

Документ подписан квалифицированной электронной подписью

Сертификат: 03561B9E0021AE10B3437E5B0E4C07E1A3

Владелец: "АНО ВО «РОССИЙСКИЙ НОВЫЙ УНИВЕРСИТЕТ»"; АН

Действителен с 2022-01-26 по 2022-01-26

**АНО ВО «Российский новый университет»**

**Елецкий филиал Автономной некоммерческой организации высшего образования «Российский новый университет»  
(Елецкий филиал АНО ВО «Российский новый университет»)**

кафедра прикладной экономики

**Рабочая программа учебной дисциплины (модуля)**

Информатика и программирование  
(наименование учебной дисциплины (модуля))

09.03.03 Прикладная информатика  
(код и направление подготовки/специальности)

Прикладная информатика в экономике  
(код и направление подготовки/специальности, в случаях, если программа разработана для разных направлений подготовки/специальностей)

---

Рабочая программа учебной дисциплины (модуля) рассмотрена и утверждена на заседании кафедры «26» января 2022, протокол № 5.

Заведующий кафедрой Прикладной экономики  
(название кафедры)

к.э.н., доцент Преснякова Д.В.

(ученая степень, ученое звание, фамилия и инициалы, подпись заведующего кафедрой)

Елец  
2022 год

## 1. НАИМЕНОВАНИЕ И ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

Настоящая программа предназначена для изучения курса «ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ» студентами, обучающимися по направлению «Прикладная информатика» (код направления 09.03.03).

Цель курса – формирование у студентов основ современной информационной культуры; выработка и закрепление навыков работы на персональном компьютере; выработка умения построения информационных моделей; развитие у студентов навыков программирования; развитие умения применять методы программирования для решения различных прикладных задач; обучение студентов алгоритмам обработки численной и текстовой информации, способам записи и реализации алгоритма на языке программирования высокого уровня.

Задачи курса – формирование у студентов основ современной информационной культуры; выработка и закрепление устойчивых навыков работы на персональном компьютере; выработка умения построения информационных моделей, анализа полученных результатов; обучение студентов основам программирования; обучение методам создания и реализации алгоритма, методам отладки и тестирования программы; формирование представлений о технологиях программирования, проектировании программных продуктов и применении их для разработки программного и информационного обеспечения.

## 2. МЕСТО ДИСЦИПЛИНЫ (МОДУЛЯ) В СТРУКТУРЕ ОП

Учебная дисциплина Информатика и программирование относится к обязательной части учебного плана и изучается на 1 курсе.

2.1. Требования к предварительной подготовке обучающегося: Параллельно с учебной дисциплиной «Информатика и программирование» изучаются дисциплины: «Математика», «Математическая логика и дискретная математика», «Визуальное программирование».

2.2. Дисциплины (модули) и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее: Курс создает предпосылки для освоения таких дисциплин, как «Программная инженерия», «Проектирование информационных систем».

Развитие у обучающихся навыков командной работы, межличностной коммуникации, принятия решений, лидерских качеств обеспечивается чтением лекций, проведением занятий, содержание которых разработано на основе результатов научных исследований, проводимых Университетом, в том числе с учетом региональных особенностей профессиональной деятельности выпускников и потребностей работодателей.

## 3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫЕ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

В результате освоения дисциплины обучающийся по программе бакалавриата должен овладеть следующими компетенциями:

*- Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач (УК-1)*

Планируемые результаты обучения

Формируемая компетенция	Планируемые результаты обучения	Код результата обучения
<b>осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач (УК-1)</b>	<b><u>Знать:</u></b>	
	основные понятия критического анализа, системного подхода для решения поставленных задач	УК-1-31
	базовые алгоритмы обработки числовой и текстовой информации	УК-1-32
	технологии и алгоритмы решения типовых задач	УК-1-33
	основные понятия информатики	УК-1-34
	назначение и основные характеристики устройств компьютера	УК-1-35
	способы кодирования и представления информации в компьютере	УК-1-36
	основные понятия и методы алгоритмизации процессов обработки информации	УК-1-37
	классификацию, основные свойства и специализацию языков программирования	УК-1-38
	<b><u>Уметь</u></b>	
	анализировать информацию и применять системный подход для решения поставленных задач	УК-1-У1
	использовать методы по решению типовых задач	УК-1-У2
	использовать алгоритмы обработки информации для решения поставленных задач	УК-1-У3
	работать в качестве пользователя персонального компьютера	УК-1-У4
	разрабатывать алгоритмы решения экономических и управленческих задач	УК-1-У5
	использовать языки и системы программирования	УК-1-У6
	выбрать метод решения прикладной задачи в соответствии с особенностями программного средства	УК-1-У7
	выбрать инструментальные средства обработки экономических данных в соответствии с поставленной задачей	УК-1-У8
	<b><u>Владеть</u></b>	
	навыками для применения системных подходов	УК-1-В1
	навыками решения практических задач	УК-1-В2
	методологией решения научных и практических задач	УК-1-В3
	навыками работы на ПК	УК-1-В4
	навыками разработки алгоритмов решения экономических и управленческих задач	УК-1-В5
	навыками использования языков и систем программирования	УК-1-В6
	технологией создания документации с помощью текстового процессора Microsoft Word	УК-1-В7
	технологией решения типовых информационных и вычислительных задач с помощью табличного процессора Microsoft Excel	УК-1-В8

**4. ОБЪЕМ ДИСЦИПЛИНЫ (МОДУЛЯ) В ЗАЧЕТНЫХ ЕДИНИЦАХ С  
УКАЗАНИЕМ КОЛИЧЕСТВА АКАДЕМИЧЕСКИХ ЧАСОВ,  
ВЫДЕЛЕННЫХ НА КОНТАКТНУЮ РАБОТУ С ПРЕПОДАВАТЕЛЕМ (ПО  
ВИДАМ УЧЕБНЫХ ЗАНЯТИЙ) И НА САМОСТОЯТЕЛЬНУЮ РАБОТУ  
ОБУЧАЮЩИХСЯ**

Общая трудоемкость дисциплины составляет 8 зачетных единиц (288 часов).

**Общий объем учебной дисциплины (модуля)**

№	Форма обучения	Семестр/сессия, курс	Общая трудоемкость		в том числе контактная работа с преподавателем						СР	Контроль	
			в з.е.	в часах	Всего	Л	Пр	КоР	зачет	Конс			экзамен
1.	Очная	1 семестр	4	144	76	36	36	1,6		2	0,4	34,4	33,6
		2 семестр	4	144	60	24	32	1,6		2	0,4	50,4	33,6
		<i>Итого:</i>	8	288	136	60	68	3,2		4	0,8	84,8	67,2
2.	Заочная	Установочная сессия, 1 курс	1	36	4	4						32	
		Зимняя сессия, 1 курс	3	108	12	4	4	1,6		2	0,4	89,4	6,6
		Летняя сессия, 1 курс	4	144	12	4	4	1,6		2	0,4	125,4	6,6
		<i>Итого:</i>	8	288	28	12	8	3,2		4	0,8	246,8	13,2

**Распределение учебного времени по темам и видам учебных занятий  
очная форма**

№	Наименование разделов, тем учебных занятий	Всего часов	Контактная работа с преподавателем						СР	Формируемые результаты обучения
			Всего	Л	Пр	КоР	Конс	Экз		
Представление информации в ЭВМ										
1.	Представления информации в ЭВМ	4	2	1	1				2	УК-1-32 УК-1-У1
Средства реализации информационных процессов										
2.	Средства реализации информационных процессов	4	2	1	1				2	УК-1-32 УК-1-33
Программное обеспечение (ПО) ПЭВМ										
3.	Программное обеспечение (ПО) ПЭВМ	5	4	2	2				1	УК-1-31 УК-1-В1
Управление персональным компьютером										
4.	Управление персональным компьютером	5	4	2	2				1	УК-1-32 УК-1-У4
Программные средства электронного офиса										
5.	Программные средства электронного офиса	5	4	2	2				1	УК-1-31 УК-1-В3

Текстовый процессор MS Word										
6.	Текстовый процессор MS Word	5	4	2	2				1	УК-1-В3 УК-1-У1
Табличный процессор MS Excel										
7.	Табличный процессор MS Excel	8	4	2	2				4	УК-1-В2 УК-1-У1 УК-1-32
Алгоритмизация и основы программирования										
8.	Алгоритмизация и основы программирования	5	4	2	2				1	УК-1-36 УК-1-У1
Языки и современные среды программирования										
9.	Языки и современные среды программирования.	5	4	2	2				1	УК-1-31 УК-1-33 УК-1-37
Язык программирования С. Линейные алгоритмы										
10.	Язык программирования С. Линейные алгоритмы	7,4	4	2	2				3,4	УК-1-32 УК-1-У2
Алгоритм с ветвлением										
11.	Алгоритм с ветвлением.	8	4	2	2				4	УК-1-33 УК-1-У2
Циклический алгоритм										
12.	Циклический алгоритм.	9	4	2	2				5	УК-1-33 УК-1-В8
Статические и динамические массивы										
13.	Статические и динамические массивы.	5	4	2	2				1	УК-1-32 УК-1-В1
Организация интерфейса пользователя										
14.	Организация интерфейса пользователя.	5	4	2	2				1	УК-1-У1 УК-1-В3
Принципы работы с файлами										
15.	Принципы работы с файлами.	5	4	2	2				1	УК-1-У3 УК-1-В3
Графика										
16.	Графика.	5	4	2	2				1	УК-1-В4 УК-1-У3
Пользовательские функции										
17.	Пользовательские функции.	5	4	2	2				1	УК-1-32 УК-1-У3
Типы данных, определяемые пользователем										
18.	Типы данных, определяемые пользователем.	5	4	2	2				1	УК-1-У5 УК-1-В6
Архитектура и возможности языков высокого уровня										
19.	Архитектура и возможности языков высокого уровня.	6	4	2	2				2	УК-1-У5 УК-1-В6
Промежуточная аттестация (экзамен)										

	<b>Промежуточная аттестация (экзамен)</b>	4	4		1,6	2	0,4			
Основные понятия объектно- ориентированного подхода к программированию										
20.	Основные понятия объектно- ориентированного подхода к программированию.	6	4	2	2				2	УК-1-37 УК-1-В8
Инициализация объектов. Конструкторы и деструкторы										
21.	Инициализация объектов. Конструкторы и деструкторы.	10	4	2	2				6	УК-1-У3 УК-1-31
Перегрузка операций										
22.	Перегрузка операций.	8	6	4	2				2	УК-1-В3 УК-1-В2
Стандартная библиотека										
23.	Стандартная библиотека	12,4	6	2	4				6,4	УК-1-У3 УК-1-В2
Принципы наследования										
24.	Принципы наследования.	10	4	2	2				6	УК-1-В4 УК-1-36
Шаблоны функций. Библиотека шаблонов STL										
25.	Шаблоны функций. Библиотека шаблонов STL.	8	4	2	2				4	УК-1-В2 УК-1-В1
Обработка исключительных ситуаций										
26.	Обработка исключительных ситуаций.	12	6	2	4				6	УК-1-У5 УК-1-У3
Проектирование программных комплексов										
27.	Проектирование программных комплексов.	10	6	2	4				4	УК-1-В7 УК-1-У8
Разработка Windows - приложений										
28.	Разработка Windows - приложений.	10	6	2	4				4	УК-1-В1 УК-1-В2
Графика в Windows - приложениях										
29.	Графика в Windows - приложениях.	12	6	2	4				6	УК-1-У4 УК-1-В2
Обзор современных средств программирования										
30.	Обзор современных средств программирования	8	4	2	2				4	УК-1-33 УК-1-В1
Промежуточная аттестация (экзамен)										
	<b>Промежуточная аттестация (экзамен)</b>	4	4			1,6	2	0,4		

