проекта. Ее формирование производится на стратегическом уровне, то есть бизнес-цель выступает в качестве связующего звена между глобальными задачами, стоящими перед

организациями, и планируемым к реализации проектом. При отходе от стратегического видения происходит смещение бизнес-цели в сторону тактических и даже операционных задач, на уровне которых целью проекта видится "просто выдать продукт", а не достичь какой-либо тактической цели, поддерживающей стратегические цели организации. Этого нельзя допускать: бизнес-цель проекта должна всегда носить тактический или стратегический характер, но в то же время быть предельно точной и ясной (очень редко удается применить широко известный метод SMART к построению бизнес-цели проекта).

Так, например, бизнес-целью проекта по приобретению и установке нового производственного оборудования является не покупка и установка оборудования, а устранение узкого места в производственном процессе и обеспечение надлежащих объемов выпуска, гарантирующих удовлетворение спроса и завоевание определенной доли рынка. Аналогично, проект внедрения информационной системы имеет своей бизнес-целью не разворачивание технических средств, а создание информационно-технологического фундамента для поддержки принятия руководством компании своевременных управленческих решений, направленных на обеспечение ее развития и роста.

Устав проекта - это инструмент, который формально авторизует проект и является звеном, соединяющим предстоящий проект с текущей работой организации. Данный документ обычно отражает ситуацию со стороны организации-заказчика, выпускается руководителем, внешним по отношению к проекту, и назначает менеджера проекта, наделяя его полномочиями на использование в проекте ресурсов организации. Это особенно актуально в функционально-ориентированных и матричных организациях, т.е. в тех компаниях, где менеджеры не имеют непосредственной власти над членами проектной команды и другими ресурсами, но несут ответственность за выполнение проекта. Для того чтобы устав имел силу в подобной ситуации, издающий его руководитель, или спонсор проекта, должен находиться на том уровне, который подразумевает наличие контроля над ресурсами. Часто датой начала проекта считается день, следующий за подписанием устава.

Играя роль документа, формально авторизирующего задачу, устав включает в свой состав базовые требования и основные ожидания заинтересованных сторон. Этот документ выполняет несколько функций, среди них важно отметить:

- функцию постановки задачи;
- функцию согласования;
- авторизационную функцию;
- функцию повышения дисциплины;
- консолидационную функцию;
- интеграционную функцию.

Разработка устава проекта начинается после издания приказа о запуске. Распорядительная часть документа формально фиксирует дату старта проектной реализации, в ней вводится его полное и краткое название, назначаются куратор, руководитель (PM), ответственные лица за ключевые блоки. Структурная схема устава приводится далее. Он разрабатывается итерационно и может иметь несколько редакций, постепенно уточняющих основные положения, которые включают следующие аспекты.

1. Обоснование выполнения уникальной задачи развития.
2. Цели, задачи и результаты.
3. Имя и фамилию PM, границы его ответственности и полномочия.
4. Определение и структуру продукта.
5. Интересы и ожидания участников.
6. Критерии успеха.

7. Принципы организации и управления проектом.

УСТАВ ПРОЕКТА

Таблица 1. Требования к уставу проекта

№	Раздел	Пояснения
1.	Название проекта	Каждый проект должен иметь название, отражающее его суть и в то же время достаточно яркое для привлечения внимания
2.	Бизнес-причина возникновения проекта	Производственная необходимость, или самое общее описание проекта и требований к продукту, производство которого является результатом выполнения проекта. Формулировка причины фактически дает ответ на вопрос, зачем выполняется данный проект. Причины возникновения проекта могут основываться на требованиях рынка, техническом прогрессе, юридических требованиях или государственном стандарте
3.	Бизнес-цель	Сформулирована заказчиком, исходя из стратегических и тактических целей компании.
4.	Требования, удовлетворяющие потребности, пожелания и ожидания заказчика, спонсора и других участников проекта	Видение организацией-заказчиком, как правило, высокоуровневое, способов достижения поставленной бизнес-цели или решения существующей проблемы. Проект считается успешным, если ожидания заказчика и участников проекта оказались выполненными, следовательно, к моменту формирования устава проекта его участники должны быть идентифицированы. Все задокументированные в уставе требования должны быть учтены при выполнении стоимостной оценки проекта
5.	Расписание основных контрольных событий	На этапе формирования устава должно быть обязательно указано время начала и завершения проекта; при необходимости отмечаются ключевые <i>вехи проекта</i> , принципиальные для организации-заказчика. Вообще рекомендуется ограничить количество <i>контрольных событий</i> теми, которые абсолютно необходимы, т.е. обычно тремя-пятью. Иными словами, принимая во внимание цель устава и соответствующий уровень детализации, совершенно излишне разрабатывать длинный список событий - это только создаст дополнительные ограничения для выбора методологии реализации проекта. Кроме того, организации, придающие значение себестоимости, имеют тенденцию указывать для основных событий специфику бюджета ресурсов или бюджета средств
6.	Участники проекта	Перечисление заинтересованных сторон проекта, иными словами, круга лиц и организаций, на которых оказывает воздействие реализация данного проекта и которые сами могут воздействовать на него.
7.	Окружение проекта	Перечисление всех организационных факторов, характеризующих обстановку вокруг проекта и на рынке. Также необходимо указать благоприятные и неблагоприятные особенности среды, в которой проект будет выполняться (внутри и вне компании), и способность организации-исполнителя к его

№	Раздел	Пояснения
		<p>осуществлению, а организации-заказчика - к использованию его результатов. Далее будет показан один из эффективных способов выполнения комплексного анализа окружения и участников проекта. При использовании этого подхода сначала определяется достаточно большое число факторов, <i>действующих в окружении проекта</i>; они заносятся в соответствующий сектор. Затем выделяются наиболее критичные из них (прямоугольники - участники, овалы - факторы окружения)</p>
8.	<p>Допущения относительно организации и окружения, а также внешние допущения</p>	<p>Набор условий, которые должны быть выполнены наряду с созданием продукта проекта, для достижения результата проекта. Допущения обуславливают риски проекта; во время проекта происходит их мониторинг. Пример допущений:</p> <ul style="list-style-type: none"> - компетенции команды проекта достаточно для выполнения предпроектного обследования; - организацией-заказчиком будет выделен персонал для выполнения работ по поддержке проекта. <p>Обратите внимание, что при составлении устава проекта допущения формулируются со стороны организации-заказчика об организации-исполнителе</p>
9.	<p>Ограничения относительно организации и окружения, а также внешние ограничения</p>	<p>Ограничение указывает на условие, которое нельзя нарушать в процессе создания продукта проекта, или условие, которому ни при каких обстоятельствах не должен удовлетворять продукт проекта. Ограничения к тому же указывают на возможности команды проекта по выбору вариантов для выполнения любых проектных работ. Пример ограничений проекта:</p> <ul style="list-style-type: none"> - увеличение стоимости проекта не более чем на 10%; - не менее 40% членов команды проекта, предоставляемых исполнителем, заняты на 100% в проекте. <p>Обратите внимание, что при составлении устава проекта ограничения формулируются со стороны организации-заказчика об организации-исполнителе и о проекте в целом</p>
10.	<p>Объем денежных средств, выделенных на достижение бизнес-цели</p>	<p>На данном этапе указывается сумма средств, которую организация-заказчик готова выделить на достижение сформулированной <i>бизнес-цели проекта</i>. Указанная сумма является результатом определения порядка величины и ошибка в оценке может составлять от ~ - 20% до +100%</p>
11.	<p>Назначение руководителей проекта и общее определение полномочий ключевых членов проектной команды: РП, спонсор, координатор</p>	<p>Руководитель проекта назначается уставом проекта и формально приступает к выполнению своих обязанностей на следующий день после подписания устава проекта. Руководитель, или менеджер, проекта несет основную ответственность за общее планирование, направление и контроль проекта в течение всех фаз его жизненного цикла, ставя целью получение желаемого результата в рамках утвержденного бюджета и расписания. Основная задача руководителя проекта - объединение усилий</p>

№	Раздел	Пояснения
		<p>всех лиц, участвующих в проекте. Для решения этой задачи менеджер проекта наделяется полномочиями по проекту, т.е. правом отдавать функциональным лидерам проекта распоряжения, необходимые для планирования, исполнения, мониторинга, оценивания и контроля работ, которые должны быть выполнены по данному проекту. Руководство проектом также включает в себя получение информации, необходимой для планирования, мониторинга, оценивания и контроля проекта. Роль спонсора проекта обычно берет на себя (не назначается!!!) менеджер высшего звена, который действует от лица руководства компании, финансирующей или исполняющей проект. Ключевая задача спонсора заключается в обеспечении ресурсов проекта, в том числе административных, а также в обеспечении связи между проектом и руководством организации-заказчика. На проекте спонсор является лицом, принимающим те решения, которые находятся за пределами полномочий руководителя проекта, например:</p> <ul style="list-style-type: none"> - утверждать бизнес-цели проекта, включая расписания и бюджет, и вносимые в них изменения; - назначать и утверждать менеджера проекта, а также утверждать соответствующую должностную инструкцию и порядок подчинения; - формировать стратегические указания для менеджера проекта по ходу отслеживания результатов проекта; - вносить и утверждать основные изменения по проекту и решения, касающиеся выделения ресурсов; - принимать решения о внесении изменений в базовую линию проекта. <p>Роль спонсора проекта обычно не предполагает работы с полной занятостью вне зависимости от размера проекта. Администратор (координатор) проекта - это специфическая функция на проекте, которая необходима для поддержки работ, связанных с администрированием и документированием функционирования проектной организации и обеспечением инфраструктуры проекта. Работа администратора имеет своей ключевой задачей поддержку руководителя проекта на операционном уровне с целью его высвобождения для интеллектуально-сложных задач. В обязанности координатора проекта может входить: администрирование проектных контрактов и договоров на протяжении всего ЖЦ, организация периодического сбора статуса выполнения проекта и т.п. сбор статуса - словосочетание, не несущее смысла, если только это не специфический термин. Формировать всю команду и тем более сразу указывать имена всех ее членов не принято - функциональные руководители обычно выделяют для проекта своих подчиненных, только когда руководитель проекта составит план потребности в ресурсах, после определения состава работ проекта, и отправит официальный запрос на ресурсы,</p>

№	Раздел	Пояснения
		утвержденный спонсором проекта

Рассмотрев пример, на основе своей предметной области (*смотри номер варианта*) разработать устав для выбранного направления.

ВАРИАНТЫ ЗАДАНИЙ

1. Прокат автомобилей.
2. Библиотечный фонд города.
3. Спортивный клуб.
4. Управление складом.
5. Автошкола.
6. Химчистка.
7. Автомастерская.
8. Компания по продаже медицинской техники.
9. Страховая компания.
10. Гостиница.
11. Банк.
12. Ломбард.
13. Оптовая база.
14. Завод по производству металлоизделий.
15. Ювелирная мастерская.
16. Предприятие по организации свадебных торжеств.
17. Бюро по трудоустройству.
18. Нотариальная контора.
19. Производство мебели.
20. Производство детских игрушек.
21. Поликлиника.
22. Магазин розничной торговли.
23. Спортивный клуб.
24. Аэропорт.
25. Магазин по ремонту и продаже компьютеров и комплектующих.
26. Строительная организация.
27. Игровая комната.
28. Строительная организация.
29. Фотоцентр.
30. Городской зоопарк.

Контрольные вопросы:

1. Что такое проект? Управление проектами?
2. Назовите основные факторы, влияющие на проект?
3. Что такое бизнес-цель проекта? Устав проекта?
4. Назовите основных участников проекта и их полномочия?
5. Перечислите требования, предъявляемые к уставу проекта?

Практическая работа № 8

Подготовка проектных документов: плана и паспорта

Цель работы: Изучить основные аспекты разработки плана и паспорта проекта.

В результате выполнения практических заданий обучающийся должен **иметь практический опыт:**

- использования инструментальных средств обработки информации;
- формирования отчетной документации по результатам работ;
- использования стандартов при оформлении программной документации;

В результате выполнения практических заданий обучающийся должен **уметь:**

- осуществлять математическую и информационную постановку задач по обработке информации,
- создавать проект по разработке приложения и формулировать его задачи, выполнять управление проектом с использованием инструментальных средств;

В результате выполнения практических заданий обучающийся должен **знать:**

- основные процессы управления проектом разработки.

Задание для практической работы:

1. Изучить теоретическую часть
2. Выполнить практическую часть
3. Ответить на контрольные вопросы
4. Оформить отчет.