**заочная форма обучения**

№	Наименование разделов, тем учебных занятий	Всего часов	Контактная работа с преподавателем					СР	Формируемые результаты обучения
			Всего	Л	Пр	КоР	Конс		
<b>Представление информации в ЭВМ</b>									
1.	Представления информации в ЭВМ	8	1	1				7	УК-1-32 УК-1-У1
<b>Средства реализации информационных процессов</b>									
2.	Средства реализации информационных процессов	8	1	1				7	УК-1-32 УК-1-33
<b>Программное обеспечение (ПО) ПЭВМ</b>									
3.	Программное обеспечение (ПО) ПЭВМ	8	1	1				7	УК-1-31 УК-1-В1
<b>Управление персональным компьютером</b>									
4.	Управление персональным компьютером	12	1	1				11	УК-1-32 УК-1-У4
<b>Программные средства электронного офиса</b>									
5.	Программные средства электронного офиса	18						18	УК-1-31 УК-1-В3
<b>Текстовый процессор MS Word</b>									
6.	Текстовый процессор MS Word	20	2	1	1			18	УК-1-В3 УК-1-У1
<b>Табличный процессор MS Excel</b>									
7.	Табличный процессор MS Excel	19	1		1			18	УК-1-В2 УК-1-У1 УК-1-32
<b>Алгоритмизация и основы программирования</b>									
8.	Алгоритмизация и основы программирования	23	2	1	1			21	УК-1-36 УК-1-У1
<b>Языки программирования и современные среды программирования</b>									
9.	Языки программирования и современные среды программирования.	11,8	1	1				10,8	УК-1-31 УК-1-33 УК-1-37
<b>Язык программирования Си. Линейные алгоритмы</b>									
10.	Язык программирования Си. Линейные алгоритмы	11	1	1				10	УК-1-32 УК-1-У2
<b>Алгоритм с ветвлением</b>									
11.	Алгоритм с ветвлением.	11	1	1				10	УК-1-33 УК-1-У2
<b>Циклический алгоритм</b>									
12.	Циклический алгоритм.	11	1	1				10	УК-1-33 УК-1-В8
<b>Статические и динамические массивы</b>									

13.	Статические и динамические массивы.	4							4	УК-1-32 УК-1-В1
Организация интерфейса пользователя										
14.	Организация интерфейса пользователя.	4							4	УК-1-У1 УК-1-В3
Принципы работы с файлами										
15.	Принципы работы с файлами.	5							5	УК-1-У3 УК-1-В3
Графика										
16.	Графика.	5							5	УК-1-В4 УК-1-У3
Пользовательские функции										
17.	Пользовательские функции.	10							10	УК-1-32 УК-1-У3
Типы данных, определяемые пользователем										
18.	Типы данных, определяемые пользователем.	5							5	УК-1-У5 УК-1-В6
Промежуточная аттестация (экзамен)										
19.	<b>Промежуточная аттестация (экзамен)</b>	4	4		1,6	2	0,4			
Архитектура и возможности языков высокого уровня										
20.	Архитектура и возможности языков высокого уровня.	5							5	УК-1-У5 УК-1-В6
Основные понятия объектно-ориентированного подхода к программированию										
21.	Основные понятия объектно-ориентированного подхода к программированию.	6							6	УК-1-37 УК-1-В8
Инициализация объектов. Конструкторы и деструкторы										
22.	Инициализация объектов. Конструкторы и деструкторы.	6							6	УК-1-У3 УК-1-31
Перегрузка операций										
23.	Перегрузка операций.	6							6	УК-1-В3 УК-1-В2
Стандартная библиотека										
24.	Стандартная библиотека	6							6	УК-1-У3 УК-1-В2
Принципы наследования										
25.	Принципы наследования.	6	1		1				5	УК-1-В4 УК-1-36
Шаблоны функций. Библиотека шаблонов STL										
26.	Шаблоны функций. Библиотека шаблонов STL.	6							6	УК-1-В2 УК-1-В1
Обработка исключительных ситуаций										



27.	Обработка исключительных ситуаций.	6							6	УК-1-У5 УК-1-У3
Проектирование программных комплексов										
28.	Проектирование программных комплексов.	6	1		1				5	УК-1-В7 УК-1-У8
Разработка Windows - приложений										
29.	Разработка Windows - приложений.	6	1		1				5	УК-1-В1 УК-1-В2
Графика в Windows - приложениях										
30.	Графика в Windows - приложениях.	6	1		1				5	УК-1-У4 УК-1-В2
Обзор современных средств программирования										
31.	Обзор современных средств программирования	8	3	2	1				5	УК-1-З3 УК-1-В1
Промежуточная аттестация (экзамен)										
32.	<b>Промежуточная аттестация (экзамен)</b>	4	4			1,6	2	0,4		

## 5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ), СТРУКТУРИРОВАННОЕ ПО ТЕМАМ (РАЗДЕЛАМ)

### ТЕМА 1. Представление информации в ЭВМ.

Информатика. Понятие информации. Общая характеристика процессов сбора, передачи и хранения информации. Технические и программные средства организации вычислительных процессов. Единицы измерения количества информации. Основные принципы кодирования информации: представление текстовой, графической, видео информации. Системы счисления Двоичная система кодирования символов. Таблица ASCII кодов. Кодировки ANSI и Unicode.

### ТЕМА2. Средства реализации информационных процессов.

ЭВМ как средство обработки информации. Структурная схема и принципы работы ЭВМ. Классификация ЭВМ. Назначение и характеристики основных узлов ПЭВМ. Устройства внешней памяти. Накопители на жестких и гибких магнитных дисках. Другие накопители и носители информации. Видеосистема. Устройства ввода-вывода информации: коммуникационное оборудование, назначение и основные типы (модемы, сетевые адаптеры). Телекоммуникационные средства обмена информацией.

### ТЕМА3. Программное обеспечение (ПО)ПЭВМ.

Понятие и классификация ПО ПЭВМ. Системное (базовое) и прикладное ПО. Понятие об информационных системах и технологиях,

банках и базах данных и знаний, системах искусственного интеллекта, информационных технологиях на сетях, гипер- и мультисредах. Операционные системы и программы оболочки, их назначение и классификация. Основные функции операционных систем. Стандартные приложения. Настройка ОС. Использование справочной системы. Начальные сведения об архитектуре ОС. Понятие о файловой системе. Правила составления имен файлов, типы файлов, шаблоны (маски) имени файлов. Исполняемые файлы (программы) и файлы документов. Основные операции с файлами и каталогами: создание, просмотр, копирование, переименование и удаление.

#### **ТЕМА4. Управление персональным компьютером.**

Служебные программы (утилиты) и их назначение. Использование сервисных программ. Резервирование информации. Программы архивации данных. Понятие компьютерного вируса, классификация компьютерных вирусов. Защита от компьютерных вирусов. Антивирусные средства, их классификация и использование. Проблема информационной безопасности информационных систем. Угрозы безопасности информации. Технические, организационные и программные средства обеспечения сохранности информации и защиты ее от несанкционированного доступа.

#### **ТЕМА 5. Программные средства электронного офиса.**

Понятие электронного офиса. **Microsoft Office**. Состав программных средств - текстовый редактор **Word**, электронные таблицы **Excel**, система управления базами данных **Access**, система подготовки презентаций **PowerPoint**. Обмен данными между приложениями и их взаимодействие. Организация совместной работы пользователей средствами **Microsoft Office**.

#### **ТЕМА6. Текстовый процессор MS Word.**

Назначение и классификация текстовых редакторов. Общая функциональная характеристика редактора **Word**. Структура окна, состав меню, панели инструментов. Настройка и создание новых панелей инструментов. Настройка параметров. Основные понятия и терминология работы в текстовом редакторе. Технология ввода и редактирования текста. Форматирование абзацев и списков. Создание и применение стилей. Таблицы в текстовых документах. Форматирование таблиц. Сортировка данных в таблицах. Вычисления в таблицах, применение формул. Диаграммы. Графические объекты в текстовых документах. Работа с базами данных. Управление информацией в источнике данных: поиск, редактирование, сортировка. Ассистент слияния. Использование форм и полей **Word**. Структура документа. Создание составного документа, режим главного документа. Создание оглавления и предметного указателя.

#### **ТЕМА7. Табличный процессор MS Excel.**

Назначение и классификация электронных таблиц. Общая функциональная характеристика процессора **Excel**. Структура окна, состав меню, панели инструментов. Настройка и создание новых панелей инструментов. Настройка параметров. Книга **Excel**, листы. Ячейка, ввод,

редактирование и форматы данных. Авто-заполнение ячеек таблицы. Абсолютные и относительные адреса, имена ячеек. Вычисления на рабочем листе. Ввод, редактирование и копирование формул и функций. Построение диаграмм. Тип и вид диаграммы. Мастер диаграмм. Работа со списками (базами данных). Сортировка, фильтрация записей. Форма. Подведение промежуточных итогов. Сводные таблицы. Мастер сводных таблиц.

#### **ТЕМА8. Алгоритмизация и основы программирования.**

Понятие алгоритма. Линейные алгоритмы, алгоритмы с ветвлением, циклические алгоритмы. Представление алгоритмов в виде блок-схемы и псевдокода. Элементарные алгоритмические конструкции. Типовые алгоритмы – суммирование, поиск максимума (минимума). Алгоритмы сортировки – подсчетом, методом вставки, методом пузырька. Алгоритм быстрой сортировки. Алгоритмы поиска – последовательный поиск, ступенчатый поиск, бинарный поиск. Динамические структуры данных: списки, очереди, стек. Оценка эффективности алгоритма.

#### **ТЕМА 9. Языки программирования и современные среды программирования.**

Языки программирования и их классификация. Высокоуровневые языки программирования. Сравнительный обзор высокоуровневых языков программирования и общие понятия высокоуровневых языков программирования – типы данных, переменные, выражения, операторы ветвления, циклы, функции ит.д. Трансляция

программы. Интерпретаторы и компиляторы. Исходный текст. Модули. Библиотеки. Исполняемая программа. Отладка и тестирование. Отладчик. Интегрированная среда программирования. Встроенный редактор. Препроцессор. Трансляция и сборка программы. Встроенный отладчик.

#### **ТЕМА10. Язык программирования Си. Линейный алгоритм.**

Алфавит языка, лексемы. Ключевые слова и идентификаторы. Типы данных. Директивы препроцессора *include* *define*. Понятие функции. Структура программы. Переменные и константы. Глобальные и локальные переменные. Область видимости переменных. Функции ввода/вывода. Операция присваивания. Арифметические выражения и операции. Операции сдвига. Преобразование типов данных. Библиотечные функции. Математические функции.

#### **ТЕМА11. Алгоритм с ветвлением.**

Операции отношения, условные выражения и логические операции. Операторы ветвления: условный оператор **if**. Полное ветвление. Неполное ветвление. Оператор множественного выбора **switch**. Конструкции **case** **default**. Оператор прерывания **break**. Передача управления: оператор безусловного перехода **goto**. Метки.

#### **ТЕМА12. Циклический алгоритм.**

Циклы с предусловием и постусловием. Реализация циклов с помощью операторов ветвления и передачи управления. Операторы цикла **while**, **dowhile**, **for**. Взаимное приведение циклов *for* *while*. Оператор продолжения

**continue.** Прерывание циклов. Вложенные циклы. Понятие об итерации. Рекурсивные и циклические алгоритмы.

### **ТЕМА13. Статические и динамические массивы.**

Массивы как однородные статические структуры данных. Числовые массивы. Алгоритмы обработки массивов: суммирование, поиск максимума (минимума), сортировка, поиск. Массивы различной размерности. Заполнение и инициализация массивов. Многомерные массивы. Алгоритмы работы с матрицами. Указатели. Динамические массивы. Арифметика указателей. Связь между массивами и указателями. Строки. Обработка строк как массивов символов. Библиотечные функции обработки строк. Массивы строк.

### **ТЕМА14. Организация интерфейса пользователя.**

Организация и средства человеко-машинного интерфейса. Работа с экраном в текстовом режиме. Задание цвета. Интерфейс командной строки. Горячие клавиши. Пассивное меню. Активное меню. Создание формы ввода на экран.

### **ТЕМА15. Принципы работы с файлами.**

Файлы. Типы файлов. Организация работы с файлами. Библиотечные функции, предназначенные для работы с файлами.

### **ТЕМА16. Графика.**

Работа с экраном в графическом режиме. Выбор графического драйвера и моды. Инициализация графического режима. Выбор инструментов рисования: пера, кисти (заливки), шрифта. Функции вывода графики. Графические примитивы.

### **ТЕМА 17. Пользовательские функции.**

Библиотечные и пользовательские функции. Прототип и описание функции. Возвращаемое значение. Передача параметров по значению и по адресу. Ссылки. Массивы в качестве параметров. Параметры со значениями по умолчанию. Функции в качестве параметров. Понятие функтора. Рекурсия. Перегрузка функций.

### **ТЕМА 18. Типы данных, определяемые пользователем.**

Пользовательские типы данных. Структуры. Массивы структур. Инициализация структур. Поля (данные) структур. Операции доступа данным. Переименование типов. Перечисления. Объединения. Битовые поля.

### **ТЕМА 19. Экзамен**

**ВОПРОСЫ К ЭКЗАМЕНУ - ПЕРВЫЙ СЕМЕСТР**

1. Что такое информация и данные?
2. Какие системы счисления используются в ПЭВМ?
3. Какие единицы информации используются для измерения емкости жестких дисков?
4. Назовите основные составные части персонального компьютера.
5. От чего зависит быстродействие персонального компьютера?
6. Какие виды памяти Вы знаете?
7. Что такое внешняя память ПЭВМ?
8. Что такое программное обеспечение ПЭВМ?

9. Что входит в состав системного (базового) программного обеспечения? 10. Каковы основные функции операционной системы? 11. Что такое файловая система? 12. Какие существуют ограничения на имена файлов в DOS? 13. Для чего используется шаблоны (маски)? 14. Что такое дефрагментация файлов и каково ее назначение? 15. В каких случаях нужна архивация данных, и какие существуют программы архивации? 16. Что такое компьютерный вирус и как он проявляется? 17. Какие мероприятия необходимы для защиты от компьютерных вирусов? 18. Как проверить дискету, диск, оперативную память на наличие вируса? 19. Назовите основные угрозы безопасности экономической информации? 20. Что такое операционные оболочки и их назначение? 21. Какие операции можно осуществлять с файлами в Norton Commander? 22. Что такое фильтрация файлов? Как она реализуется в Norton Commander? 23. Чем отличается перемещение от копирования файлов? 24. Как отыскать файл средствами Norton Commander? 25. В чем заключается процесс отладки программы? 26. Какие существуют способы копирования файлов в Windows? 27. Что такое многозадачный режим работы? 28. Можно ли восстановить удаленный в Windows файл? 29. Как в Windows определить размер свободного дискового пространства? 30. Как просмотреть содержимое буфера обмена Windows? 31. Функциональные возможности Проводника и системной папки Мой компьютер. 32. Как можно осуществить настройку Word для потребностей конкретного пользователя? 33. Перечислите параметры форматирования абзацев. 34. Укажите основные характеристики шрифтов. 35. Что такое стили в редакторе Word? 36. Как выполнить копирование фрагмента документа (текста или рисунка) Word в другой документ с помощью буфера обмена. 37. Как просуммировать данные в строках и столбцах таблицы в Word? 38. Как создать диаграмму в редакторе Word? 39. Какие объекты могут быть вставлены в документ Word? 40. Электронные таблицы Excel и их назначение. 41. Чем отличается абсолютный адрес ячейки Excel от относительного? 42. Что такое рабочий лист и рабочая книга? 43. Как пользоваться мастером функций? 44. Какие возможности предоставляет Excel при работе с диаграммами? 45. Что такое фильтрация данных? Что такое пользовательский фильтр? 46. Какую таблицу Excel, программа рассматривает как базу данных (список)? 47. Как подвести промежуточные итоги в списке? 48. Что такое сводная таблица, когда используется и как ее создать? 49. Что входит в состав электронного офиса? 50. Как можно организовать совместную работу в сети сотрудников офиса? 51. Какое обеспечение необходимо для работы в локальной вычислительной сети? 52. Какие функции выполняет сетевая операционная система? 53. Каковы преимущества работы в локальной сети? 54. Что такое сервер? 55. Какие виды угроз безопасности информации вы знаете? 56. В чем состоят мероприятия по обеспечению безопасности информации? 57. Перечислите методы и средства защиты информации. 58. Какие существуют способы описания алгоритмов? 59. Что такое разветвляющийся алгоритм? Приведите пример. 60. В каких случаях необходима циклическая обработка данных? Приведите пример. 61. Для чего служат директивы препроцессора? 62. В чем отличия между интерпретатором и компилятором? 63. Что такое язык программирования высокого уровня. Приведите пример. 64. Что такое язык программирования низкого уровня. Приведите пример. 65. Что содержит заголовочный файл? 66. Поясните концепцию типов данных. 67. В чем отличие констант от переменных? 68. Что такое глобальная переменная? 69. В чем разница между значением и именем переменной? 70. Чем отличаются формальные переменные от фактических?

**ТЕМА20. Архитектура и возможности языков высокого уровня.**  
Принципы объектно-ориентированного программирования.

Функциональное программирование. Визуальное программирование. Программирование баз данных. Программирование в компьютерных сетях. Программирование мультимедиа-сред.

**ТЕМА21. Основные понятия объектно-ориентированного подхода к программированию..**

Понятие инкапсуляции, полиморфизма, наследования, модульности и абстракции объектов. Понятие класса и объекта. Объявление класса. Данные-члены (свойства) и функции-члены (методы). Доступ к членам класса: открытые, закрытые и защищенные члены класса. Объекты. Обращение к членам объекта. Дружественные классы и функции. Передача объекта в качестве параметра функции.

**ТЕМА22. Инициализация объектов. Конструкторы и деструкторы.**

Конструкторы. Свойства конструкторов. Конструктор по умолчанию. Параметры конструкторов. Конструктор преобразования. Деструктор. Вызов деструктора. Виртуальный деструктор. Указатель **this**. Оператор присваивания. Конструктор копий. Поразрядное (буквальное) копирование и развернутое копирование. Реализация копирования через присваивание.

**ТЕМА23. Перегрузка операций.**

Понятие перегрузки. Перегрузка функций. Перегрузка операций. Перегружаемые операции. Операции, не допускающие перегрузки. Перегрузка унарных операций. Перегрузка бинарных операций. Перегрузка операций как методов класса и как дружественных функций. Сравнение способов перегрузки операций.

**ТЕМА 24. Стандартная библиотека.**

Строки C++. Класс `string`. Конструкторы, операции и методы класса строк. Поточные классы. Предопределенные объекты `cin` `cout`. Форматирование потоков. Методы обмена с потоками. Файловые потоки.

**ТЕМА 25. Принцип наследования**

Базовый и производный классы. Режимы доступа к членам базового класса. Конструкторы и деструкторы производных классов. Полиморфизм. Виртуальные методы. Множественное наследование. Использование виртуальных базовых классов. Чисто виртуальная функция. Абстрактные классы.

**ТЕМА 26. Шаблоны функций. Библиотека шаблонов STL.**

Перегрузка функций. Шаблоны функций. Создание шаблонов функций. Классы-шаблоны. Создание и использование шаблонов классов. Библиотека STL. Алгоритмы и контейнеры. Вектор, список, стек, очередь.

**ТЕМА27. Обработка исключительных ситуаций.**

Понятие исключения. Генерация исключения, обработчики исключений. Абсолютный обработчик. Стандартный обработчик и его перегрузка. Исключения в классах. Исключения в функциях.

**ТЕМА28. Проектирование программных комплексов.**

Этапы создания объектно-ориентированных программ. Понятие проекта. Понятие компонента. Реализация проекта. Отладка проектов. Средства проектирования.

**ТЕМА29. Разработка Windows -приложений.**

Структура и выполнение Windows –программ. Инициализация окон. Обработка сообщений. Управление памятью. Классы приложений. Классы-шаблоны и виртуальные функции для работы с Windows. Окно и его ресурсы. Методы создания ресурсов окна.

**ТЕМА30. Графика в Windows -приложениях.**

Общие принципы создания графических программ. Вывод графических образов на экран. Хранение графических образов. Программирование графики для Windows. Графические объекты Windows. Библиотеки графических образов.

**ТЕМА31. Обзор современных средств программирования.**

Методика проектирования, ориентированная на потоки данных. Визуальное программирование: компоненты и события, обработка событий. Многооконный интерфейс и средства его создания. Средства автоматизации программирования (CASE - технологии).

**ТЕМА32. Экзамен.**

ВОПРОСЫ К ЭКЗАМЕНУ \_ ВТОРОЙ СЕМЕСТР

1. Что такое указатель? 2. Какие операции применяются с указателями? 3. Зачем нужны операции преобразования типа? Приведите пример. 4. Перечислите операторы выбора. 5. Почему в операторе множественного выбора switch надо использовать оператор break? 6. Зачем в операторе множественного выбора применяется ключевое слово default? 7. К чему приведет ошибка в написании ключевого слова default? 8. Перечислите операторы цикла. 9. В чем отличие между инициализацией и присваиванием? 10. Перечислите арифметические операции. 11. Перечислите операции отношения. 12. Перечислите логические операции. 13. Что такое тернарная операция? 14. Поясните концепцию модульного программирования. 15. Что такое массив? Приведите пример объявления массива. 16. Как можно инициализировать массив? 17. Какие методы упорядочивания массива Вы знаете? 18. Какие методы поиска в массиве Вы знаете? 19. Что такое динамический массив? 20. Что такое многомерный массив? Приведите пример. 21. Какие поразрядные (битовые) операции Вы знаете? 22. Что такое функция? 23. Что такое параметр функции? 24. В чем отличие между формальными и фактическими параметрами? 25. Что такое возвращаемое значение функции? 26. Что такое прототип функции. Когда он необходим? 27. В чем различие в передаче параметра по значению и по адресу? 28. В чем различие в передаче параметра по значению и по ссылке? 29. Допустимы ли у параметров значения по умолчанию? 30. Какие математические функции Вы знаете? 31. Какие функции ввода/вывода Вы знаете? 32. Какие функции обработки строк Вы знаете? 33. Что такое графический примитив? 34. Что такое активное меню? 35. Что такое пассивное меню? 36. Что такое пользовательский тип данных. Приведите пример. 37. Что такое структура в С#?