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

1. Паспорт проекта

Паспорт (устав) проекта - основополагающий документ проекта. Разработка паспорта проекта - процесс, который относится к группе процессов инициации проекта. В разработке первоначального содержания и паспорта проекта важную роль играет инвестор проекта.

В паспорте проекта документируются первоначальные требования к проекту, удовлетворяющие потребностям и ожиданиям заинтересованных сторон.

Паспорт проекта является завершением этапа инициации и началом этапа планирования проекта. Паспорт проекта предназначен для описания проекта и необходим для управления им.

Паспорт проекта готовит куратор проекта совместно с инициатором, руководителем и при необходимости представителем заказчика проекта.

Форма паспорта проекта не подлежит корректировке при заполнении.

Паспорт проекта оформляется в нескольких экземплярах, которые хранятся у руководителя проекта, куратора проекта и председателя экспертной комиссии по рассмотрению проектов при органе исполнительной власти, государственном органе области.

Базовая (первая) версия паспорта проекта утверждается ответственным за его утверждение лицом и является признанием того, что работы по проекту могут быть начаты. Все изменения, касающиеся целей проекта, согласовываются с заказчиком (и/или) инвестором проекта и обязательно вносятся в паспорт проекта.

Паспорт проекта документирует бизнес-потребности, текущее понимание потребностей заказчика, а также новый продукт, услугу или результат, который планируется создать, например:

- назначение или обоснование проекта;
- измеримые цели проекта и соответствующие критерии успеха;
- требования высокого уровня;
- описание проекта высокого уровня;
- риски высокого уровня;

- сводное расписание контрольных событий;
- сводный бюджет;
- требования к одобрению проекта (что составляет успех проекта, кто решает, что проект оказался успешным, и кто подписывает проект);
- назначенный менеджер проекта, уровень ответственности и полномочий;
- имя и полномочия спонсора или другого лица (лиц), утверждающего устав проекта

За подготовку и внесение изменений в паспорт проекта в компании необходимо назначить ответственное лицо.

Изменения в паспорт проекта должны вноситься в следующих случаях:

- произошли изменения в параметрах проекта (сроки, содержание, бюджет, качество);
- произошли изменения в требованиях заказчика (и/или) инвестора проекта;
- произошли изменения в параметрах продукта проекта;
- произошли изменения во внутренних/внешних условиях проекта.

Изменения в паспорт проекта должны быть оформлены соответствующим образом, а внесение изменений в паспорт проекта должны быть санкционированы руководителем проекта.

В случае возникновения необходимости внесения корректировок в подписанный всеми сторонами паспорт проекта изменения оформляются отдельным документом – «Ведомостью изменений», который после утверждения будет являться неотъемлемой частью паспорта проекта.

По завершении проекта паспорт проекта и все изменения к нему должны быть включены в архив проекта. После завершения проекта, которое фиксируется приказом, изменения в паспорте проекта запрещены

По завершении проекта все версии паспорта проекта должны быть включены в архив проекта.

Для разработки паспорта проекта используется метод экспертной оценки.

Метод экспертной оценки широко применяется в менеджменте и представляет собой оценку чего-либо (проблемы, документа, ситуации, объекта и т.д.) группой экспертов.

Цель метода экспертной оценки - получить мнение нескольких специалистов сразу и выработать решение на основе их мнения. Мнение группы считается менее субъективным и более точным, чем мнение одно человека.

Эксперт - человек, обладающий специальными знаниями, навыками, опытом. В качестве экспертов могут выступать ключевые сотрудники организации, представляющие разные функциональные области, приглашенные эксперты, в качестве которых могут выступать консультанты, общественные и частные организации и т.д.

2. План управления проектом

Процесс разработки плана управления проектом есть процесс документации действий, необходимых для определения, подготовки, интеграции и координации всех вспомогательных планов. Корректно составленный план управления проектом является основным источником информации о том, как проект будет планироваться, оцениваться, контролироваться и закрываться. План управления проектом обновляется и редактируется в рамках процесса осуществления интегрированного управления изменениями проекта, для поддержки версионности документа рекомендуется использовать лист управления документом.

План управления проектом может быть либо резюмирующим, либо детализированным и состоять из одного или нескольких вспомогательных планов и прочих элементов.

План управления проектом рекомендуется разделять на 3 блока по характеру содержащейся в них информации.

1. Вспомогательные планы управления проектом, в число которых входят:

- план управления содержанием проекта;
- план управления расписанием проекта;
- план управления стоимостью проекта;
- план управления качеством проекта;
- план управления обеспечением персоналом;
- план управления коммуникациями проекта;
- план управления рисками проекта;
- план управления конфигурацией.

2. Базовая линия проекта, состоящая из:

- базового расписания проекта;
- базового плана по стоимости;
- базового плана по качеству;
- базового плана по конфигурации;
- реестра рисков.

3. Результаты анализа, проведенного проектной командой в отношении содержания, объема и сроков проекта.

Создавая планы – важно правильно к ним относиться:

План – это не «клятва», а «прогноз». Во многих сферах (в том числе и в сфере информационных технологий) невозможно совершенно точно прогнозировать продолжительность, а порой и состав работ. Может показаться, что это дискредитирует идею планирования, однако это не так. Помните – что нельзя спланировать, нельзя и сделать.

Планируйте с «диапазоном». Не выдавайте (и не требуйте) точную оценку там, где ее дать нельзя. Донесите до всех членов команды и экспертов, которые помогают вам планировать проект, что вы не просите с них клятву «уложиться к определенной дате», вам нужна реалистичная оценка и возможные отклонения от нее. Однако настаивайте на реалистичности оценок.

Опасайтесь раздувания оценок (padding). Член команды, эксперт или вы сами, поставленный в жесткие условия («выдать реалистичные оценки») испытывает соблазн зависеть свои прогнозы, чтобы подстраховаться. Чтобы противостоять padding, особенно общаясь с техническими экспертами, многократно превосходящими вас в своих инженерных компетенции – нужен очень серьезный навык общения и знание психологии людей.

Планы будут изменяться. На это необходимо ориентироваться сразу. В отличие от устава, единожды созданного и практически не подверженного изменениям, планы проекта – документы весьма «живые». По ходу выполнения работ, мы будем узнавать и выявлять новые нюансы, детали, столкнемся с форс-мажорами. Чтобы наши планы оставались достоверными – придется их корректировать.

ПРАКТИЧЕСКАЯ ЧАСТЬ

1. **Заполните паспорт своего проекта (см. Приложение 1).**

1.1. *Титульный лист паспорта проекта*

На титульном листе указывается наименование проекта. Наименование проекта необходимо начинать со слов, выражающих действие, далее указывается объект или сфера воздействия, завершать рекомендуется указанием территории реализации проекта, например:

- «Благоустройство склона по ул. Д. Бедного г. Валуйки»;
- «Внедрение программно-технического комплекса «Регистратура» в поликлинических учреждениях Белгородской области».

В поле «Идентификационный номер» титульного листа указывается номер, соответствующий Вашему порядковому номеру в классном журнале.

В поле «УТВЕРЖДАЮ (представитель координирующего органа)» указывается фамилия, имя и отчество преподавателя, ставится подпись преподавателя проекта и дата утверждения документа.

В поле «ПРИНЯТО К ИСПОЛНЕНИЮ (руководитель проекта, студент, разрабатывающий проект)» указывается группа, фамилия, имя, отчество студента, ставится его подпись и дата подготовки документа.

1.2. Общие сведения о документе

Данный раздел проекта не изменяется.

1.3. Группа управления проектом

Данный раздел содержит подробную информацию об исполнителе, координирующем органе и их представителях в проекте. Также необходимо указать данные относительно представителя заказчика в том случае, если он введен в состав группы управления проектом.

В графе «Название организации» таблицы указывается полное наименование организации (ОПК СТИ НИТУ «МИСиС»).

В графе «ФИО, должность, контактные данные представителя» таблицы указываются фамилия, имя и отчество принимающего проект преподавателя.

1.4. Цель и результат проекта, сроки реализации

Данный раздел содержит подробную информацию о цели, способе её достижения, результате и пользователях проекта.

В строке «Измеримая цель проекта» указывается цель, которую необходимо достичь по факту окончания проекта. Цель проекта – запланированное желаемое состояние объекта управления, она должна соответствовать следующим требованиям:

- отражать ожидаемый социально-экономический полезный эффект от реализации проекта;
- иметь измеримые количественные показатели и сроки достижения;
- быть достижимой в реальных условиях, в которых осуществляется проект;
- полностью находиться в сфере ответственности и влияния исполнителя проекта.

Пример: организовать переработку муки в объеме 29,5 тыс. тонн в год на территории Шебекинского района к концу 2016 года.

Цель проекта, утвержденная в паспорте проекта, не подлежит изменению на протяжении всех этапов жизненного цикла проекта.

В строке «Способ достижения цели» необходимо указать оптимальный путь достижения обозначенной цели, например: строительство и организация работы мельницы.

В строке «Результат проекта» в графе «Результат» указывается измеримое выражение социальных, экономических, интеллектуальных и иных эффектов, планируемых получить в результате реализации проекта (материальный объект, предоставленная услуга, нематериальный актив, знания и т.д.). Фактически результат - это подтверждение достижения цели.

Пример: организована работа мельницы по переработке муки с проектной мощностью не менее 29,5 тыс. тонн в год на территории Шебекинского района.

В графе «Вид подтверждения» обозначается документ, подтверждающий достижение указанного результата проекта.

В строке «Требования к результату проекта» в графе «Требование» указываются качественные и количественные характеристики результата, которые будут учитываться при его достижении (площадь объекта; количество смен работы; требования, указанные в техническом задании; наличие определенного вида оборудования; создание ряда объектов и т.д.). В графе «Вид подтверждения» обозначается документ, подтверждающий достижение указанного требования. Данные документы должны входить в состав документов, подтверждающих выполнение работ проекта.

Для экономических проектов обязательными требованиями к результату являются:

- количество созданных новых рабочих мест;
- размер среднемесячной заработной платы (необходимо руководствоваться установленным размером среднемесячной заработной платы в целом по области);
- размер налоговых поступлений в консолидированный бюджет области (в соответствии с утвержденными нормативами налоговой нагрузки на уровне региона).

В строке «Пользователи результатом проекта» указывается круг потребителей (область применения) результата проекта.

При необходимости паспорт проекта может быть дополнен следующими приложениями:

- технико-экономическое обоснование проекта;
- бизнес-план проекта;
- концепция проекта;

прочие материалы, которые дают более детальное представление о проекте

2. Заполните план управления проектом (см. Приложение 2).

План управления проектом - обязательный документ, наличие которого является завершением этапа планирования и началом этапа реализации проекта. Данный документ описывает плановые величины длительности, стоимости и объема работ. План управления проектом разрабатывается руководителем проекта совместно с рабочей группой, утверждается куратором проекта и председателем экспертной комиссии по рассмотрению проектов при органе исполнительной власти, государственном органе области.

Задачи и сроки исполнения, установленные в плане управления проектом, являются обязательными для исполнения руководителем проекта и всеми участниками рабочей группы. Форма плана управления проектом не подлежит корректировке при заполнении.

В случае возникновения необходимости внесения корректировок в подписанный всеми сторонами план управления проектом изменения оформляются отдельным документом – «Ведомостью изменений», который после утверждения будет являться неотъемлемой частью плана управления проектом.

2.1. Титульный лист плана управления проектом.

На титульном листе указывается наименование проекта. Наименование проекта необходимо начинать со слов, выражающих действие, далее указывается объект или сфера воздействия, завершать рекомендуется указанием территории реализации проекта, например:

- «Благоустройство склона по ул. Д. Бедного г. Валуйки»;
- «Внедрение программно-технического комплекса «Регистратура» в поликлинических учреждениях Белгородской области».

В поле «Идентификационный номер» титульного листа указывается номер, соответствующий Вашему порядковому номеру в классном журнале.

В поле «УТВЕРЖДАЮ (представитель координирующего органа)» указывается фамилия, имя и отчество преподавателя, ставится подпись преподавателя проекта и дата утверждения документа.

В поле «ПРИНЯТО К ИСПОЛНЕНИЮ (руководитель проекта, студент, разрабатывающий проект)» указывается группа, фамилия, имя, отчество студента, ставится его подпись и дата подготовки документа.

2.2. Общие сведения о документе

Данный раздел проекта не изменяется.

2.3. Календарный план-график работ по проекту.