Приведите пример. 38. Что означает перечисление? 39. Что такое файл прямого доступа? 40. Какие функции работы с файлами Вы знаете? 41. В чем состоит концепция объектно-ориентированного программирования? 42. Перечислите различия между структурой и классом в языке C++. 43. Поясните понятие инкапсуляции. 44. Поясните понятие полиморфизма. 45. Для чего нужен конструктор? 46. Сколько конструкторов может быть определено в классе? 47. Когда используется операция присваивания и конструктор копий? 48. Что такое дружественная функция? 49. В каких случаях обязательно явно определять в классе операцию присваивания и конструктор копий? 50. В каких случаях обязательно явно определять в классе деструктор? 51. Можно ли операцию меньше определить, как метод класса? 52. Можно ли операцию меньше определить, как дружественную функцию? 53. Можно ли операцию присваивания определить, как дружественную функцию? 54. Что такое конструктор преобразования? 55. Какие члены класса наследуются? 56. Какие члены класса не наследуются? 57. Приведите пример иерархии классов. 58. Что такое множественное наследование? 59. Что такое ромбовидное наследование? 60. Что такое виртуальный метод? 61. Зачем нужен виртуальный деструктор? 62. Поясните понятие позднего связывания. 63. В чем отличие перегрузки от полиморфизма? 64. Что такое шаблон функции? Приведите пример. 65. Что такое шаблон класса? Приведите пример. 66. Что такое контейнер. Перечислите контейнеры библиотеки STL. 67. Что такое итератор? 68. Зачем нужна обработка исключений? 69. Что такое абсолютный обработчик? 70. Какие операции определены в классе string? 71. Поясните смысл операций ">>" и "<<" в Си. Почему в Си++ они могут применяться в инструкциях ввода/вывода? 72. Для чего применяются манипуляторы потоков? 73. Как можно форматировать потоки? 74. Какие классы окон вы знаете? 75. Какие методы класса приложения Вы знаете? 76. Зачем нужны процедуры (функции) обработки событий? 77. Какие виды ресурсов Вы знаете? 78. Что такое контекст устройства? 79. Какие методы контекста устройства Вы знаете? 80. Какие параметры пера можно изменить? 81. Какие параметры кисти можно задать? 82. Что такое графический примитив? 83. Что такое проект? 84. Какие стандартные типы проектов Вы знаете? 85. Перечислите основные файлы проекта в изучаемой в курсе среде программирования. 86. Что такое компонент. 87. Какие компоненты Вам известны? 88. Что такое свойства компонента? 89. Что такое событие? 90. Чем отличается визуальный компонент от не визуального? 91. Какие стандартные диалоги Вам известны? 92. Что такое модальный диалог?

### *Планы практических занятий*

#### *очная форма обучения*

##### **Тема 1. Представление информации в ЭВМ.**

Время - 1 час.

##### Основные вопросы:

1. Кодирование информации 2. представление чисел в различных системах счисления.

##### **Тема 2. Средства реализации информационных процессов. .**

Время - 1 час.

##### Основные вопросы:

1, Состав ПЭВМ 2. Коммуникационное оборудование.

##### **Тема 3. Программное обеспечение (ПО) ПЭВМ. .**

Время - 2 час.

##### Основные вопросы:



**Тема 4. Управление персональным компьютером. .**

Время - 2 час.

Основные вопросы:

1. Операционные системы и программы оболочки. 2. Работа с файловой системой. 3. Настройка ОС. Использование справочной системы.

**Тема 5. Программные средства электронного офиса. .**

Время - 2 час.

Основные вопросы:

1. Состав программных средств электронного офиса. 2. Система подготовки презентаций PowerPoint. 3. Настольная издательская система Publisher. 4. Обмен данными между приложениями и их взаимодействие.

**Тема 6. Текстовый процессор MS Word. .**

Время - 2 час.

Основные вопросы:

1.Общая функциональная характеристика редактора Word. Настройка параметров. 2.Технология ввода и редактирования текста. Форматирование абзацев и списков. Создание и применение стилей. 3.Таблицы в текстовых документах. Диаграммы. Графические объекты в текстовых документах. 4.Работа с базами данных. Управление информацией в источнике данных: поиск, редактирование, сортировка. Ассистент слияния. Использование форм и полей Word.

**Тема 7. Табличный процессор MS Excel. .**

Время - 2 час.

Основные вопросы:

1.Структура окна, состав меню, панели инструментов. Настройка и создание новых панелей инструментов. Настройка параметров. 2. Автозаполнение ячеек таблицы. Абсолютные и относительные адреса, имена ячеек. 3. Ввод, редактирование и копирование формул и функций. 4. Построение диаграмм. Тип и вид диаграммы. Мастер диаграмм. 5. Работа со списками (базами данных). Сортировка, фильтрация записей. Форма. Подведение промежуточных итогов. Сводные таблицы. Сводные диаграммы.

**Тема 8. Алгоритмизация и основы программирования. .**

Время - 2 час.

Основные вопросы:

1.Разработка алгоритмов

**Тема 9. Языки программирования и современные среды программирования.**

Время - 2 час.

Основные вопросы:

1.Этапы разработки и выполнения программы 2. Трансляторы. Этапы трансляции программы 3. Выполнение программы в режиме интерпретации

**Тема 10. Язык программирования Си. Линейный алгоритм. .**

Время - 2 час.

Основные вопросы:

1. Язык программирования С и С++. История возникновения. Алфавит языка. 2. Основные типы данных. Структура программы. Стандартные библиотеки. 3. Основные операторы ввода-вывода (printf, scanf). 4. Ввод – вывод данных в языке С++ (cin, cout). 5. Реализация линейных алгоритмов на С++.

**Тема 11. Алгоритм с ветвлением. .**

Время - 2 час.

Основные вопросы:

1. Язык программирования С и С++. История возникновения. Алфавит языка. 2. Основные типы данных. Структура программы. Стандартные библиотеки. 3. Основные операторы ввода-вывода (printf, scanf). 4. Ввод – вывод данных в языке С++ (cin, cout). 5. Реализация разветвляющихся алгоритмов.

**Тема 12. Циклический алгоритм.**

Время - 2 час.

Основные вопросы:

1. Операторы цикла с предусловием и постусловием. Синтаксис, структура операторов 2. Оператор с предусловием for. Примеры решения задач. 3. Оператор с предусловием while. Примеры решения задач. 4. Оператор с постусловием do-while. Примеры решения задач.

**Тема 13. Статические и динамические массивы.**

Время - 2 час.

Основные вопросы:

1. Массивы, определение. Одномерные, двумерные массивы. 2. Основные операции: ввод, обработка, вывод элементов массива. 3. Статические массивы. 4. Указатели, основные типы и действия над указателями. 5. Динамические массивы.

**Тема 14. Организация интерфейса пользователя.**

Время - 2 час.

Основные вопросы:

1. Понятие пользовательского интерфейса (ПИ). 2. Виды ПИ. 3. Основные элементы ПИ. 4. Разработка ПИ.

**Тема 15. Принципы работы с файлами. .**

Время - 2 час.

Основные вопросы:

1. Определение файла. Физический и программный файл. 2. Виды файлов: типизированные, текстовые. 3. Классификация файлов по типу доступа: прямого, последовательного доступа. 4. Основные операции с файлами. 5. Обработка файлов

**Тема 16. Графика.**

Время - 2 час.

Основные вопросы:

1. Выбор инструментов рисования. 2. Функции вывода графики. 3. Графические примитивы.

**Тема 17. Пользовательские функции.**

Время - 2 час.

Основные вопросы:

1. Понятие функций в С++. 2. Пользовательские функции: структура, форма записи. 3. Формальные и фактические параметры, локальные и глобальные переменные. 4. Вызов функции по значению и по ссылке. 5. Примеры разработки программ. 6. Рекурсивные функции.

**Тема 18. Типы данных, определяемых пользователем.**

Время - 2 час.

Основные вопросы:

1. Организация сложных типов данных.
2. Операции доступа к данным.
3. Переименование типов.

**Тема 20. Архитектура и возможности языков высокого уровня.**

Время - 2 час.

Основные вопросы:

1. Принципы выбора средств программирования для реализации прикладных задач.
2. Анализ современных языков высокого уровня.

**Тема 21. Основные понятия объектно-ориентированного подхода к программированию.**

Время - 2 час.

Основные вопросы:**Тема 22. Инициализация объектов. Конструкторы и деструкторы.**

Время - 2 час.

Основные вопросы:

1. Основные члены класса: конструкторы, деструкторы.
2. Назначение и описание конструкторов и деструкторов.
3. Свойства конструкторов и деструкторов.
4. Использование конструкторов и деструкторов при разработке программ

**Тема 23. Перегрузка операций.**

Время - 2 час.

Основные вопросы:

1. Применение перегрузки в программах.
2. Разработка примеров.

**Тема 24. Стандартная библиотека.**

Время - 4 час.

Основные вопросы:

1. Решение задач по теме.

**Тема 25. Принцип наследования.**

Время - 2 час.

Основные вопросы:

1. Понятие наследования, свойства.
2. Единичное и множественное наследование.
3. Использование механизма наследования при разработке программ

**Тема 26. Шаблоны функций. Библиотека шаблонов STL .**

Время - 2 час.

Основные вопросы:

1. Перегрузка функций. Шаблоны функций.
2. Классы-шаблоны. Создание и использование шаблонов классов.
3. Библиотека STL.
4. Алгоритмы и контейнеры.
5. Вектор, список, стек, очередь.

**Тема 27. Обработка исключительных ситуаций.**

Время - 4 час.

Основные вопросы:

1. Понятие исключения, исключительной ситуации.
2. Обработка исключительных ситуаций try – catch. Структура оператора.
3. Использование обработки исключительных ситуаций при разработке программ.

**Тема 28. Проектирование программных комплексов.**

Время - 4 час.

Основные вопросы:

1. Этапы проектирования программных проектов. 2. Состав проекта. 3. Средства проектирования. 4. Отладка проектов.

**Тема 29. Разработка Windows-приложений.**

Время - 4 час.

Основные вопросы:

1. Понятие Windows – приложения, отличительные особенности. 2. Способы разработки Windows – приложений. 3. Разработка программ с использованием Windows – приложений. 4. Отладка программ - Windows – приложений.

**Тема 30. Графика в Windows-приложениях.**

Время - 4 час.

Основные вопросы:

1. Понятие графики в Windows – приложениях, отличительные особенности. 2. Способы разработки графики в Windows – приложениях. 3. Хранение графических образов. 4. Отладка программ - Windows – приложений.

**Тема 31. Обзор современных средств программирования.**

Время - 2 час.

Основные вопросы:

### *заочная форма обучения*

**Тема 6. Текстовый процессор MS Word. .**

Время - 1 час.

Основные вопросы:

1.Общая функциональная характеристика редактора Word. Настройка параметров. 2.Технология ввода и редактирования текста. Форматирование абзацев и списков. Создание и применение стилей. 3.Таблицы в текстовых документах. Диаграммы. Графические объекты в текстовых документах. 4.Работа с базами данных. Управление информацией в источнике данных: поиск, редактирование, сортировка. Ассистент слияния. Использование форм и полей Word.

**Тема 7. Табличный процессор MS Excel. .**

Время - 1 час.

Основные вопросы:

1.Структура окна, состав меню, панели инструментов. Настройка и создание новых панелей инструментов. Настройка параметров. 2. Автозаполнение ячеек таблицы. Абсолютные и относительные адреса, имена ячеек. 3. Ввод, редактирование и копирование формул и функций. 4. Построение диаграмм. Тип и вид диаграммы. Мастер диаграмм. 5. Работа со списками (базами данных). Сортировка, фильтрация записей. Форма. Подведение промежуточных итогов. Сводные таблицы. Сводные диаграммы.

**Тема 8. Алгоритмизация и основы программирования. .**

Время - 1 час.

Основные вопросы:

1.Разработка алгоритмов

**Тема 25. Принцип наследования.**

Время - 1 час.

Основные вопросы:

1. Понятие наследования, свойства. 2. Единичное и множественное наследование. 3. Использование механизма наследования при разработке программ

**Тема 28. Проектирование программных комплексов.**

Время - 1 час.

Основные вопросы:

1. Этапы проектирования программных проектов. 2. Состав проекта. 3. Средства проектирования. 4. Отладка проектов.

**Тема 29. Разработка Windows-приложений.**

Время - 1 час.

Основные вопросы:

1. Понятие Windows – приложения, отличительные особенности. 2. Способы разработки Windows – приложений. 3. Разработка программ с использованием Windows – приложений. 4. Отладка программ - Windows – приложений.

**Тема 30. Графика в Windows-приложениях.**

Время - 1 час.

Основные вопросы:

1. Понятие графики в Windows – приложениях, отличительные особенности. 2. Способы разработки графики в Windows – приложениях. 3. Хранение графических образов. 4. Отладка программ - Windows – приложений.

**Тема 31. Обзор современных средств программирования.**

Время - 1 час.

Основные вопросы:

## **6. ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ**

6.1.1. Основные категории учебной дисциплины для самостоятельного изучения:

**Язык программирования** - формальный язык, предназначенный для записи компьютерных программ. Язык программирования определяет набор лексических, синтаксических и семантических правил, определяющих внешний вид программы и действия, которые выполнит исполнитель (обычно - ЭВМ) под её управлением.

**Алфавит языка** - основные символы языка - буквы, цифры и специальные символы.

**Лексема** — последовательность допустимых символов языка программирования, имеющая смысл для транслятора.

**Идентификатор (ID)** – это имя программного объекта (константы, переменной, метки, типа, функции и т.д.). В идентификаторе могут использоваться латинские буквы, цифры и знак подчеркивания; первый символ ID – не цифра; пробелы внутри ID не допускаются.

**Ключевые слова** - идентификаторы, смысл которого зафиксирован правилами языка программирования и который используется для распознавания предложений в программе

**Псевдокод** - компактный (зачастую неформальный) язык описания алгоритмов, использующий ключевые слова императивных языков программирования, но опускающий

несущественные подробности и специфический синтаксис. Псевдокод обычно опускает детали, несущественные для понимания алгоритма человеком. Такими несущественными деталями могут быть описания переменных, системно-зависимый код и подпрограммы.

**Блок-схема** - распространенный тип схем (графических моделей), описывающих алгоритмы или процессы, в которых отдельные шаги изображаются в виде блоков различной формы, соединенных между собой линиями, указывающими направление последовательности.

**Трансляция программы** - преобразование программы, представленной на одном из языков программирования, в машинный код или, иногда, в язык ассемблера. Транслятор обычно выполняет также диагностику ошибок, формирует словари идентификаторов, выдаёт для печати текст программы и т. д.

**Транслятор** - программа или техническое средство, выполняющее трансляцию программы.

**Компоновщик** (также редактор связей, линкер - от англ. link editor, linker) - программа, которая производит компоновку - принимает на вход один или несколько объектных модулей и собирает по ним исполнимый модуль. Для связывания модулей, компоновщик использует таблицы имён, созданные компилятором в каждом из объектных модулей.

**Компилировать** - проводить трансляцию машинной программы с проблемно-ориентированного языка на машинно-ориентированный язык

**Сборка** (англ. build) (имя существительное) - подготовленный для использования информационный продукт. Чаще всего сборка - исполняемый файл - двоичный файл, содержащий исполняемый код (машинные инструкции) программы или библиотеки.

**Сборка** (англ. build) (глагол) - процесс получения информационного продукта из исходного кода. Чаще всего включает компиляцию и компоновку, выполняется инструментами автоматизации.

**Листинг** - исходный код (также исходный текст) – текст компьютерной программы на каком-либо языке программирования. В обобщённом смысле любые входные данные для транслятора. Исходный код либо транслируется в исполняемый код при помощи компилятора, либо исполняется непосредственно по тексту при помощи интерпретатора.

**Исполняемый код** (Машинный код, платформенно-ориентированный код), машинный язык — система команд (набор кодов операций) конкретной вычислительной машины, которая интерпретируется непосредственно процессором или микропрограммами этой вычислительной машины. Компьютерная программа, записанная на машинном языке, состоит из машинных инструкций, каждая из которых представлена в машинном коде в виде т. н. опкода — двоичного кода отдельной операции из системы команд машины. Для удобства программирования вместо числовых опкодов, которые только и понимает процессор, обычно используют их условные буквенные мнемоники. Набор таких мнемоник, вместе с некоторыми дополнительными возможностями (например, некоторыми макрокомандами, директивами), называется языком ассемблера.

**Инструкция или оператор** (англ. statement) - наименьшая автономная часть языка программирования; команда или набор команд. Программа обычно представляет собой последовательность инструкций. Многие языки (например, Си) различают инструкцию и определение. Различие в том, что инструкция исполняет код, а определение создаёт идентификатор.

**Точка входа** (англ. Entry Point (EP) - точка входа) - адрес в оперативной памяти, с которого начинается выполнение программы. Другими словами - адрес, по которому хранится первая команда программы. Однако не надо путать её с «первыми командами» программы на языке высокого уровня. Например, программа на C++ начинает выполнение с функции main(), на самом деле, программа в памяти начинается с первой исполняемой инструкции этой функции.

**Интегрированная среда разработки, ИСР** (англ. Integrated development

environment - IDE), также единая среда разработки, ЕСП - комплекс программных средств, используемый программистами для разработки программного обеспечения (ПО). Среда разработки включает в себя: текстовый редактор, компилятор и/или интерпретатор, средства автоматизации сборки, отладчик.

**Отладка** - этап разработки компьютерной программы, на котором обнаруживают, локализуют и устраняют ошибки. Чтобы понять, где возникла ошибка, приходится:

**Отладчик** - это программный модуль, который позволяет выполнить основные задачи, связанные с мониторингом процесса выполнения результирующей прикладной программы. Этот процесс называется отладкой и включает в себя следующие основные возможности: последовательное пошаговое выполнение результирующей программы на основе шагов по машинным командам или по операторам входного языка; выполнение результирующей программы до достижения ею одной из заданных точек останова (адресов останова); выполнение результирующей программы до наступления некоторых заданных условий, связанных с данными и адресами, обрабатываемыми этой программой; просмотр содержимого областей памяти, занятых командами или данными результирующей программы.

**Трассировка** - процесс пошагового выполнения программы. В режиме трассировки программист видит последовательность выполнения команд и значения переменных на данном шаге выполнения программы, что позволяет легче обнаруживать ошибки. Трассировка может быть начата и окончена в любом месте программы, выполнение программы может останавливаться на каждой команде или на точках останова, трассировка может выполняться с заходом в процедуры и без заходов, а также осуществляться в обратном порядке (шаг назад).

**Визуальное программирование** - способ создания программы для ЭВМ путём манипулирования графическими объектами вместо написания её текста. Визуальное программирование часто представляют как следующий этап развития текстовых языков программирования.

**Визуальные средства разработки** - как правило, под ними подразумевают средства проектирования интерфейсов или какую либо CASE-систему для быстрой разработки приложений.

**Компонент** - в программировании, множество классов и языковых конструкций, объединённых по общему признаку. В большинстве языков программирования нет языковых конструкций прямо отражающих понятие компонента. Компоненты реализуются с помощью стандартных конструкций, таких как классы

**UML** – унифицированный язык моделирования (Unified Modeling Language) – это система обозначений, которую можно применять для объектно-ориентированного анализа и проектирования. Его можно использовать для визуализации, спецификации, конструирования и документирования программных систем.

**STL** – (Standard Template Library) является программной библиотекой для C++ языка программирования, который влиял на многие части стандартной библиотеки C++. Он предоставляет четыре компонента, которые называются алгоритмами, контейнерами, функциями и итераторами.

**Библиотека визуальных компонентов** (англ. Visual Component Library, VCL) - объектно-ориентированная библиотека для разработки программного обеспечения, разработанная компанией Borland (на данный момент поддерживается Embarcadero) для поддержки принципов визуального программирования. VCL входит в комплект поставки Delphi, C++ Builder и Embarcadero RAD Studio и является, по сути, частью среды разработки, хотя разработка приложений в этих средах возможна и без использования VCL. VCL предоставляет огромное количество готовых к использованию компонентов для работы в самых разных областях программирования, таких, например, как интерфейс пользователя (экранные формы и элементы управления - т. н. «контролы», «контроли»), работа с базами данных, взаимодействие с операционной системой, программирование

сетевых приложений и прочее. Net.

## 6.1.2. Задания для повторения и углубления приобретаемых знаний

№	Задание	Код результата обучения
1.	1. Какие существуют способы описания алгоритмов? 2. Что такое разветвляющийся алгоритм? Приведите пример. 3. В каких случаях необходима циклическая обработка данных? Приведите пример. 4. Для чего служат директивы препроцессора?	УК-1-31
2.	5. В чем отличия между интерпретатором и компилятором? 6. Что такое язык программирования высокого (низкого) уровня. Приведите пример. 7. Что содержит заголовочный файл? 8. Поясните концепцию типов данных.	УК-1-31
3.	9. В чем отличие констант от переменных? 10. Что такое глобальная переменная? 11. В чем разница между значением и именем переменной? 12. Что такое указатель? 13. Какие операции применяются с указателями? 14. Зачем нужны операции преобразования типа? Приведите пример. 15. Перечислите операторы выбора. 16. Почему в операторе множественного выбора switch надо использовать .	УК-1-32
4.	17. Зачем в операторе множественного выбора применяется ключевое слово default? 18. К чему приведет ошибка в написании ключевого слова default? 19. Перечислите операторы цикла. 20. В чем отличие между инициализацией и присваиванием? 21. Перечислите арифметические операции. 22. Перечислите операции отношения. 23. Перечислите логические операции. 24. Что такое тернарная операция?	УК-1-32
5.	25. Поясните концепцию модульного программирования. 26. Что такое массив? Приведите пример объявления массива. 27. Как можно инициализировать массив? 28. Какие методы упорядочивания массива Вы знаете? 29. Какие методы поиска в массиве Вы знаете? 30. Что такое динамический массив? 31. Что такое многомерный массив? Приведите пример. 32. Какие поразрядные (побитовые) операции Вы знаете?	УК-1-33
6.	33. Что такое функция? 34. Что такое параметр функции? 35. В чем отличие между формальными и фактическими параметрами? 36. Что такое возвращаемое значение функции? 37. Что такое прототип функции. Когда он необходим? 38. В чем различие в передаче параметра по значению и по адресу? 39. В чем различие в передаче параметра по значению и по ссылке? 40. Допустимы ли у параметров значения по умолчанию?	УК-1-33
7.	41. Какие существуют способы копирования файлов в Windows? 42. Что такое многозадачный режим работы? 43. Можно ли восстановить удаленный в Windows файл? 44. Как в Windows определить размер свободного дискового пространства? 45. Как просмотреть содержимое буфера обмена Windows? 46. Сравните возможности Проводника и системной папки Мой компьютер. 47. Какими средствами можно осуществить настройку Word для потребностей конкретного пользователя? 48. Перечислите параметры форматирования абзацев. 49. Укажите основные характеристики шрифтов. 50. Что такое стили в редакторе Word? 51. Как выполнить копирование фрагмента документа (текста или рисунка) Word в другой документ с помощью буфера	УК-1-34