Данный раздел предназначен для формализации мероприятий проекта, содержит перечень работ и процессов, которые планируется выполнить в рамках проекта.

Под кодом задач проекта (графа «Код работы/процесса. Код») понимается цифровое обозначение работ и процессов в порядке их следования в проекте (1, 2, 3, 4 и т.д.). Задача может состоять из подзадач, имеющих подчиненный уровень иерархии (1.1, 1.2, 2.1 и т.д.).

В графе «Код работы/процесса. Тип (Р/П)» обозначается принадлежность выполняемых действий в рамках проекта к процессу или работе. Для работ в данной графе необходимо отразить сокращение в виде «Р», для процессов – в виде «П».

В графе «Название работы/процесса» в хронологической последовательности перечисляются блоки работ с детализацией на работы и процессы нижнего уровня.

При выделении процессов в рамках проекта необходимо руководствоваться типовыми процессами, представленными в таблице 1.

Таблица 1 Типовые процессы проекта

Направление	Типовой процесс
Строительство и реконструкция	Экспертиза проектно-сметной документации
	Формирование пакета документов для разработки проектно-сметной документации
	Разработка технического задания
	Разработка проектно-сметной документации
	Формирование пакета документов для получения разрешения на строительство
	Контроль за исполнением строительно-монтажных работ
	Формирование пакета документов для получения разрешения на ввод объекта в эксплуатацию
	Приемка объекта в эксплуатацию
Работа с информационными системами	Передача объекта на баланс
	Контроль и систематическая актуализация сведений в существующих информационных системах
	Мониторинг работоспособности программного продукта/программно-аппаратного комплекса
	Устранение неисправностей и тестирование действующего программного продукта/программно-аппаратного комплекса
Имущественные и земельные отношения	Регистрация новых пользователей существующих систем
	Сопровождение и техническая поддержка информационной системы и оборудования
	Оформление договора/акта аренды, купли-продажи (безвозмездного пользования) на имущество или земельный участок
	Подготовка документации, организация и проведения аукциона (конкурса) на право заключения договора аренды/продажи (безвозмездного

Направление	Типовой процесс
	<p>пользования) на имущество или земельный участок</p> <p>Государственная регистрация права (сделки) на недвижимое имущество или земельный участок</p> <p>Постановка объекта недвижимости или земельного участка на государственный кадастровый учет</p> <p>Получение сведений из государственного кадастра недвижимости и Единого государственного реестра прав на недвижимое имущество и сделок с ним об объектах недвижимости и земельных участках</p> <p>Внесение изменений в государственный кадастр недвижимости и Единый государственный реестр прав на недвижимое имущество и сделок с ним</p> <p>Подготовка схемы расположения земельного участка на кадастровом плане территории</p> <p>Выполнение кадастровых работ</p> <p>Подбор земельного участка для строительства</p> <p>Инвентаризация имущества</p>
Контроль мониторинг, экспертиза и анализ	<p>Вычитка, экспертиза, редактирование, подготовка к визированию документов/представленной информации на соответствие установленным требованиям и нормативной правовой документации</p> <p>Контроль за исполнением качественных и количественных характеристик нормативных правовых актов</p> <p>Оценка динамики и качества показателей, численности; изменений объекта исследования</p> <p>Определение влияния отклонений от установленных требований/нормативных значений</p> <p>Оценка оптимальности расходов бюджета различных уровней</p> <p>Определение качества показателей</p> <p>Свод показателей</p> <p>Мониторинг показателей, состояния объекта, текущей ситуации, существующего опыта и практик и прочее</p> <p>Организация и проведение проверок на постоянной основе</p> <p>Проведение оценочных процедур</p>
Организация финансирования и предоставление форм государственной поддержки	<p>Подготовка пакета документов для Инвестиционного совета при Губернаторе области</p> <p>Выполнение процессов в рамках процедур по предоставлению грантов, гарантий, обеспечению субсидиями</p> <p>Контроль за использованием грантов, гарантий, субсидий и прочее</p> <p>Подготовка пакета документов для включения объектов в инвестиционные программы по обеспечению инженерной инфраструктурой хозяйствующих субъектов</p> <p>Осуществление финансирования проектных работ (контракта, мероприятий и прочее)</p> <p>Контроль освоения денежных средств</p> <p>Выплаты стипендий и иные социальные выплаты</p>
Информационное сопровождение	<p>Разработка, согласование и утверждение схем, концепций, макетов рекламного продукта</p> <p>Изготовление полиграфической, рекламной продукции</p> <p>Подготовка, размещение, выпуск пресс-релизов, статей, видеороликов и другого рода информации в средствах массовой информации; организация брифингов</p> <p>Организация участия средств массовой информации в мероприятиях</p> <p>Организация и проведение аккредитации средств массовой информации</p>

Направление	Типовой процесс
	Проведение мониторинга публикаций
Реорганизационные процедуры	Процедуры лицензирования, аккредитации, аттестации
	Процедуры реорганизации и ликвидации
Кадровые процедуры и формирование компетенций	Организация и сопровождение процедур назначения и освобождения от должности
	Организация и сопровождение процедур заключения и оформления служебных контрактов, трудовых договоров
	подготовка и оформление наградных документов и документов о поощрении
	организация и проведение профилактических медицинских осмотров
	организация конкурсов и включение в кадровые резервы
	Организация стажировки
	Организация и проведение систематической профессиональной подготовки, повышения квалификации, обучения
	Оценка обучения (тестирование и прочее)
	Оказание методической поддержки
	Запрос и подготовка коммерческих предложений
Договорная работа, закупочная деятельность	Подготовка конкурсной документации
	Разработка проекта договора, контракта, соглашения (в т.ч. гос. контрактов) и прочее
	Разработка государственного задания
	Объявление и проведение торгов
	Согласование, корректировка в процессе согласования и заключение договора, контракта и прочее
	Государственная регистрация договора, контракта и прочее
	Иные работы, связанные с разработкой и заключением договора, контракта и прочее

При определении степени детализации блоков работ проекта рекомендуется руководствоваться следующим принципом: в случае, если ответом на ниже представленные вопросы является «нет», то данный блок работ, возможно декомпозировать далее:

- за работой или процессом закреплён единственный ответственный?
- блок работ означает получение единственного результата?
- требования к ресурсам в ходе выполнения работ или процессов блока работ значительных изменений не претерпят?
- отсутствуют риски, которые могут потребовать дальнейшей детализации блока работ?
- можно ли рассчитать продолжительность и стоимость блока работ с достаточной точностью?
- достаточна ли детализация для того, чтобы установить взаимозависимости между блоками работ?
- ясно ли определен блок работ, соблюдено ли единообразие наименований?
- можно ли точно определить, когда блок работ реализован?
- понятна ли суть блока работ соответствующим заинтересованным лицам?

Пример перечня блоков работ по проекту:

1.	Архитектурное проектирование
1.1.	Разработка генерального плана застройки

1.2.	Разработка проекта планировки территории
1.3.	Разработка ТЗ на проектирование
1.4.	Определение проектной организации
1.5.	Разработка эскизного проекта
1.6.	Согласование эскизного проекта
1.7.	Разработка проектно-сметной документации
1.8.	Проведение государственной экспертизы
2.	Организация строительства
2.1.	Оформление документов на земельный участок
2.2.	Получение разрешения на строительство
2.3.	Подготовка пакета документов для проведения конкурса по выбору генподрядчика
2.4.	Проведение конкурса по отбору генподрядчика
2.5.	Заключение договора генподряда
2.6.	Подготовка и утверждение графика строительства и финансирования
2.7.	Выбор подрядчика
2.8.	Подготовка и утверждение графика строительства
3.	Обеспечение финансирования
3.1.	Формирование реестра потенциальных арендаторов
3.2.	Разработка и утверждение бизнес-плана
3.3.	Открытие кредитной линии
4.	Выполнение строительно-монтажных работ
4.1.	Выполнение земляных работ
4.2.	Подключение к инженерным сетям
4.3.	Устройство свайных фундаментов
4.4.	Устройство перекрытия на отметке -2,150
4.5.	Устройство стен техподполья
4.6.	Устройство перекрытий на отм. 0,000
4.7.	Устройство стен и перегородок 1, 2, 3, 4, 5 этажей
4.7.1.	устройство стен и перегородок 1 этажа
4.7.2.	устройство стен и перегородок 2 этажа
4.7.3.	устройство стен и перегородок 3 этажа
4.7.4.	устройство стен и перегородок 4 этажа
4.7.5.	устройство стен и перегородок 5 этажа
4.8.	Устройство перекрытий на 1, 2, 3, 4, 5 этажах
4.8.1.	устройство перекрытий на 1 этаже
4.8.2.	устройство перекрытий на 2 этаже
4.8.3.	устройство перекрытий на 3 этаже
4.8.4.	устройство перекрытий на 4 этаже
4.8.5.	устройство перекрытий на 5 этаже
4.9.	Устройство покрытия
4.10.	Монтаж лестничных маршей

4.11.	Устройство кровли
4.12.	Наружная отделка
4.13.	Установка оконных и дверных блоков
4.14.	Монтаж витражей
4.15.	Закрыт тепловой контур
4.16.	Поставки и монтаж инвентаря и оборудования
4.17.	Внутренняя отделка
4.18.	Благоустройство территории
4.19.	Разводка системы водоснабжения и водоотведения
4.20.	Разводка системы отопления
4.21.	Устройство системы вентиляции
4.22.	Устройство системы газоснабжения
4.23.	Разводка силового электрооборудования и освещения
4.24.	Подключение инженерных сетей
4.25.	Сдача объекта госкомиссии
4.26.	Регистрация права собственности на объект
5.	Сдача объекта в аренду
5.1.	Распределение квартир по организациям
5.2.	Распределение нежилых помещений по целевому назначению
5.3.	Заключение договоров аренды квартир с организациями, кв.м
5.4.	Заключение договоров аренды квартир с жильцами, кв.м
5.5.	Заключение договоров аренды нежилых помещений площадей, кв.м
6.	Организация вспомогательных сервисов
6.1.	Разработка проекта сервисного обслуживания квартир
6.2.	Разработка проекта сервисного обслуживания нежилых помещений
6.3.	Осуществление сервисного обслуживания квартир
6.4.	Осуществление сервисного обслуживания нежилых помещений
7.	Организация эксплуатации объекта и заселение
7.1.	Выбор управляющей компании
7.2.	Заключение договора с управляющей компанией
7.3.	Приемка объекта управляющей компанией
7.4.	Размещение резидентов, кв.м

В графе «Документ, подтверждающий выполнение» указывается документ, который служит подтверждением выполнения работ или процессов, например: акт выполненных работ, протокол конкурсной комиссии.

Для блоков работ подтверждающий документ не указывается.

В случае, если работа или процесс выполнены до момента утверждения базового плана управления проектом, данный факт отражается в скобках после наименования работы или процесса (с помощью пометки «выполнено»).

Документы, указываемые в данной графе, на стадии реализации проекта являются основанием для подтверждения выполнения соответствующей работы или процесса.

В графе «ФИО ответственного исполнителя» указываются члены рабочей группы (ФИО), ответственные за выполнение работ и процессов, предусмотренных планом-графиком проекта. При распределении ответственности за работы и процессы проекта необходимо руководствоваться принципом «за каждой работой или процессом закреплен единственный ответственный».

2.4. Команда проекта.

Раздел содержит сведения о команде проекта, которая будет заниматься реализацией проекта. В команду проекта входят представители группы, необходимые службы и организации, которые заняты в работах по проекту.

В графе «№ п/п» указывается порядковый номер в арифметической последовательности.

В графе «ФИО, основное место работы, должность» указывается фамилия, имя и отчество, место работы и должность лица, входящего в команду проекта.

В графе «Роль в проекте» в отношении каждого участника команды проекта указывается соответствующая роль: куратор проекта, руководитель проекта, администратор проекта, оператор мониторинга проекта, ответственный за блок работ и члены рабочей группы.

Для члена рабочей группы, выполняющего работу в рамках проекта, необходимо отразить сокращение в виде «(Р)», для участника, выполняющего процесс – в виде «(П)»

2.5. Планирование коммуникаций.

Для налаживания коммуникаций в проекте рекомендуется пользоваться правилом «5 К»: планировать коммуникации в проекте можно с помощью таблицы, ответив на 5 вопросов относительно пути движения информации: «Какая?», «Кто?», «Кому?», «Когда?» и «Как?». Во избежание лишних коммуникаций между участниками проекта рекомендуем задавать контрольный проверочный вопрос «Зачем?».

В данном разделе представлен оптимальный шаблон таблицы, отражающей планирование коммуникаций.

Укажите «Когда?» и «Как?» передается информация в вашем проекте.

2.6. Заинтересованные лица, инвесторы.

Раздел содержит подробную информацию о лицах, непосредственно заинтересованных в результате проекта, но не вошедших в команду проекта. Здесь необходимо отразить данные об инвесторе проекта (в случае его наличия) и иную информацию о крупных потенциальных покупателях и поставщиках проекта с указанием сферы их интересов.

Подумайте, кому может быть интересен ваш проект и его реализация. Укажите данные организаций и физических лиц.

При необходимости план управления проектом может быть дополнен следующими приложениями:

- план управления качеством проекта;
- глоссарий терминов и определений;
- план совершенствования процессов проекта;
- план управления закупками;
- план управления персоналом и др.

Контрольные вопросы:

1. Что такое Паспорт проекта? Для чего он разрабатывается?
2. Расскажите про метод экспертной оценки?
3. Опишите блоки плана управления проектом?

УТВЕРЖДАЮ:

(представитель координирующего органа)
преподаватель ЧОУ ВО «Балтийский
Гуманитарный Институт»

_____ / Азорская Н.В.
(подпись) (ФИО преподавателя)

« ____ » _____ 2023 г.

ПАСПОРТ ПРОЕКТА

Создание и внедрение в эксплуатацию ювелирной мастерской «Золотой орел»:
(полное наименование проекта)

Идентификационный номер _____ 15 _____

ПРИНЯТО К ИСПОЛНЕНИЮ:

(руководитель проекта)

студент группы _____ П-31 _____

_____ / Дубовская И.В.
(подпись) (ФИО студента)

« ____ » _____ 2023 г.

Общие сведения о документе

Назначение документа:	Регламентация взаимодействия между основными участниками проекта, закрепление полномочий и ответственности каждой из сторон в связи с реализацией проекта
Количество экземпляров и место хранения:	Выпускается в 1-м экземпляре, который хранятся у руководителя проекта и сдается представителю координирующего органа
Содержание:	<ol style="list-style-type: none"> 1. Группа управления проектом 2. Цель и результат проекта, сроки реализации
Изменения:	Изменения в паспорт проекта вносятся путем оформления ведомости изменений

1. Группа управления проектом

Название организации	ФИО, должность, контактные данные представителя
Координирующий орган: Частное образовательное учреждение высшего образования «Балтийский Гуманитарный Институт»	Представитель координирующего органа проекта: Преподаватель ЧОУ ВО «БГИ» <u>Азорская Надежда Владимировна</u> (ФИО преподавателя)
Исполнитель: <u>Дубовская Ирина Валентиновна</u> группа ДО-282-сп (ФИО студента, группа)	Руководитель проекта: <u>Дубовская Ирина Валентиновна</u> (ФИО студента) <hr/> <u>П-31</u> (группа) Телефон: 8-568-63-89 E-mail: dubovskaja2000@mail.ru

2. Цель и результат проекта, сроки реализации

Измеримая цель проекта:	Произвести 1000 обручальных колец к началу 2020 года	
Способ достижения цели:	Организовать регулярную работу мастерской	
Результат проекта:	Результат:	Вид подтверждения:
	Организована регулярная работа мастерской в области производства обручальных колец	<ul style="list-style-type: none"> • Свидетельство на собственность помещения; • Бухгалтерская отчетность; • Журнал учета доходов /расходов;
Требования к результату проекта:	Требование:	Вид подтверждения:
	Имеется помещение (мастерская)	Свидетельство на собственность помещения
	Имеются рабочие места	Договоров, соглашений, контрактов;
	Размер среднемесячной заработной платы (20000 рублей)	1. Журнал учета рабочего времени / оплаты труда
	Оборудование(компьютер)	Чек
Пользователи результатом проекта:	Население города Бийска	

УТВЕРЖДАЮ:

(представитель координирующего органа)
преподаватель ЧОУ ВО «Балтийский
Гуманитарный институт»

_____ / Азорская Н.В.
(подпись) (ФИО преподавателя)

«___» _____ 2023 г.

ПЛАН УПРАВЛЕНИЯ ПРОЕКТОМ

Создание и внедрение в эксплуатацию ювелирной мастерской «Золотой орел»
(полное наименование проекта)

Идентификационный номер _____ 15 _____

ПРИНЯТО К ИСПОЛНЕНИЮ:

(руководитель проекта)

студент группы П-31

_____ / Дубовская И.В.
(подпись) (ФИО студента)

«___» _____ 2023 г.

Общие сведения о документе

Назначение документа:	детализация паспорта проекта и инициация блока работ по планированию проекта, с точки зрения человеческих, финансовых и временных ресурсов
Количество экземпляров и место хранения:	выпускается в 1-м экземпляре, который хранится у руководителя проекта и сдается представителю координирующего органа
Содержание:	<ol style="list-style-type: none">1. Календарный план-график работ по проекту2. Бюджет проекта3. Риски проекта4. Команда проекта5. Планирование коммуникаций6. Заинтересованные лица, инвесторы
Изменения:	изменения в плане управления проекта выполняются путем оформления ведомости изменений

1. План-график работ по проекту

Код работы/процесса		Название работы/процесса	Документ, подтверждающий выполнение	ФИО ответственного исполнителя
код	тип (Р/П)			
0		Начало работ		
1		Открытие мастерской		Дубовская Ирина Валентиновна
1.1	П	Аренда помещения	Договор об аренде	Дубовская Ирина Валентиновна
2		Ремонт мастерской		
2.1	Р	Малярные работы	Акт выполненных работ	Дубовская Ирина Валентиновна
2.2	Р	Монтажные работы	Акт выполненных работ	Дубовская Ирина Валентиновна
2.3	Р	Отделочные работы	Акт выполненных работ	Дубовская Ирина Валентиновна
3		Покупка мебели		
3.1	Р	Покупка мебели	Расчётные чеки	Дубовская Ирина Валентиновна
3.2	Р	Сборка мебели	Акт выполненных работ	Дубовская Ирина Валентиновна
3.3	Р	Расстановка мебели	Акт выполненных работ	Дубовская Ирина Валентиновна
4		Заказ ресурсов		
4.1	П	Драгоценные камни	Расчётные чеки	Дубовская Ирина Валентиновна

Код работы/процесса		Название работы/процесса	Документ, подтверждающий выполнение	ФИО ответственного исполнителя
4.2	П	Полудрагоценные камни	Расчётные чеки	Дубовская Ирина Валентиновна
4.3	П	Желтое золото	Расчётные чеки	Дубовская Ирина Валентиновна
4.4	Р	Белое золото	Расчётные чеки	Дубовская Ирина Валентиновна
5		Персонал		
5.1		Поиск персонала	Список требований к кандидату	Дубовская Ирина Валентиновна
5.2	П	Утверждение персонала	Акт выполненных работ	Дубовская Ирина Валентиновна
6		ИТ		
6.1	П	Закупка техники и ПО	Расчётные чеки	Дубовская Ирина Валентиновна
6.2	Р	Доставка техники	Акт выполненных работ	Дубовская Ирина Валентиновна
6.3	Р	Установка техники	Расчётные чеки	Дубовская Ирина Валентиновна
6.4	Р	Настройка техники	Расчётные чеки	Дубовская Ирина Валентиновна
6.5	П	Обучение персонала	Акт выполненных работ	Дубовская Ирина Валентиновна
7		Завершение проекта		

4. Команда проекта

№ п/п	ФИО, должность и основное место работы	Роль в проекте
	Азорская Надежда Владимировна	Куратор проекта
	Дубовская Ирина Валентиновна	Руководитель проекта
	Дубовская Ирина Валентиновна	Администратор проекта
	Дубовская Ирина Валентиновна	Оператор мониторинга проекта
	Дубовская Ирина Валентиновна	Член рабочей группы (Р)
	Дубовская Ирина Валентиновна	Член рабочей группы (П)

5. Планирование коммуникаций

№ п/п	Какая информация передается	Кто передает информацию	Кому передается информация	Когда передает информацию	Как передается информация
1.	Статус проекта	Руководитель проекта	Представителю заказчика, куратору	Еженедельно	Телефонная связь электронная почта
2.	Обмен информацией о текущем состоянии проекта	Администратор проекта	Участникам проекта	Раз в месяц	Телефонная связь/ электронная почта
3.	Документы и информация по проекту	Ответственный по направлению	Администратору проекта и адресатам	Раз в месяц / в день поступления информации	Телефонная связь/ электронная

					почта/Совещание
4.	О выполнении работы или процесса	Администратор проекта	Руководителю проекта, оператору мониторинга	В день поступления информации	Телефонная связь/ электронная почта
5.	Отчет о выполнении блока работ	Администратор проекта	Группе управления, оператору мониторинга	Раз в месяц / в день поступления информации	Совещание
6.	Ведомость изменений	Администратор проекта	Группе управления, оператору мониторинга	В день поступления информации	Телефонная связь/ электронная почта
7.	Мониторинг реализации проекта	Оператор мониторинга	В проектный офис	Раз в месяц / в день поступления информации	Совещание
8.	Информация о наступивших или возможных рисках и отклонениях по проекту	Администратор проекта, ответственное лицо по направлению	Руководителю проекта	В день поступления информации	Телефонная связь/ электронная почта
9.	Информация о наступивших рисках и осложнениях по проекту	Руководитель проекта	Куратору	В день поступления информации	Телефонная связь
10.	Информация о неустранимом отклонении по проекту	Руководитель проекта	Куратору	В день поступления информации	Телефонная связь
11.	Обмен опытом, текущие вопросы	Руководитель проекта	Рабочей группе и приглашенным	Еженедельно	Совещание

12.	Приглашения на совещания	Администратор проекта	Участникам совещания	В день поступления информации	Телефонная связь
13.	Передача поручений, протоколов, документов	Администратор проекта	Адресатам	В день поступления информации	Электронная почта / Телефонная связь/Совещание
14.	Подведение итогов	Руководитель проекта	Куратору	В день поступления информации	Совещание

6.Заинтересованные лица, инвесторы

Заинтересованное лицо, инвестор	Должность, контактные данные
<p><u>Для юридических лиц:</u> Роль в проекте (инвестор): Азорская Надежда Владимировна Частное образовательное учреждение высшего образования «Балтийский Гуманитарный Институт»</p> <p>Телефон: 8 800 770 03 80, (812) 647 63 14 Адрес: г. Санкт-Петербург, пр. Стачек, д. 72, лит. А E-mail: info@bhi.spb.ru</p>	<p>Фамилия имя отчество должность руководителя организации. Азорская Надежда Владимировна Телефон: - E-mail: asorskaja@gmail.com</p>
<p><u>Для физических лиц:</u> Роль в проекте (инвестор):</p> <p>Фамилия имя отчество Адрес:</p>	<p>Должность по основному месту работы</p> <p>Телефон: E-mail:</p>

Практическая работа №9

Реализация индивидуального проекта

В результате выполнения практических заданий обучающийся должен **иметь практический опыт:**

- использования инструментальных средств обработки информации;
- формирования отчетной документации по результатам работ;
- использования стандартов при оформлении программной документации;

В результате выполнения практических заданий обучающийся должен **уметь:**

- осуществлять математическую и информационную постановку задач по обработке информации,
- создавать проект по разработке приложения и формулировать его задачи, выполнять управление проектом с использованием инструментальных средств;

В результате выполнения практических заданий обучающийся должен **знать:**

- основные процессы управления проектом разработки.

Задание для практической работы:

На основе практических работ №1-6 выполнить следующее:

1. Составить список работ по достижению целей проекта (не менее 30) со сроками выполнения.
2. Создать иерархическую структуру работ – группировка работ по этапам выполнения.
3. Продумать связи между работами и нарисовать сетевую диаграмму (она не должна быть линейной).
4. Составить список необходимых ресурсов со стоимостями.
5. Просчитать ориентировочную стоимость и продолжительность проекта.
6. Найти критические задачи.
7. Провести анализ и оптимизацию плана проекта, сформировав отчеты по проекту.
8. Предложить один вариант, с помощью которых возможно сократить сроки проекта, рассчитать на сколько. Рассчитать, как предложенные мероприятия отразятся на стоимости проекта и качестве продукта проекта.
9. Предложить один вариант, с помощью которых возможно сократить стоимость проекта, рассчитать на сколько. Рассчитать, как предложенные мероприятия отразятся на сроках проекта и качестве продукта проекта.
10. Оформить описание выполнения проекта в виде отчета.