	обмена? 52. Как просуммировать данные в строках и столбцах таблицы в Word? 53. Как создать диаграмму в редакторе Word? 54. Какие объекты могут быть вставлены в документ Word?	
8.	55. Электронные таблицы Excel и их назначение. 56. Чем отличается абсолютный адрес ячейки Excel от относительного? 57. Что такое рабочий лист и рабочая книга? 58. Как пользоваться мастером функций? 59. Какие возможности предоставляет MS Excel при работе с диаграммами? 60. Что такое фильтрация данных? Что такое пользовательский фильтр? 61. Какую таблицу Excel, программа рассматривает как базу данных (список)? 62. Как подвести промежуточные итоги в списке? 63. Что такое сводная таблица, когда используется и как ее создать? 64. Что входит в состав электронного офиса? 65. Какими средствами можно организовать совместную работу в сети сотрудников офиса? 66. Какое программное и аппаратное обеспечение необходимо для работы в локальной вычислительной сети? 67. Какие функции выполняет сетевая операционная система? 68. Каковы преимущества работы в локальной сети? 69. Что такое сервер? 70. Какие виды угроз безопасности информации вы знаете? 71. В чем состоят мероприятия по обеспечению безопасности	УК-1-34
9.	73. Что такое информация и данные? 74. Какие системы счисления используются в ПЭВМ? 75. Какие единицы информации используются для измерения емкости жестких дисков? 76. Назовите основные составные части персонального компьютера. 77. От чего зависит быстродействие персонального компьютера? 78. Какие виды памяти Вы знаете? 79. Что такое внешняя память ПЭВМ? 80. Что такое программное обеспечение ПЭВМ? 81. Что входит в состав системного (базового) программного обеспечения? 82. Каковы основные функции операционной системы? 83. Что такое файловая система? 84. Какие существуют ограничения на имена файлов в DOS? 85. Для чего используется шаблоны (маски)? 86. Что такое дефрагментация файлов и каково ее назначение?	УК-1-35
10	87. В каких случаях нужна архивация данных, и какие существуют программы архивации? 88. Что такое компьютерный вирус и как он проявляется? 89. Какие мероприятия необходимы для защиты от компьютерных вирусов? 90. Как проверить дискету, диск, оперативную память на наличие вируса? 91. Назовите основные угрозы безопасности экономической информации? 92. Что такое операционные оболочки и их назначение? 93. Какие операции можно осуществлять с файлами в Norton Commander? 94. Что такое фильтрация файлов? Как она реализуется в Norton Commander? 95. Чем отличается перемещение от копирования файлов? 96. Как отыскать файл средствами Norton Commander? 97. В чем заключается процесс отладки программы?	УК-1-35
11.	98. Какие математические функции Вы знаете? 99. Какие функции ввода/вывода Вы знаете? 100. Какие функции обработки строк Вы знаете? 101. Что такое графический примитив? 102. Что такое активное меню? 103. Что такое пассивное меню?	УК-1-36
12.	104. Что такое пользовательский тип данных. Приведите пример. 105. Что такое структура в Си? Приведите пример. 106. Что означает перечисление? 107. Что такое файл прямого доступа? 108. Какие функции работы с файлами Вы знаете?	УК-1-36
13.	109. В чем состоит концепция объектно-ориентированного программирования? 110. Перечислите различия между структурой и	УК-1-37

	классом в языке C++. 111. Поясните понятие инкапсуляции. 112. Поясните понятие полиморфизма. 113. Для чего нужен конструктор? 114. Сколько конструкторов может быть определено в классе? 115. Когда используется операция присваивания и конструктор копий? 116. Что такое дружественная функция? 117. В каких случаях обязательно явно определять в классе операцию присваивания и конструктор копий? 118. В каких случаях обязательно явно определять в классе деструктор? 119. Можно ли операцию меньше определить, как метод класса? 120. Можно ли операцию меньше определить, как дружественную функцию?	
14.	121. Можно ли операцию присваивания определить, как дружественную функцию? 122. Что такое конструктор преобразования? 123. Какие члены класса наследуются? 124. Какие члены класса не наследуются? 125. Приведите пример иерархии классов. 126. Что такое множественное наследование? 127. Что такое ромбовидное наследование? 128. Что такое виртуальный метод? 129. Зачем нужен виртуальный деструктор? 130. Поясните понятие позднего связывания. 131. В чем отличие перегрузки от полиморфизма? 132. Что такое шаблон функции? Приведите пример. 133. Что такое шаблон класса? Приведите пример. 134. Что такое контейнер. Перечислите контейнеры библиотеки STL. 135. Что такое итератор? 136. Зачем нужна обработка исключений? 137. Что такое абсолютный обработчик? 138. Какие операции определены в классе string? 139. Поясните смысл операций >> и << в Си. Почему в Си++ они могут применяться в инструкциях ввода/вывода? 140. Для чего применяются манипуляторы потоков? 141. Как можно форматировать потоки?	УК-1–37
15.	142. Какие классы окон вы знаете? 143. Какие методы класса приложения Вы знаете? 144. Зачем нужны процедуры (функции) обработки событий? 145. Какие виды ресурсов Вы знаете? 146. Что такое контекст устройства? 147. Какие методы контекста устройства Вы знаете? 148. Какие параметры пера можно изменить? 149. Какие параметры кисти можно задать? 150. Что такое графический примитив?	УК-1–38
16.	151. Что такое проект? 152. Какие стандартные типы проектов Вы знаете? 153. Перечислите основные файлы проекта в изучаемой в курсе среде программирования. 154. Что такое компонент. 155. Какие компоненты Вам известны? 156. Что такое свойства компонента? 157. Что такое событие? 158. Чем отличается визуальный компонент от не визуального? 159. Какие стандартные диалоги Вам известны? 160. Что такое модальный диалог?	УК-1–38

## 6.2. Задания, направленные на формирование профессиональных умений

№	Задание	Код результата обучения
17.	1. Предложить студентам самостоятельно перевести числа из десятичной системы счисления в двоичную, восьмеричную и шестнадцатеричную системы счисления,	УК-1–У1
18.	2. Выполнить обратные преобразования. Представить число в дополнительном коде и в формате с плавающей запятой.	УК-1– У1
19.	3. Составлять простейшие программы на языке программирования Си.	УК-1– У2

20.	4. Добавить в заготовку программы необходимые инструкции. Все переменные должны быть инициализированы различными значениями. Отладить программу и запустить её на исполнение. Сравнить значения и порядок следования адресов локальных и глобальных переменных. Ответить на вопросы.	УК-1– У2
21.	5. Основные характеристики текстового процессора MS Word. Меню. Настройка меню и панели инструментов «Стандартная» и «Форматирование». Форматирование абзацев и списков. Задание – набрать пункты 1-10 плана занятий, приведенного выше.	УК-1– У3
22.	6. 1) Набрать предложенный текст в MS Word. Вставить рисунок из Clipart, задав режим обтекания текстом. Добавить еще один рисунок, поместив его за текстом 2) На новой странице поместить таблицу: N: ФИО История Математика Информатика Средняя 1. Миллер 2 4 3 3 2. Ющенко 2 3 2 2,33 Максимальная 2 4 3 Таблицу заполнить для десяти учащихся. В строке «Максимальная» и столбце «Средняя» должны быть вставлены формулы. Построить гистограмму. 3) Создать новый документ: Уважаемый ... Довожу до Вашего сведения, что студент Кошкин сдал контрольную работу по курсу «Математические методы в экономике» у студента Мышкина. Искренне Ваш, Честный студент. Осуществить слияние с базой данных: Обращение ФИО Организация Должность Город Г-н Крюковский А. С. РОСНОУ декан Москва Г-жа Шарапова Л.В. РОСНОУ зам. зав. кафедры Москва Г-жа Камышная И.Н. РОСНОУ зам. декана Москва Г-н Кофи Анан ООН Генсек Нью-Йорк Вставьте поля в текст по образцу: «Город» «Организация» Уважаем... «Должность» «ФИО» Довожу до (далее текст Вашей кляузы) Пользуясь кнопкой «Добавить поле Word», обеспечьте вывод правильного окончания слова Уважаем... в зависимости от пола (обращение). Просмотрите результат слияния.	УК-1– У3
23.	7. Задание по командам DOS должно включать в себя создание, копирование, перемещение и удаление файлов и каталогов. Просмотр содержимого файлов и директорий. Использование команд навигации по дискам и каталогам.	УК-1– У4
24.	8. Практическое занятие по изучению Norton Commander должно включать в себя создание, копирование, перемещение и удаление файлов и каталогов. Фильтрацию и сортировку отображаемой информации, а также настройку пользовательского меню и обработки расширений файлов.	УК-1– У4
25.	9. Арифметические выражения и операции. Параметры функции. 1) Написать три варианта функции, вычисляющей сумму двух чисел. Результат сложения передать через параметр – указатель, через параметр - ссылку и через возвращаемое значение. Прототипы функций имеют вид: void pSum(int a, int b, int* s); void vSum(int a, int b, int& s); int iSum(int a, int b); В головной функции модуля (main) задать значение переменных и подсчитать соответствующие суммы, используя все варианты функции Sum. Вывести результаты.	УК-1– У5
26	10. Написать программу, вычисляющую выражение: $(ax + \sin^2(b)) / (4a + be^{-a})$ . Все переменные должны быть целого типа. Обеспечить правильный результат (без потери точности). Например, при $a = 1, b = 0, x = 1$ , программа должна выводить результат 0,25. Проверить работоспособность программы. Оформить вычисление выражения в виде функции. 11. Написать пользовательские функции,	УК-1– У5