Практическая работа №10

Оформление презентации проекта Microsoft Office PowerPoint.

Презентацию проекта разрабатывает автор проекта, при необходимости с представителем заказчика проекта. Пример содержание презентации представлен в Приложении.

Для подготовки презентации проекта предлагается использовать программу для создания и проведения презентаций, например: Microsoft Office PowerPoint. Презентация проекта оформляется по установленной форме в электронном формате в соответствии со следующими требованиями:

- используемый шрифт – Franklin Gothic Book, для заголовков используется Franklin Gothic Medium (по умолчанию предлагаемые шрифты обозначены в шаблоне слайдов);
- минимальный размер шрифта – 12;
- выравнивание текста относительно верхних и нижних границ ячеек таблиц – по середине.

Содержание разделов презентации проекта

- *Титульный лист паспорта проекта (слайд 1).*

На титульном листе презентации проекта указывается:

- логотипы;
- наименование проекта, которое необходимо начинать со слов, выражающих действие, далее указывается объект или сфера воздействия, завершать рекомендуется указанием территории реализации проекта, например: «Благоустройство склона по ул. Д. Бедного г. Валуйки», «Внедрение программно-технического комплекса «Регистратура» в поликлинических учреждениях Белгородской области».

- ФИО и должность докладчика;
- место (географическое расположение: поселок, город и т.д.) и год рассмотрения презентации проекта.

- *Введение в предметную область (описание ситуации «как есть») (слайд 2).*

В данном разделе тезисно описывается ситуация и параметры соответствующей области (относительно объекта управления) до начала реализации проекта с использованием изображений, графиков и таблиц. Также обозначается проблема, на решение которой направлена реализация данного проекта. При необходимости количество слайдов по данному направлению может увеличиваться.

- *Цель и результат проекта (слайд 3).*

Слайд содержит следующие сведения:

- в строке «Цель проекта» указывается цель, которую необходимо достичь по факту окончания проекта. Цель проекта – запланированное желаемое состояние объекта управления, она должна соответствовать следующим требованиям:
 - отражать ожидаемый социально-экономический полезный эффект от реализации проекта;
 - иметь измеримые количественные показатели и сроки достижения;
 - быть достижимой в реальных условиях, в которых осуществляется проект;
 - полностью находиться в сфере ответственности и влияния исполнителя проекта.

Пример: организовать переработку муки в объеме 29,5 тыс. тонн в год на территории Шебекинского района к концу 2016 года;

- в строке «Способ достижения цели» необходимо указать оптимальный путь достижения обозначенной цели, например: строительство мельницы;

• в строке «Результат проекта», графа «Результат» указывается измеримое выражение социальных, экономических, интеллектуальных и иных эффектов, полученных в результате реализации проекта (материальный объект, предоставленная услуга, нематериальный актив, знания и т.д.). Фактический результат - это подтверждение достижения цели, например: организована работа мельницы по переработке муки с проектной мощностью не менее 29,5 тыс. тонн в год Шебекинского района;

- в строке «Требования к результату проекта» указываются качественные и количественные характеристики результата, которые будут учитываться при его достижении (площадь объекта; количество смен работы; требования, указанные в техническом задании; наличие определенного вида оборудования; создание ряда объектов и т.д.).

- в строке «Пользователи результатом проекта» указывается круг потребителей (область применения) результата проекта.

- *Введение в предметную область (описание ситуации «как будет»)* (слайд 4).

В данном разделе тезисно описывается ситуация и параметры соответствующей области (относительно объекта управления) после окончания реализации проекта с использованием изображений, графиков и таблиц. Для экономических проектов необходимо обозначить рынок сбыта продукции/услуг. При необходимости количество слайдов по данному направлению может увеличиваться.

- **Основные блоки работ проекта (слайд 5).**

В данном разделе необходимо указать перечень основных блоков работ, которые планируется выполнить в рамках реализации проекта, без декомпозиции до отдельных работ.

Слайд содержит следующие данные:

- в графе «Название задачи» перечисляются блоки работ;
- в графах «Дата начала» и «Дата окончания» указываются соответственно плановые даты начала и окончания блоков работ в формате дд.мм.гг;
- в графе «Длительность» указывается планируемая продолжительность блоков работ в рабочих днях;
- графическое представление основных блоков работ путем разбивки правой части представленной таблицы на отчетные периоды (месяц, квартал) и окрашивания необходимого периода в соответствии с представленными датами. Завершенные блоки работ на момент демонстрации проекта закрашиваются зеленым цветом, планируемые – синим.

- **Команда проекта (слайд 6).**

Слайд содержит сведения о команде проекта, которая будет заниматься реализацией проекта. В команду проекта входят представители органов власти, местного самоуправления и сотрудники хозяйствующих субъектов, которые заняты в работах по проекту.

В графе «ФИО» указывается фамилия, имя и отчество участника команды проекта.

В графе «Должность и основное место работы» указываются соответствующие данные в отношении всех участников команды проекта.

В графе «Выполняемые в проекте работы» указываются работы, которые выполняет данное лицо в проекте.

В отношении обязательных ролей: куратора проекта, руководителя проекта, администратора проекта и оператора мониторинга проекта выполняемые работы не указываются в случае, если у данных лиц кроме обязанностей по указанным ролям отсутствует дополнительная ответственность в рамках проекта.

- **Контактные данные (слайд 7).**

На слайде представляются контактные данные руководителя проекта: телефон и адрес электронной почты.

СЛАЙД 1

Добрый день!

Представляю вашему вниманию проект «ОРГАНИЗАЦИЯ ФОРУМА НАРОДНОГО ТВОРЧЕСТВА, СЕМЕЙНЫХ ЦЕННОСТЕЙ И КУЛЬТУРЫ «ОСКОЛЕЦ».

СЛАЙД 2

Жизнь человека неразрывно связана с семьёй. Именно там происходит становление и реализация личности. Понятие счастья почти каждый человек связывает, прежде всего, с семьёй: счастлив тот, кто счастлив в своем доме.

В настоящее время родительская семья находится в стадии отчуждения, что приводит к потере связи между поколениями. В подавляющем большинстве искусство, окружающее человека, - массовое западное, а не отечественное. Такая ситуация представляет серьезную опасность нивелирования национального своеобразия культуры. Закономерно, российская культура позиционируется, прежде всего, через русскую с учетом диалога культур разных народов страны.

СЛАЙД 3

Проект организации Форума народного творчества, семейных ценностей и культуры «ОСКОЛЕЦ» направлен на формирование ценностей семьи: любовь, верность, взаимное уважение, дружба, согласие, забота о детях.

Целью проекта является организация и проведение ФОРУМА НАРОДНОГО ТВОРЧЕСТВА, СЕМЕЙНЫХ ЦЕННОСТЕЙ И КУЛЬТУРЫ «ОСКОЛЕЦ» на территории хутора Чумаки с привлечением более 1000 представителей общественных организаций, творческих коллективов, молодых семей, учащихся образовательных учреждений к 08 июля 2015 года – Дню семьи, любви и верности.

Для *достижения цели* планируется организация работы пяти площадок по направлениям: «Литературная мастерская», «Выставка мастеров народного творчества», «Историческая реконструкция», «Спортивные соревнования среди молодых семей», «Фестиваль народного творчества и культуры».

Результатом проекта является вовлечение не менее 1000 участников в мероприятия Форума.

К результату проекта поставлено *требование* наличия культурно-образовательной Программы Проекта.

Пользователем результата проекта является население Старооскольского городского округа.

СЛАЙД 4

Основание для открытия проекта представлено на слайде.

СЛАЙД 5

В рамках проведения Форума *планируется* работа 5 площадок по числу направлений.

Участники «Литературной мастерской» представят свои произведения на выставке-продаже, проведут поэтическое ристалище - состязание поэтов и бардов.

«Выставка мастеров народного творчества» в своей экспозиции покажет достижения мастеров традиционных ремёсел и народных промыслов. Художники боди-арта и мастера традиционных ремёсел и народных промыслов проведут мастер-классы.

«Историческую реконструкцию» представят коллективы, которые покажут фрагменты традиционных праздников народного календаря, обычаев, обрядов, игр, спортивных состязаний и боёв.

Для молодых семей с детьми пройдут «Спортивные соревнования», в которых участники должны будут показать единство и взаимовыручку команды.

На площадке «Фестиваля народного творчества и культуры» представят свои номера вокальные коллективы, исполняющие народную музыку: хоры и ансамбли, хореографические коллективы с исполнением русских народных танцев, инструментальные коллективы: оркестры и ансамбли с исполнением народной музыки.

Формы деятельности творческих площадок Форума выполняют информационно-просветительскую, коммуникативную и воспитательную функции, благодаря которым формируется положительный облик организации семейных взаимоотношений в деятельности.

СЛАЙД 6

Работы по реализации проекта разделены на 3 этапа. *Подготовительный* включает в себя формирование организационного комитета, разработку и утверждение программы и Положения о проведении Форума, а также разработку и подготовку плана информационного сопровождения проекта.

СЛАЙД 7

На *организационном этапе* разрабатывается и утверждается программа площадок народного творчества и культуры, проводится расширенная информационная компания и формируется список участников Форума.

СЛАЙД 8

Заключительный этап предполагает проведение Форума «ОСКОЛЕЦ», подготовку презентации и итогового отчета о реализации проекта.

СЛАЙД 9

К *рискам* проекта относятся неблагоприятные погодные условия при проведении Форума, низкая мотивация молодёжи к участию в проекте, недостаточное количество организаторов площадок.

Проект не имеет бюджетных финансовых затрат.

СЛАЙД 10

Показателем социальной эффективности проекта является охват населения не менее 1000 участников форума.

СЛАЙД 11

Команда проекта представлена на слайде, в рабочую группу включены представители управлений по делам молодежи, образования, социальной защиты населения, ЗАГС администрации Старооскольского городского округа, ...

СЛАЙД 12

МАУ «Центр молодежных инициатив», МУК «Старооскольский краеведческий музей», МУК «Старооскольский Дом ремесел», а также представители общественных организаций и творческих коллективов.

Проект «ОРГАНИЗАЦИЯ ФОРУМА НАРОДНОГО ТВОРЧЕСТВА, СЕМЕЙНЫХ ЦЕННОСТЕЙ И КУЛЬТУРЫ «ОСКОЛЕЦ» вовлекает молодежь в изучение историко-культурного наследия Старооскольского городского округа, способствует формированию семейных ценностей и укреплению института семьи и семейных отношений.

СЛАЙД 13

Спасибо за внимание!

4. Критерии оценивания

В процессе выполнения практической работы каждый студент составляет индивидуальный отчет, который включает расчетную часть и/или графическую часть.

Оценивание проводится по пятибалльной шкале.

Оценка «зачтено» выставляется, если обучающийся активно работает в течение всего практического занятия, вопросы освещены полно в соответствии с планом практического занятия и показывает при этом овладение лекционным материалом.

Оценка «не зачтено» выставляется в случае, когда обучающийся обнаружил несостоятельность осветить вопросы освещены неправильно, бессистемно, с грубыми ошибками, отсутствуют понимания основной сути вопросов, обнаружено неумение решать учебные задачи.

II. ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ «ПРОЕКТНАЯ ДЕЯТЕЛЬНОСТЬ И УПРАВЛЕНИЕ ПРОЕКТАМИ»

1. Форма проведения промежуточной аттестации: экзамен.

2. Процедура проведения: Экзамен может проводиться по итогам текущей успеваемости и сдачи практических работ, предусмотренных программой дисциплины и (или) путем организации специального опроса, проводимого в устной форме.

3. Перечень вопросов для подготовки к экзамену:

Лекция 1. ИСТОРИЯ И КОНЦЕПЦИЯ УПРАВЛЕНИЯ ПРОЕКТАМИ

1. Назовите основные этапы развития проектной деятельности за рубежом.
2. Назовите основные этапы развития проектной деятельности в России.
3. В каких отраслях экономики управление проектами применяется наиболее широко?
4. В каком направлении расширялась область применения проектной деятельности?

Лекция 2. ВВЕДЕНИЕ В УПРАВЛЕНИЕ ПРОЕКТАМИ

1. Дайте определение проекта.
2. Дайте определение управления проектами.
3. Каковы основные признаки проектной деятельности?
4. В чем состоит отличие проектов и бизнес-процессов?
5. В чем состоит различие требований к функциональным и проектным менеджерам?

6. Приведите пример проекта и дайте его полную классификацию.

7. Зачем нужна классификация проектов?

8. Что такое среда проекта?

9. Что включает в себя непосредственное окружение проекта?

Лекция 3. УПРАВЛЕНИЕ СОДЕРЖАНИЕМ ПРОЕКТА

1. Назовите этапы жизненного цикла проекта.
2. Каковы основные причины появления проектов?
3. Как построить презентацию проекта?
4. Что включает в себя описание проекта?
5. Почему описание проекта должно включать в себя мотивы (причины) инициации проекта?

6. Что в описании проекта помогает очертить его границы?

7. Какие особые условия могут диктоваться обстоятельствами? Приведите пример.

Лекция 4. УПРАВЛЕНИЕ ВРЕМЕНЕМ

1. Какие инструменты управления временем вы знаете?
2. Что такое план проекта? Что он должен включать?
3. Что такое декомпозиция задач?
4. Что такое трудоемкость, каковы единицы ее измерения?
5. Чем трудоемкость отличается от времени выполнения задачи?
6. Что такое контрольные точки?
7. Что такое резерв времени, от чего он зависит?

Лекция 5. УПРАВЛЕНИЕ СТОИМОСТЬЮ

1. Что такое управление стоимостью проекта?
2. Какие инструменты управления стоимостью вы знаете?
3. Какие стадии проходит формирование бюджета?
4. В чем отличие бюджета от сметы?
5. Какие виды расходов существуют в проекте?
6. Что такое расходы на персонал?
7. Что такое расходы на выполнение работ по проекту?
8. Что такое организационные расходы?

9. Можно ли в целях экономии средств обойтись без организационных расходов?
10. Что такое фиксированные и переменные расходы?
11. Для чего в бюджете проекта необходимо разделять фиксированные и переменные расходы?
12. Для чего нужна таблица анализа ресурсов?
13. Что такое график использования денежных средств?
14. Для чего нужен график использования денежных средств?
15. Что такое резерв бюджета?
16. От чего зависит величина резерва бюджета?

Лекция 6. УПРАВЛЕНИЕ ЧЕЛОВЕЧЕСКИМИ РЕСУРСАМИ

1. Кто такие участники проекта?
2. Могут ли роли участников совпадать?
3. Кто является главным участником проекта? Почему?
4. Как формируется структура команды проекта?
5. По каким принципам подбирается команда проекта?
6. Каковы преимущества и недостатки совместительства для членов команды?
7. Каковы характеристик эффективной проектной группы?
8. Каковы важнейшие составляющие благоприятного социально-психологического климата?
9. Каковы стадии жизненного цикла команды?

Лекция 7. УПРАВЛЕНИЕ КАЧЕСТВОМ

1. Что такое качество?
2. Являются ли понятия «качественный» и «первосортный» тождественными? Почему?
3. Какие процессы включает в себя управление качеством?
4. Должно ли качество планироваться?
5. Какие аспекты управления качеством вы знаете?

Лекция 8. УПРАВЛЕНИЕ РИСКАМИ

1. Что такое риски?
2. Что включает в себя управление рисками?
3. Каковы этапы управления рисками?
4. Какие методы снижения рисков вы знаете?
5. Какие основные типы факторов риска вам известны?
6. Какие существуют подходы к классификации рисков?

Лекция 9. УПРАВЛЕНИЕ ИЗМЕНЕНИЯМИ

1. Что такое управление изменениями?
2. Какие стадии включает в себя процесс контроля реализации изменений?
3. Что такое срок согласования изменения?
4. Кто принимает окончательное решение о внесении изменения в проект?
5. Должен ли заказчик визировать форму регистрации изменений? Зачем это нужно?

Лекция 10. ВЫПОЛНЕНИЕ ПРОЕКТА

1. Какие задачи стоят перед менеджером при реализации проекта?
2. Что включает в себя анализ текущего положения дел?
3. Как нужно собирать информацию для оценки хода работ?
4. Какие виды контроля вы знаете?
5. Что включают в себя текущие отчеты для заказчика?
6. Как часто нужно отчитываться перед заказчиком о ходе работ?
7. Какие способы ликвидации отставания по срокам вы знаете?
8. Какие существуют правила использования резерва времени?
9. Какие типичные проблемы могут возникать в проекте?
10. Нужно ли информировать заказчика о возникающих проблемах?

Лекция 11. ЗАВЕРШЕНИЕ ПРОЕКТА

1. Что такое завершение проекта?
2. Что включает в себя завершение договорных обязательств?
3. Как происходит освобождение персонала из проекта?
4. Что включает в себя анализ проекта?
5. Нужно ли анализировать незавершенные и неосуществленные проекты?
6. Что такое руины проекта?

4. Критерии оценивания

«Отлично» – за глубокое и полное овладение содержанием учебного материала, в котором студент легко ориентируется, владение понятийным аппаратом за умение связывать теорию с практикой, решать практические задачи, высказывать и обосновывать свои суждения. Отличная отметка предполагает грамотное, логичное изложение ответа (как в устной, так и в письменной форме), качественное внешнее оформление;

«Хорошо» – если студент полно освоил учебный материал, владеет понятийным аппаратом, ориентируется в изученном материале, осознанно применяет знания для решения практических задач, грамотно излагает ответ, но содержание и форма ответа имеют некоторые неточности;

«Удовлетворительно» – если студент обнаруживает знание и понимание основных положений учебного материала, но излагает его неполно, непоследовательно, допускает неточности в определении понятий, в применении знаний для решения практических задач, не умеет доказательно обосновать свои суждения;

«Неудовлетворительно» – если студент имеет разрозненные, бессистемные знания, не умеет выделять главное и второстепенное, допускает ошибки в определении понятий, искажает их смысл, беспорядочно и неуверенно излагает материал, не может применять знания для решения практических задач; за полное незнание и непонимание учебного материала или отказ отвечать.

Учебная практика по ПМ.01

ПАСПОРТ ОЦЕНОЧНЫХ СРЕДСТВ

Перечень формируемых компетенций:

ОК 1 Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.

ОК 2 Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.

ОК 3 Планировать и реализовывать собственное профессиональное и личностное развитие.

ОК 4 Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.

ОК 5 Осуществлять устную и письменную коммуникацию на государственном языке с учётом особенностей социального и культурного контекста.

ОК 6. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей.

ОК 7. Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях.

ОК 8. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности.

ОК 9 Использовать информационные технологии в профессиональной деятельности.

ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языке.

ПК 1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием

ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием

ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств

ПК 1.4 Выполнять тестирование программных модулей

ПК 1.5 Осуществлять рефакторинг и оптимизацию программного кода

ПК 1.6 Разрабатывать модули программного обеспечения для мобильных платформ.

Сформированность вышеперечисленных компетенций предполагает следующие результаты прохождения учебной практики по ПМ.01 Разработка модулей программного обеспечения для компьютерных систем:

Компетенция	Планируемые результаты практики	Наименование оценочного средства
ОК 1: Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.	Знает: актуальный профессиональный и социальный контекст, в котором приходится работать и жить; основные источники информации и ресурсов для решения задач и проблем в профессиональном и/или социальном контексте; актуальные стандарты выполнения работ в профессиональной и смежных областях; актуальные методы работы в профессиональной и	Задание 1

	<p>смежных сферах.</p> <p>Умеет: распознавать задачу и/или проблему в профессиональном и/или социальном контексте; анализировать задачу и/или проблему и выделять ее составные части; правильно определить и найти информацию, необходимую для решения задачи и/или проблемы; составить план действия, определить необходимые ресурсы; владеть актуальными методами работы в профессиональной и смежных сферах; реализовать составленный план; оценить результат и последствия своих действий (самостоятельно или с помощью наставника). Имеет практический опыт: распознавания сложных проблем в знакомых ситуациях; выделения составной части проблемы и описывать ее причины и ресурсы, необходимые для ее решения в целом; определения потребности в информации; поиска и анализа информации; выделения главных и альтернативных источников нужных ресурсов; разработки детального плана действий и следования этому плану; оценивания результата своей работы, выделения в нем сильных и слабых сторон.</p>	
<p>ОК 2: Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.</p>	<p>Знает: номенклатуру информационных источников, применяемых в профессиональной деятельности; приемы структурирования информации; формат оформления результатов поиска информации.</p> <p>Умеет: определять задачи поиска информации; определять необходимые источники информации; планировать процесс поиска; структурировать получаемую информацию; выделять наиболее значимое в перечне информации; оценивать практическую значимость результатов поиска; оформлять результаты поиска.</p> <p>Имеет практический опыт: планирования информационного поиска из широкого набора источников, необходимого для выполнения профессиональных задач; проведения анализа полученной информации, выделения в ней главных</p>	<p>Задание 1</p>

	аспектов; структурирования отобранной информации в соответствии с параметрами поиска; интерпретации полученной информации	
ОК 3: Планировать и реализовывать собственное профессиональное и личностное развитие.	Знает: содержание актуальной нормативно-правовой документации; современную научную и профессиональную терминологию; возможные траектории профессионального развития и самообразования. Умеет: определять актуальность нормативно-правовой документации в профессиональной деятельности. Имеет практический опыт: использования актуальной нормативно-правовой документации по профессии (специальности); применения современной научно профессиональной терминологии; определения траектории профессионального развития и самообразования	Задание 1
ОК 4: Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.	Знает: психологию коллектива; психологию личности; основы проектной деятельности. Умеет: организовывать работу коллектива и команды; взаимодействовать с коллегами, руководством, клиентами. Имеет практический опыт: участия в деловом общении для эффективного решения деловых задач; планирования профессиональной деятельности.	Задание 1
ОК 5: Осуществлять устную и письменную коммуникацию на государственном языке с учётом особенностей социального и культурного контекста.	Знает: особенности социального и культурного контекста; правила оформления документов. Умеет: излагать свои мысли на государственном языке; оформлять документы. Имеет практический опыт: грамотно устно и письменно излагать свои мысли по профессиональной тематике на государственном языке; проявления толерантности в рабочем коллективе.	Задание 1 Отчёт
ОК 6. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей.	Знает: сущность гражданско-патриотической позиции; общечеловеческие ценности; правила поведения в ходе выполнения профессиональной деятельности. Умеет: описывать значимость своей профессии; презентовать структуру профессиональной деятельности по профессии (специальности)	Задание 1
ОК 7. Содействовать сохранению	Знать: правила экологической	Задание 1

<p>окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях.</p>	<p>безопасности при ведении профессиональной деятельности; основные ресурсы, задействованные в профессиональной деятельности; пути обеспечения ресурсосбережения Уметь: соблюдать нормы экологической безопасности; определять направления ресурсосбережения в рамках профессиональной деятельности по профессии (специальности)</p>	
<p>ОК 8. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности.</p>	<p>Знать: роль физической культуры в общекультурном, профессиональном и социальном развитии человека; основы здорового образа жизни; условия профессиональной деятельности и зоны риска физического здоровья для профессии (специальности); Средства профилактики перенапряжения Уметь: использовать физкультурно-оздоровительную деятельность для укрепления здоровья, достижения жизненных и профессиональных целей; применять рациональные приемы двигательных функций в профессиональной деятельности; пользоваться средствами профилактики перенапряжения характерными для данной профессии (специальности)</p>	<p>Задание 1</p>
<p>ОК 9: Использовать информационные технологии в профессиональной деятельности.</p>	<p>Знает: современные средства и устройства информатизации; порядок их применения и программное обеспечение в профессиональной деятельности Умеет: применять средства информационных технологий для решения профессиональных задач; использовать современное программное обеспечение Имеет практический опыт: применения средств информатизации и информационных технологий для реализации профессиональной деятельности</p>	<p>Задание 1 Отчёт</p>
<p>ПК 1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием</p>	<p>Знает: основные принципы создания алгоритмов. Умеет: составлять алгоритм работы программного модуля. Имеет практический опыт: написания программного кода.</p>	<p>Задание 1</p>
<p>ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием</p>	<p>Знает: основные принципы технологии структурного и объектно-ориентированного программирования Умеет: составлять алгоритм работы программного модуля.</p>	<p>Задание 1</p>