	предназначенные для ввода переменной, передаваемой через параметр, вычисления факториала ( $n! = 1*2*3*...*n$ , указание - использовать рекурсию), вычисления ряда (приближение функции $\sin(x)$ ): $f(x) = x - \frac{(x)^3}{3!} + \frac{(x)^5}{5!} - \frac{(x)^7}{7!} + \dots$ и функцию, осуществляющую вывод значений переменной $x$ , ряда $f(x)$ и $\sin(x)$ . В функции <code>main</code> помимо декларации переменных должны содержаться только вызовы перечисленных выше функций. Глобальных переменных не использовать! Варианты: приближенное вычисление $\cos(x)$ , $e^x$ , $\ln(1+x)$ и т.п.	
27.	12. Практические упражнения включают в себя предложение нарисовать блок-схему алгоритма по заданному псевдокоду, и наоборот. Например, ввести два числа и вывести сумму: Блок - схема Псевдокод 1. Обнулить переменную $c$ 2. Ввести первое число ( $a$ ) 3. Ввести второе число ( $b$ ) 4. Посчитать сумму ( $c = a + b$ ) 5. Вывести результат	УК-1– У6
28.	13. Придумайте и решите аналогичную задачу. Например поиск максимума( минимума) в одномерном массиве.	УК-1– У6
29.	14. Алгоритм с ветвлением. Функции – возвращаемое значение. 1) Написать программу, проверяющую значение числового пароля, вводимого пользователем. При правильном значении пароля программа должна выводить «hello!», в противном случае «go out!». Оформить проверку пароля в виде отдельной функции. 2) Написать программу, запрашивающую значения трех чисел – длин отрезков и проверяющую, возможно ли из этих отрезков составить треугольник. Оформить проверку в виде отдельной функции. Указание: в треугольнике сумма длин любых двух сторон больше третьей. 3) Оформить в виде отдельной функции вычисление выражения:	УК-1– У7
30.	15. Написать программу, запрашивающую значения трех чисел и выводящую $f(x)$ .	УК-1– У7
31.	16. Обработка одномерных массивов чисел. 1) Написать функции ввода/вывода числовых массивов, суммирования их элементов и поиска максимума или минимума. В функции через параметры должны передаваться массив и количество элементов, подлежащих обработке. Написать программу, использующую эти функции. 2) Написать программу, осуществляющую упорядочивание элементов массива. 3) Написать программу поиска в массиве элемента с заданным значением. Программа должна выводить индекс найденного элемента массива или $-1$ . 4) Оформить решение задач сортировки и поиска в массиве в виде функций. 5*) Написать программу, имитирующую работу с иерархическим многоуровневым меню. Выбор пунктов меню должен осуществляться курсорными клавишами, клавиша Esc – должна возвращать пользователя к предыдущему уровню меню, клавиша ввода (Enter) – переход к вложенному меню или вывод сообщения.	УК-1– У8
32.	17. Обработка двумерных массивов чисел. 1) Написать функции ввода/вывода двумерных числовых массивов. 2) Написать программу умножения матриц. 3) Написать программу транспонирования матрицы, без использования дополнительных массивов (т.е. саму в себя). 4*) Написать программу расчета определителей произвольного порядка. Указание: использовать разложение определителя по строке.	УК-1– У8

## 6.3.Задания, направленные на формирование профессиональных навыков, владений

№	Задание	Код результата обучения
33.	18. В данном задании для организации циклов использовать операторы <code>if</code> и <code>goto</code> . 1) Написать программу, выводящую на экран пассивное меню: Enter value from one to four: 1 – brown, 2 – blond, 3 – red, 4 – black. В случае если введенное пользователем значение не соответствует ни одному пункту меню, приложение должно предложить повторно выбрать допустимый пункт. 2) Добавить реакцию на выбранный пункт: если выбрана 1, вывести ALIBI; 2 – DEAD; 3 – WITNESS; 4 – WANTED; в остальных случаях – BALD 3) Добавить в программу запрос на повторный ввод данных: <code>once more? (y/n)</code> и в случае, если пользователь нажмет клавишу “y”, возврат к пункту 1) задания. 4) Оформить вывод меню и обработку выбранного значения отдельными функциями.	УК-1–В1
34.	19. Структуры, массивы структур. 1) Описать структуру «Студент (номер, имя, средний балл)». 2) Написать функции ввода/вывода массива структур «Студент». 3) Написать функцию упорядочивания массива структур «Студент» по среднему баллу. 4) Написать функцию поиска в массиве структур «Студент» нужной записи. Функция должна возвращать индекс найденной записи или –1. 5) Написать и отладить программу, использующую эти функции. 6*) Разработать структуру – фрейм меню. Написать программу, имитирующую работу с иерархическим многоуровневым меню. Выбор пунктов меню должен осуществляться курсорными клавишами, клавиша Esc – должна возвращать пользователя к предыдущему уровню меню, клавиша ввода (Enter) – переход к вложенному меню или вывод сообщения.	УК-1– В1
35.	20. В данном задании для организации циклов использовать операторы <code>while</code> и <code>do while</code> . 1) Написать программу проверки пароля. В случае если пароль не верен, приложение должно повторно запрашивать пароль. 2) Переписать программы предыдущего задания с помощью инструкций <code>while</code> и <code>do while</code> . 3) Написать программу суммирования и подсчета арифметического среднего вводимых с клавиатуры <code>n</code> чисел. Число <code>n</code> вводится с клавиатуры пользователем. 4) Написать программу выбора максимального или минимального значения из вводимых с клавиатуры <code>n</code> чисел. Число <code>n</code> вводится с клавиатуры пользователем. 5) Написать варианты программ суммирования и поиска максимального или минимального значения из вводимых с клавиатуры чисел для случая, когда их количество неизвестно ( <code>n</code> не задается), а подсчет заканчивается, если очередное введенное число оказалось равно 2006. Число 2006 не должно учитываться при вычислении суммы и среднего и поиске максимума или минимума.	УК-1– В2
36.	21. Написать пользовательские функции работы со строками. Прототипы функций: <code>int strlen(char* s);</code> // длина строки <code>char* strcpy(char* out,char* in);</code> // копирование <code>char* strcat(char* out,char* in);</code> // конкатенация <code>char* strchr(char* s,char c);</code> // поиск символа в строке <code>char* strstr(char* s, char* str);</code> // поиск вхождения В головной функции продемонстрировать работоспособность Ваших функций – задать строку, скопировать её, дописать текст в конец строки и вывести результаты. 22. Стандартные функции обработки строк. Массивы строк. 1) Заменить в программе, созданной на предыдущем задании пользовательские функции на библиотечные ( <code>string.h</code> ). Проверить	УК-1– В2

	<p>работу программы. 2) Написать функцию проверки пароля, состоящего из произвольных символов. 3) Написать программу подсчета количества заданного пользователем символа или вхождения в строке (strchr или strstr). 4) Написать программу подсчета количества слов в строке (strtok). 5) Написать функции упорядочивания массива строк и поиска строки в массиве. Функция поиска должна выводить индекс найденной в массиве строки или -1. 6*) Пользователь вводит строку, содержащую арифметическое выражение. Например, <math>12+3*4-(2+8)/2</math>. Проверить корректность выражения. Посчитать результат.</p>																																																													
37.	<p>23. Excel - 1. 1) На первом листе создать таблицу: номер название книги тематический раздел цена в у.е. куплено экземпляров стоимость в у.е. стоимость в рублях</p> <table border="1"> <tr> <td>1</td> <td>Язык C/C++ программирование</td> <td>10</td> <td>13</td> <td>104</td> </tr> <tr> <td>2</td> <td>Конан фэнтези</td> <td>4</td> <td>120</td> <td>336</td> </tr> <tr> <td>3</td> <td>Сварог фэнтези</td> <td>5</td> <td>150</td> <td>525</td> </tr> <tr> <td>4</td> <td>Математика учебная</td> <td>3</td> <td>1</td> <td>3</td> </tr> <tr> <td>5</td> <td>VBasic программирование</td> <td>8</td> <td>11</td> <td>79,2</td> </tr> <tr> <td>6</td> <td>Экономика учебная</td> <td>2</td> <td>0</td> <td>0</td> </tr> </table> <p>Стоимость в у.е. должна рассчитываться по формуле с учетом скидки – при стоимости заказа свыше 300 у.е. – 30%, свыше 100 у.е. – 20%, свыше 50 у.е. – 10%. Текущее значение курса валюты поместить на второй лист, присвоить соответствующей ячейке имя. В формуле расчета стоимости заказа в рублях применять имя ячейки с текущим курсом валюты. 2) Подсчитать среднюю цену книги, максимальное количество заказанных экземпляров и общую стоимость используя соответствующие функции. 3) Добавить диаграмму, построенную по таблице.</p>	1	Язык C/C++ программирование	10	13	104	2	Конан фэнтези	4	120	336	3	Сварог фэнтези	5	150	525	4	Математика учебная	3	1	3	5	VBasic программирование	8	11	79,2	6	Экономика учебная	2	0	0	УК-1– В3																														
1	Язык C/C++ программирование	10	13	104																																																										
2	Конан фэнтези	4	120	336																																																										
3	Сварог фэнтези	5	150	525																																																										
4	Математика учебная	3	1	3																																																										
5	VBasic программирование	8	11	79,2																																																										
6	Экономика учебная	2	0	0																																																										
38.	<p>24. Excel - 2. 1) На первом листе создать таблицу: номер фамилия должность стаж цвет волос пол оклад</p> <table border="1"> <tr> <td>3</td> <td>Мышкин инженер</td> <td>2</td> <td>белый</td> <td>муж</td> <td>110</td> </tr> <tr> <td>9</td> <td>Воробьева инженер</td> <td>10</td> <td>белый</td> <td>жен</td> <td>100</td> </tr> <tr> <td>6</td> <td>Львова программист</td> <td>2</td> <td>белый</td> <td>жен</td> <td>300</td> </tr> <tr> <td>5</td> <td>Мышкина админ</td> <td>8</td> <td>рыжий</td> <td>жен</td> <td>100</td> </tr> <tr> <td>8</td> <td>Кошкина лаборант</td> <td>2</td> <td>рыжий</td> <td>жен</td> <td>50</td> </tr> <tr> <td>1</td> <td>Зайцев программист</td> <td>10</td> <td>рыжий</td> <td>муж</td> <td>550</td> </tr> <tr> <td>4</td> <td>Львов программист</td> <td>4</td> <td>рыжий</td> <td>муж</td> <td>400</td> </tr> <tr> <td>2</td> <td>Кошкин админ</td> <td>5</td> <td>черный</td> <td>муж</td> <td>120</td> </tr> <tr> <td>10</td> <td>Орлов админ</td> <td>1</td> <td>черный</td> <td>муж</td> <td>110</td> </tr> <tr> <td>7</td> <td>Синицын лаборант</td> <td>3</td> <td>черный</td> <td>муж</td> <td>80</td> </tr> </table> <p>2) Пользуясь пунктом меню «Данные» добавить в список несколько записей. Упорядочить записи по различным полям. Применяя фильтр оставить записи только по программистам. Отобразить всех, с окладом от 300 до 600 у. е. Применяя пункт меню «Итоги» посчитать количество мужчин и женщин, затем средний оклад по каждой должности. 3) Создать на основе базы данных сводную таблицу подсчитать суммарный оклад для всех должностей (строки), с разбивкой по половому признаку (столбцы). Предусмотреть возможность анализа по цвету волос (страницы). На отдельном листе создать диаграмму по сводной таблице.</p>	3	Мышкин инженер	2	белый	муж	110	9	Воробьева инженер	10	белый	жен	100	6	Львова программист	2	белый	жен	300	5	Мышкина админ	8	рыжий	жен	100	8	Кошкина лаборант	2	рыжий	жен	50	1	Зайцев программист	10	рыжий	муж	550	4	Львов программист	4	рыжий	муж	400	2	Кошкин админ	5	черный	муж	120	10	Орлов админ	1	черный	муж	110	7	Синицын лаборант	3	черный	муж	80	УК-1– В3
3	Мышкин инженер	2	белый	муж	110																																																									
9	Воробьева инженер	10	белый	жен	100																																																									
6	Львова программист	2	белый	жен	300																																																									
5	Мышкина админ	8	рыжий	жен	100																																																									
8	Кошкина лаборант	2	рыжий	жен	50																																																									
1	Зайцев программист	10	рыжий	муж	550																																																									
4	Львов программист	4	рыжий	муж	400																																																									
2	Кошкин админ	5	черный	муж	120																																																									
10	Орлов админ	1	черный	муж	110																																																									
7	Синицын лаборант	3	черный	муж	80																																																									
39.	<p>25. Задание должно включать в себя изучение систем навигации – «Проводник» и «Мой компьютер», работу с файлами, ярлыками.</p>	УК-1– В4																																																												
40.	<p>26. Рассмотреть процедуры настройки «Панели задач» и меню кнопки «Пуск», задачи поиска файлов и папок и стандартные программы, входящие в состав Windows.</p>	УК-1– В4																																																												
41.	<p>27. В данном задании для организации циклов использовать оператор for. 1) Переписать программы предыдущего задания (задание 14, пункты с 3 по 5) с помощью оператора for. 2*) Используя только целые типы данных и арифметические операции написать программу, запрашивающую от пользователя целое число и выводящую целое число, записанное в обратном порядке цифр. Например, ввели 134,</p>	УК-1– В5																																																												