	Имеет практический опыт: написания программного кода.	
ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств	Знает: основные принципы отладки и тестирования программных продуктов; Умеет: выполнять отладку и тестирование программы на уровне модуля; Имеет практический опыт: использования инструментальных средств на этапе отладки программного продукта;	Задание 1
ПК 1.4 Выполнять тестирование программных модулей	Знает: основные принципы отладки и тестирования программных продуктов; Умеет: выполнять отладку и тестирование программы на уровне модуля; Имеет практический опыт: проведения тестирования программного модуля по определенному сценарию	Задание 1
ПК 1.5 Осуществлять рефакторинг и оптимизацию программного кода	Знает: основные этапы разработки программного обеспечения; основные принципы технологии структурного и объектно-ориентированного программирования; Умеет: осуществлять разработку кода программного модуля на современных языках программирования; Имеет практический опыт: разработки алгоритма поставленной задачи и реализации его средствами автоматизированного проектирования	Задание 1
ПК 1.6 Разрабатывать модули программного обеспечения для мобильных платформ	Знает: Основные этапы разработки программного обеспечения. Основные принципы технологии структурного и объектно-ориентированного программирования. Умеет: Осуществлять разработку кода программного модуля на современных языках программирования. Оформлять документацию на программные средства Имеет практический опыт: разработки мобильных приложений	Задание 1

1. ТЕКУЩИЙ КОНТРОЛЬ ПО ПРАКТИКЕ

Типовые контрольные задания, необходимые для оценки планируемых результатов учебной практики УП.01.01 по ПМ.01 Разработка модулей программного обеспечения для компьютерных систем соответствуют видам работ, определенным в РП профессионального модуля:

ОЦЕНОЧНОЕ СРЕДСТВО Задание 1

1. Цель: освоить необходимые компетенции по виду работ «Разработка модулей программного обеспечения для компьютерных систем»
2. Проверяемые компетенции (код): ОК 1, ОК 2, ОК 3, ОК 4, ОК 5, ОК 6, ОК 7, ОК 8, ОК 9, ОК 10, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6.
3. Пример оценочного средства:

ПРАКТИЧЕСКИЕ ЗАДАНИЯ

№	Перечень заданий / вопросов
1.	Провести анализ предметной области «Поиск и отбор соискателей на вакантную должность»
2.	Подготовить техническое задание на разработку ПО
3.	Выполнить модификацию хранилища данных
4.	Сформулировать и реализовать запросы к БД согласно потребностям пользователей системы
5.	Создать графический интерфейс для работы с БД
6.	Реализовать основной функционал системы
7.	Подготовить план тестирования ПО, разработать тест-кейсы/ чек-лист
8.	Провести тестирование ПО, оформить отчёт по результатам тестирования
9.	Разработать документацию по эксплуатации ПО. Руководство администратора, руководство пользователя (описав процедуры инсталляции системы и её настройки)
10.	Рассчитать экономическую эффективность ПО

4. Критерии оценивания

Уровень освоения	Критерии
Оценка «отлично»	Практическое задание выполнено в срок и в полном объеме, не содержит фактических ошибок, студент отвечает на все поставленные вопросы, уверенно ориентируется в теоретическом материале и этапах выполнения практического задания
Оценка «хорошо»	Практическое задание выполнено в срок и в полном объеме, не содержит фактических ошибок, но при выполнении задания у студента возникали затруднения и в задании имеются недочеты, не снижающие общего качества работы. Студент отвечает на все поставленные вопросы, уверенно ориентируется в теоретическом материале и этапах выполнения практического задания
Оценка «удовлетворительно»	Практическое задание выполнено в срок, в большей степени, либо имеются 1-2 фактических ошибки. Студент не всегда отвечает на поставленные вопросы, и не всегда ориентируется в теоретическом материале и этапах выполнения практического задания
Оценка «неудовлетворительно»	Практическое задание не выполнено в срок, либо выполнено меньше половины задания, при этом имеются фактические ошибки. Студент не ориентируется в теоретическом материале и этапах выполнения практического задания и затрудняется ответить на поставленные вопросы

2. ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ ПО ПРАКТИКЕ

1. Форма поведения промежуточной аттестации дифференцированный зачет

2. Процедура проведения промежуточной аттестации:

Студенты предоставляют распечатанный отчёт и отвечают на вопросы по его содержанию и о заданиях, выполненных на практике.

3. ОЦЕНОЧНОЕ СРЕДСТВО Отчет по практике

4. Критерии оценивания отчета по практике

4-бальная шкала (уровень освоения)	Показатели	Критерии
Отлично	1. Соответствие содержания отчета требованиям программы практики; 2. Структурированность и полнота собранного материала; 3. Полнота устного выступления, правильность ответов на вопросы при защите; 4. Обоснованность сформулированных предложений.	При защите отчета студент продемонстрировал глубокие и системные знания, полученные при прохождении практики, свободно оперировал данными исследования и внес обоснованные предложения. Студент правильно и грамотно ответил на поставленные вопросы. Студент получил положительный отзыв от руководителя
Хорошо		При защите отчета студент показал глубокие знания, полученные при прохождении практики, свободно оперировал данными исследования. В отчете были допущены ошибки, которые носят несущественный характер. Студент ответил на поставленные вопросы, но допустил некоторые ошибки, которые при наводящих вопросах были исправлены. Студент получил положительный отзыв от руководителя
Удовлетворительно		Отчет имеет поверхностный анализ собранного материала, нечеткую последовательность его изложения материала. Студент при защите отчета по практике не дал полных и аргументированных ответов на заданные вопросы. В отзыве руководителя имеются существенные замечания.
Неудовлетворительно		Отчет не имеет детализированного анализа собранного материала и не отвечает установленным требованиям. Студент затрудняется ответить на поставленные вопросы или допускает в ответах принципиальные ошибки. В отзыве руководителя имеются существенные критические замечания.

5. Перечень вопросов для проведения устного опроса

Перечень вопросов	
1.	Основные этапы разработки ПО
2.	Состав типовой документации на разработку ПО
3.	Преимущества и недостатки настольных систем и систем с веб-интерфейсом
4.	Особенности обеспечения безопасности персональных данных
5.	Какие технологии использовались при разработке ПО

6.	Какие фреймворки использовались в работе и их назначение
7.	Понятие реинжиниринга
8.	Виды и этапы тестирования ПО

6. Критерии оценивания ответов на устные вопросы:

4-бальная шкала (уровень освоения)	Показатели	Критерии
Отлично	1. Полнота и правильность ответа на поставленный вопрос 2. Степень осознанности, понимания изученного 3. Применения профессиональной терминологии в беседе	Студент полно излагает материал, дает правильное определение основных понятий. Понимает материал, может обосновать свои суждения, привести необходимые примеры. Излагает материал последовательно, используя профессиональную терминологию
Хорошо		Студент полно излагает материал, дает правильное определение основных понятий. Студент допускает некоторые ошибки, имеются недочеты в последовательности излагаемого материала
Удовлетворительно		Студент излагает материал неполно и допускает неточности в определении понятий, не умеет достаточно глубоко и доказательно обосновать свои суждения и привести примеры. Излагает материал непоследовательно и допускает ошибки в терминологии.
Неудовлетворительно		Незнание большей части соответствующего вопроса, допускает ошибки в формулировке определений.

7. Методические материалы, определяющие процедуры оценивания результатов учебной практики по ПМ.01 Разработка модулей программного обеспечения для компьютерных систем

Учебная практика проводится концентрировано после изучения профессиональных модулей в учебных кабинетах института.

Все студенты перед началом практики обязаны присутствовать на организационном собрании, которое проводят руководители учебной практики - преподаватели института. Руководитель практики перед началом практики проводит разъяснения целей, задач и содержания учебной практики. В ходе прохождения студентами практики преподаватель (руководитель практики):

- оказывает методическую помощь при выполнении заданий; – проводит индивидуальные и групповые консультации;
- проверяет ход прохождения практики студентами;
- контролирует условия проведения практики, в том числе требования охраны труда, безопасности жизнедеятельности и пожарной безопасности в соответствии с правилами и нормами.

Во время учебной практики студенты работают непосредственно на рабочих местах в компьютерных аудиториях института. Каждый студент индивидуально

выполняет работы, предусмотренные программой учебной практики и настоящим фондом оценочных средств.

Текущий контроль результатов освоения программы учебной практики осуществляется руководителем практики от института в процессе выполнения обучающимися предусмотренных программой и фондом оценочных средств видов работ.

В результате освоения программы учебной практики УП.01.01 по ПМ 01 разработка модулей программного обеспечения для компьютерных систем студенты проходят промежуточную аттестацию в форме дифференцированного зачета, который выставляется на основании выполнения студентами всех, предусмотренных программой практики и фондом оценочных средств видов работ, с учетом их объема и качества, и их защиты в форме опроса.

Производственная практика ПМ.01

ПАСПОРТ ОЦЕНОЧНЫХ СРЕДСТВ

Перечень формируемых компетенций:

ОК 1 Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.

ОК 2 Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.

ОК 3 Планировать и реализовывать собственное профессиональное и личностное развитие.

ОК 4 Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.

ОК 5 Осуществлять устную и письменную коммуникацию на государственном языке с учётом особенностей социального и культурного контекста.

ОК 6. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей.

ОК 7. Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях.

ОК 8. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности.

ОК 9 Использовать информационные технологии в профессиональной деятельности.

ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языке.

ПК 1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием

ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием

ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств

ПК 1.4 Выполнять тестирование программных модулей

ПК 1.5 Осуществлять рефакторинг и оптимизацию программного кода

ПК 1.6 Разрабатывать модули программного обеспечения для мобильных платформ.

Сформированность вышеперечисленных компетенций предполагает следующие результаты прохождения производственной практики по ПМ.01 Разработка модулей программного обеспечения для компьютерных систем:

Компетенция	Планируемые результаты практики	Наименование оценочного средства
ОК 1: Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.	Знает: актуальный профессиональный и социальный контекст, в котором приходится работать и жить; основные источники информации и ресурсов для решения задач и проблем в профессиональном и/или социальном контексте; актуальные стандарты выполнения работ в профессиональной и смежных	Задание 1

	<p>областях; актуальные методы работы в профессиональной и смежных сферах.</p> <p>Умеет: распознавать задачу и/или проблему в профессиональном и/или социальном контексте; анализировать задачу и/или проблему и выделять ее составные части; правильно определить и найти информацию, необходимую для решения задачи и/или проблемы; составить план действия, определить необходимые ресурсы; владеть актуальными методами работы в профессиональной и смежных сферах; реализовать составленный план; оценить результат и последствия своих действий (самостоятельно или с помощью наставника). Имеет практический опыт: распознавания сложных проблем в знакомых ситуациях; выделения составной части проблемы и описывать ее причины и ресурсы, необходимые для ее решения в целом; определения потребности в информации; поиска и анализа информации; выделения главных и альтернативных источников нужных ресурсов; разработки детального плана действий и следования этому плану; оценивания результата своей работы, выделения в нем сильных и слабых сторон.</p>	
<p>ОК 2: Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.</p>	<p>Знает: номенклатуру информационных источников, применяемых в профессиональной деятельности; приемы структурирования информации; формат оформления результатов поиска информации.</p> <p>Умеет: определять задачи поиска информации; определять необходимые источники информации; планировать процесс поиска; структурировать получаемую информацию; выделять наиболее значимое в перечне информации; оценивать практическую значимость результатов поиска; оформлять результаты поиска.</p> <p>Имеет практический опыт: планирования информационного поиска из широкого набора источников, необходимого для выполнения профессиональных задач; проведения анализа</p>	<p>Задание 1</p>

	полученной информации, выделения в ней главных аспектов; структурирования отобранной информации в соответствии с параметрами поиска; интерпретации полученной информации	
ОК 3: Планировать и реализовывать собственное профессиональное и личностное развитие.	Знает: содержание актуальной нормативно-правовой документации; современную научную и профессиональную терминологию; возможные траектории профессионального развития и самообразования. Умеет: определять актуальность нормативно-правовой документации в профессиональной деятельности. Имеет практический опыт: использования актуальной нормативно-правовой документации по профессии (специальности); применения современной научно профессиональной терминологии; определения траектории профессионального развития и самообразования	Задание 1
ОК 4: Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.	Знает: психологию коллектива; психологию личности; основы проектной деятельности. Умеет: организовывать работу коллектива и команды; взаимодействовать с коллегами, руководством, клиентами. Имеет практический опыт: участия в деловом общении для эффективного решения деловых задач; планирования профессиональной деятельности.	Задание 1
ОК 5: Осуществлять устную и письменную коммуникацию на государственном языке с учётом особенностей социального и культурного контекста.	Знает: особенности социального и культурного контекста; правила оформления документов. Умеет: излагать свои мысли на государственном языке; оформлять документы. Имеет практический опыт: грамотно устно и письменно излагать свои мысли по профессиональной тематике на государственном языке; проявления толерантности в рабочем коллективе.	Задание 1 Отчёт
ОК 6. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей.	Знать: сущность гражданско-патриотической позиции; общечеловеческие ценности; правила поведения в ходе выполнения профессиональной деятельности Уметь: описывать значимость своей профессии; презентовать структуру профессиональной деятельности по профессии	Задание 1

	(специальности)	
ОК 7. Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях.	Знать: правила экологической безопасности при ведении профессиональной деятельности; основные ресурсы, задействованные в профессиональной деятельности; пути обеспечения ресурсосбережения Уметь: соблюдать нормы экологической безопасности; определять направления ресурсосбережения в рамках профессиональной деятельности по профессии (специальности)	Задание 1
ОК 8. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности.	Знать: роль физической культуры в общекультурном, профессиональном и социальном развитии человека; основы здорового образа жизни; условия профессиональной деятельности и зоны риска физического здоровья для профессии (специальности); Средства профилактики перенапряжения Уметь: использовать физкультурно-оздоровительную деятельность для укрепления здоровья, достижения жизненных и профессиональных целей; применять рациональные приемы двигательных функций в профессиональной деятельности; пользоваться средствами профилактики перенапряжения характерными для данной профессии (специальности)	Задание 1
ОК 9: Использовать информационные технологии в профессиональной деятельности.	Знает: современные средства и устройства информатизации; порядок их применения и программное обеспечение в профессиональной деятельности Умеет: применять средства информационных технологий для решения профессиональных задач; использовать современное программное обеспечение Имеет практический опыт: применения средств информатизации и информационных технологий для реализации профессиональной деятельности	Задание 1 Отчёт
ПК 1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием	Знает: основные принципы создания алгоритмов. Умеет: составлять алгоритм работы программного модуля. Имеет практический опыт: написания программного кода.	Задание 1
ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием	Знает: основные принципы технологии структурного и объектно-ориентированного программирования	Задание 1

	<p>Умеет: составлять алгоритм работы программного модуля.</p> <p>Имеет практический опыт: написания программного кода.</p>	
<p>ПК 1.3 Выполнять отладку программных модулей с использованием специализированных программных средств</p>	<p>Знает: основные принципы отладки и тестирования программных продуктов; Умеет: выполнять отладку и тестирование программы на уровне модуля;</p> <p>Имеет практический опыт: использования инструментальных средств на этапе отладки программного продукта;</p>	Задание 1
<p>ПК 1.4 Выполнять тестирование программных модулей</p>	<p>Знает: основные принципы отладки и тестирования программных продуктов; Умеет: выполнять отладку и тестирование программы на уровне модуля;</p> <p>Имеет практический опыт: проведения тестирования программного модуля по определенному сценарию</p>	Задание 1
<p>ПК 1.5 Осуществлять рефакторинг и оптимизацию программного кода</p>	<p>Знает: основные этапы разработки программного обеспечения; основные принципы технологии структурного и объектно-ориентированного программирования;</p> <p>Умеет: осуществлять разработку кода программного модуля на современных языках программирования;</p> <p>Имеет практический опыт: разработки алгоритма поставленной задачи и реализации его средствами автоматизированного проектирования</p>	Задание 1
<p>ПК 1.6 Разрабатывать модули программного обеспечения для мобильных платформ</p>	<p>Знает: Основные этапы разработки программного обеспечения. Основные принципы технологии структурного и объектно-ориентированного программирования.</p> <p>Умеет: Осуществлять разработку кода программного модуля на современных языках программирования. Оформлять документацию на программные средства</p> <p>Имеет практический опыт: разработки мобильных приложений</p>	Задание 1

1. ТЕКУЩИЙ КОНТРОЛЬ ПО ПРАКТИКЕ

Типовые контрольные задания, необходимые для оценки планируемых результатов производственной практики по ПМ.01 Разработка модулей программного обеспечения для

компьютерных систем соответствуют видам работ, определенным в РП профессионального модуля:

ОЦЕНОЧНОЕ СРЕДСТВО Задание 1

1. Цель: освоить необходимые компетенции по виду работ «Разработка модулей программного обеспечения для компьютерных систем»

2. Проверяемые компетенции (код): ОК 1, ОК 2, ОК 3, ОК 4, ОК 5, ОК 6, ОК 7, ОК 8, ОК 9, ОК 10, ПК 1.1, ПК 1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6.

3. Пример оценочного средства:

ПРАКТИЧЕСКИЕ ЗАДАНИЯ

№	Перечень заданий / вопросов
1.	Провести анализ предметной области «Поиск и отбор соискателей на вакантную должность»
2.	Подготовить техническое задание на разработку ПО
3.	Выполнить модификацию хранилища данных
4.	Сформулировать и реализовать запросы к БД согласно потребностям пользователей системы
5.	Создать графический интерфейс для работы с БД
6.	Реализовать основной функционал системы
7.	Подготовить план тестирования ПО, разработать тест-кейсы/ чек-лист
8.	Провести тестирование ПО, оформить отчёт по результатам тестирования
9.	Разработать документацию по эксплуатации ПО. Руководство администратора, руководство пользователя (описав процедуры инсталляции системы и её настройки)
10.	Рассчитать экономическую эффективность ПО

4. Критерии оценивания

Уровень освоения	Критерии
Оценка «отлично»	Практическое задание выполнено в срок и в полном объеме, не содержит фактических ошибок, студент отвечает на все поставленные вопросы, уверенно ориентируется в теоретическом материале и этапах выполнения практического задания
Оценка «хорошо»	Практическое задание выполнено в срок и в полном объеме, не содержит фактических ошибок, но при выполнении задания у студента возникали затруднения и в задании имеются недочеты, не снижающие общего качества работы. Студент отвечает на все поставленные вопросы, уверенно ориентируется в теоретическом материале и этапах выполнения практического задания
Оценка «удовлетворительно»	Практическое задание выполнено в срок, в большей степени, либо имеются 1-2 фактических ошибки. Студент не всегда отвечает на поставленные вопросы, и не всегда ориентируется в теоретическом материале и этапах выполнения практического задания
Оценка «неудовлетворительно»	Практическое задание не выполнено в срок, либо выполнено меньше половины задания, при этом имеются фактические ошибки. Студент не ориентируется в теоретическом материале и этапах выполнения практического задания и затрудняется ответить на поставленные вопросы

2. ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ ПО ПРАКТИКЕ

1. Форма поведения промежуточной аттестации дифференцированный зачет

2. Процедура проведения промежуточной аттестации:

Промежуточная аттестация по производственной практике проходит в форме дифференцированного зачета на основе:

- наличия положительного аттестационного листа;
- заполненного дневника практики, содержащего положительную характеристику студента;
- отчета о практике в соответствии с заданием на практику.

Отчет студента по производственной практике должен максимально отражать его работу в период прохождения производственной практики в соответствии с ее программой.

Отчет по производственной практике состоит из следующих элементов:

- титульный лист;
- содержание;
- введение;
- основная часть в соответствии с программой производственной практики (по профилю специальности);
- заключение;
- список использованной литературы и источников.

Общий объем отчета – 15-25 страниц печатного текста.

Основная часть отчета может быть структурирована по главам. Первая глава, как правило, аналитическая и содержит описание структуры и направлений деятельности организации и т.п., вторая глава, соответственно, содержит описание выполненных работ. Возможно также представление отчета в виде структурированного по разделам описания всех видов выполняемых работ, предусмотренных программой производственной практики (по профилю специальности).

3. ОЦЕНОЧНОЕ СРЕДСТВО Отчет по практике

4. Критерии оценивания отчета по практике

4-бальная шкала (уровень освоения)	Показатели	Критерии
Отлично	1. Соответствие содержания отчета требованиям программы практики; 2. Структурированность и полнота собранного материала; 3. Полнота устного выступления, правильность ответов на вопросы при защите; 4. Обоснованность сформулированных предложений.	При защите отчета студент продемонстрировал глубокие и системные знания, полученные при прохождении практики, свободно оперировал данными исследования и внес обоснованные предложения. Студент правильно и грамотно ответил на поставленные вопросы. Студент получил положительный отзыв от руководителя
Хорошо		При защите отчета студент показал глубокие знания, полученные при прохождении практики, свободно оперировал данными исследования. В отчете были допущены ошибки, которые носят несущественный характер. Студент ответил на поставленные вопросы, но допустил некоторые ошибки, которые при наводящих вопросах были исправлены. Студент получил положительный отзыв от руководителя

Удовлетворительно		Отчет имеет поверхностный анализ собранного материала, нечеткую последовательность его изложения материала. Студент при защите отчета по практике не дал полных и аргументированных ответов на заданные вопросы. В отзыве руководителя имеются существенные замечания.
Неудовлетворительно		Отчет не имеет детализированного анализа собранного материала и не отвечает установленным требованиям. Студент затрудняется ответить на поставленные вопросы или допускает в ответах принципиальные ошибки. В отзыве руководителя имеются существенные критические замечания.

5. Перечень вопросов для проведения устного опроса

Перечень вопросов	
1.	Основные этапы разработки ПО
2.	Состав типовой документации на разработку ПО
3.	Преимущества и недостатки настольных систем и систем с веб-интерфейсом
4.	Особенности обеспечения безопасности персональных данных
5.	Какие технологии использовались при разработке ПО
6.	Какие фреймворки использовались в работе и их назначение
7.	Понятие реинжиниринга
8.	Виды и этапы тестирования ПО

6. Критерии оценивания ответов на устные вопросы:

4-бальная шкала (уровень освоения)	Показатели	Критерии
Отлично	1. Полнота и правильность ответа на поставленный вопрос 2. Степень осознанности, понимания изученного 3. Применения профессиональной терминологии в беседе	Студент полно излагает материал, дает правильное определение основных понятий. Понимает материал, может обосновать свои суждения, привести необходимые примеры. Излагает материал последовательно, используя профессиональную терминологию
Хорошо		Студент полно излагает материал, дает правильное определение основных понятий. Студент допускает некоторые ошибки, имеются недочеты в последовательности излагаемого материала
Удовлетворительно		Студент излагает материал неполно и допускает неточности в определении понятий, не умеет достаточно глубоко и доказательно обосновать свои суждения и привести примеры.

		Излагает материал непоследовательно и допускает ошибки в терминологии.
Неудовлетворительно		Незнание большей части соответствующего вопроса, допускает ошибки в формулировке определений.

7. Методические материалы, определяющие процедуры оценивания результатов производственной практики по ПМ.01 Разработка модулей программного обеспечения для компьютерных систем.

Производственная практика проводится в организациях на основе прямых договоров, заключаемых между институтом и организациями.

Сроки проведения производственной практики (по профилю специальности) устанавливаются в соответствии с учебным планом при освоении профессионального модуля. Период проведения производственной практики включается в график учебного процесса.

Организация (предприятие, учреждение, фирма) как база производственной практики должно:

- иметь сферы деятельности, предусмотренные программой производственной практики;
- располагать квалифицированными кадрами для руководства производственной практикой.

Для прохождения производственной практики студенту предоставляется право выбора базы производственной практики.

При прохождении практики в организациях трудоемкость для студентов составляет 36 часов в неделю.

Студенты при прохождении производственной практики в организациях обязаны:

- полностью выполнять задания, предусмотренные программой практики (по профилю специальности);
- соблюдать действующие в организациях правила внутреннего трудового распорядка;
- изучать и строго соблюдать нормы охраны труда и правила пожарной безопасности.

Организацию и руководство практикой по профилю специальности осуществляют руководители практики от института и от организации.

Направление на производственную практику оформляется приказом по институту с указанием закрепления каждого студента за организацией, вида и сроков прохождения производственной практики, руководителя практики от института.

По результатам производственной практики руководителями практики от института и от организации формируется аттестационный лист, содержащий сведения об уровне освоения обучающимся профессиональных компетенций. Руководитель производственной практики от организации пишет характеристику на студента, отмечает полученные им навыки, отражающие уровень освоения профессиональных компетенций, уровень подготовки и профессиональные качества.

В процессе прохождения производственной практики обучающиеся заполняют дневники практики, в которые ежедневно вносят записи о проделанной работе.

В том случае, если студент проходит производственную практику (в той организации, где нет возможности освоить навыки по выполнению какого-либо вида работ, он должен выполнить индивидуальное задание для полного освоения компетенций (индивидуальные задания представлены в Фонде оценочных средств).