	<p>получили 431. 3*) Используя только целые типы данных и арифметические операции написать программу, находящую наибольший общий делитель двух целых чисел (применить алгоритм Евклида). 4*) Используя только целые типы данных и арифметические операции написать программу перевода десятичного целого в двоичное представление.</p>	
42	<p>28. Функция <math>\sin(x)</math> может быть представлена в виде ряда: 1) Написать пользовательскую функцию, вычисляющую значение этого ряда в точке <math>x</math> с заданной точностью <math>\varepsilon</math> (то есть последний член ряда). 2) Написать ПФ, вычисляющую сумму первых <math>n</math> членов этого ряда в точке <math>x</math>. 3) Написать функцию, выводящую результаты вычислений в виде таблицы в диапазоне от <math>x_0</math> до <math>x_k</math> с шагом <math>\Delta x</math>: <math>x</math> <math>f(x)</math> <math>\sin(x)</math> 0.00 0.0000000 0.0000000 0.10 0.0998334 0.0998334 ..... В последнем столбце должно приводиться значение, рассчитанное с помощью стандартной библиотечной функции <math>\sin(x)</math>. 4) Написать и отладить программу, запрашивающую диапазон, шаг и желаемую точность вычислений (или количество членов ряда) и выдающую результат в виде таблицы. Ввод всех параметров оформить в виде отдельной функции. Глобальных переменных не использовать! Варианты: приближенное вычисление <math>\cos(x)</math>, <math>e^x</math>, <math>\ln(1+x)</math> и т. п</p>	УК-1– В5
43.	<p>29. При нажатии клавиш клавиатуры функция <code>getch()</code> возвращает целое число код клавиши. Для алфавитно-цифровых клавиш это число не нулевое, однако для других клавиш, например, функциональных или клавиш управления курсором результатом будет ноль. Для определения (расширенного) кода таких клавиш необходим повторный вызов функции <code>getch()</code>. Конструкция <code>int c; ... while((c=getch())!=0);</code> позволит получить не нулевой код независимо от типа клавиши. 1) Написать программу, выводящую в цикле символьное значение нажатой клавиши и ее код. Окончание программы – нажатие клавиши Esc (код – 27). Найти и законспектировать коды клавиш управления курсором, пробела, табуляции и клавиши ввода (Enter). 2) Используя операторы цикла и <code>switch</code>, написать программу, обрабатывающую выбор пользователем клавиш клавиатуры и сообщающую, какая клавиша была нажата: «Вы нажали ...» для клавиш, коды которых найдены в предыдущем пункте задания. Для остальных клавиш должно выдаваться сообщение «Смотри что жмешь!!!». Выход из программы – Esc. Все сообщения должны быть на русском языке. Указание: для перекодировки кириллицы из ANSI (Windows) в ASCII (DOS) можно использовать функцию <code>void CharToOem(char* источник_ANSI, char* результат_ASCII)</code>, определенную в заголовочном файле <code>windows.h</code> или утилиту <code>Fconvert</code> в Borland C++. 3) Написать программу, выводящую в левый верхний угол окна активное вертикальное меню. Выбор пунктов меню должен осуществляться курсорными клавишами <math>\uparrow</math> и <math>\downarrow</math>. Нажатие остальных клавиш должно приводить к выдаче в правом углу сообщений, аналогично предыдущему пункту задания. Выбранный пункт меню выделить инверсными цветами. Выход из программы – Esc. Все сообщения и пункты меню должны быть на русском языке. Указания: Использовать функции <code>clrscr</code>, <code>clreol</code>, <code>textcolor</code>, <code>textbackground</code>, <code>gotoxy</code>, <code>cputs</code> и т.п., определенные в <code>conio.h</code>. Алгоритм может иметь следующий вид: 1. перекодировка кириллицы 2. начало внешнего цикла 3. задание цветов меню 4. начало внутреннего цикла – вывод меню 5. выбор строки окна 6. вывод очередного пункта меню 7. если</p>	УК-1– В6

	<p>меню нарисовано не полностью – переход к п.4 8. выбор инверсных цветов 9. выбор строки окна 10. вывод выбранного пункта меню инверсными цветами 11. считывание кода нажатой клавиши 12. действия, в зависимости от кода клавиши – switch 13. если не нажата клавиша Esc – возврат к п.2 14. выход 4*) Написать программу, выводящую в первой строке окна активное горизонтальное меню. Выбор пунктов меню должен осуществляться курсорными клавишами ← и →.</p>	
44.	<p>30. Используя функции, написанные на предыдущем занятии и алгоритмы работы с активным меню (рассмотренные в задании 16), разработать базу данных «Студент». Программная реализация должна предусматривать решение задач ввода и удаления данных, просмотр, сортировку и поиск. Активное меню должно иметь вид: "Открыть " "Сохранить " "Добавить " "Удалить " "Просмотр " "Сортировка" "Найти " "Выйти " Функции открытия и сохранения базы данных будут рассмотрены позднее. Выход из программы продублировать клавишей Esc. Нажатие непредусмотренных клавиш должно приводить к звуковому сигналу - puts("\a"). Указание: Просмотр записей и удаление выбранной записи можно осуществлять клавишами ← , → и Del, которые должны становиться доступными в режимах поиска и удаления. Соответствующие этим режимам функции желательно подробно разобрать с преподавателем.</p>	УК-1– В6
45.	<p>31. Доработка базы данных «Студент». 1) Написать функции записи в текстовый файл и чтения из файла массива структур «Студент». 2) Написать функции записи в бинарный файл и чтения из файла массива структур «Студент». 3) Написать функцию чтения из файла произвольного доступа записи «Студент» с переданным через параметр номером. 4) Доработать базу данных «Студент», дополнив её функциями открытия и сохранения.</p>	УК-1– В7
46.	<p>32. Рекурсия, параметры по умолчанию, перегрузка, шаблон функции. Во всех примерах задания необходимо написать программу, иллюстрирующую работу функций. 1) Не используя циклы написать функцию, вычисляющую сумму натуральных чисел <math>1+2+\dots+n</math>. Число <math>n</math> является параметром функции. Указание: применить рекурсию. 2) Написать функцию, вычисляющую сумму <math>n</math> элементов массива начиная с элемента <math>n_0</math>. Переменные <math>n</math> и <math>n_0</math> – параметры функции, причем параметр <math>n_0</math> должен иметь значение по умолчанию, равное 0. 3) Написать несколько перегруженных функций, выдающих сообщение о типе параметра функции. Например, <code>void message (char* str) { printf("Это строка - %s",str); }</code> 4) Написать шаблон функции, возвращающей сумму значений двух параметров функции. В случае если параметрами являются строки, результатом считать конкатенацию. 33. Графика в консольном приложении (graphics.h). 1) Используя графические примитивы, нарисовать картинку, сопроводив её подписью. Например, 2) Создать консольное приложение, иллюстрирующее движение нескольких шариков, отражающихся от стенок (границ окна). Указание: При абсолютно упругом отражении от вертикальной стенки значение горизонтальной компоненты скорости меняет знак. При отражении от горизонтальных границ меняет знак вертикальная компонента скорости. 3*) Создать консольное приложение, позволяющее управлять с помощью курсорных клавиш движением шарика, отражающегося от границ окна. Указание: В цикле,</p>	УК-1– В7



	<p>организуящем перерисовку шарика использовать временную задержку delay(dos.h параметр – время в мсек). Опрос клавиатуры производится, если функция kbhit() возвращает значение «истина».</p>	
47.	<p>34. Описать структуру Студент, с полями: имя, оценка, номер. Объявить массив структур – Группа. Прочитать данные из созданного в блокноте текстового файла. Упорядочить данные. Записать в бинарный файл. 35. Написать программу, считывающую информацию о студенте из созданного на предыдущем занятии бинарного файла по номеру записи. Ввести имя студента, найти нужную запись в файле и вывести всю информацию (имя, оценка, номер) об искомом студенте. 36. Описать класс Студент, с полями: имя, оценка, номер. Поле оценка должно быть скрытым. В классе определить несколько конструкторов, методы доступа к скрытому полю, с проверкой корректности оценки, и методы ввода/вывода на консоль. Объявить массив структур – Группа. Ввести данные с клавиатуры и записать в файл. 37. В определение класса Студент добавить конструктор копий, операцию присваивания и деструктор. Объявить массивы структур – «Первый курс» и «Второй курс». Прочитать данные из файла, созданного на предыдущем занятии в таблицу «Первый курс», скопировать во вторую таблицу и вывести данные из таблицы «Второй курс». 38. В классе Студент перегрузить операции «меньше», «равно», помещения в потоки и извлечение из потока (&lt;&lt; и &gt;&gt;). Объявить массив структур – Группа. Заполнить таблицу Группа. Упорядочить данные. Ввести имя студента, найти нужную запись в файле и вывести всю информацию (имя, оценка, номер) об искомом студенте. 39. Описать класс комплексных чисел. В классе определить арифметические операции.</p>	УК-1– В8
48.	<p>40. Описать структуру Студент, с полями: имя, оценка, номер. Поле имя должно быть указателем. В классе определить несколько конструкторов, методы ввода/вывода на консоль, конструктор копий, операция присваивания и деструктор. Объявить массивы структур – «Первый курс» и «Второй курс». Заполнить таблицу «Первый курс», скопировать во вторую таблицу и вывести данные из таблицы «Второй курс». 41. Описать структуру Студент, с полями: имя, оценка, номер. Поле имя должно быть объектом стандартного класса string. В классе определить несколько конструкторов, методы ввода/вывода на консоль и операции «меньше», «равно». Объявить массив структур – Группа. Заполнить таблицу Группа. 42. Описать класс Аспирант, являющийся наследником класса Студент. В наследнике добавит поле – шеф и перегрузить операции ввода вывода родительского класса. 43. Написать шаблоны функций поиска в массиве максимума, искомого значения и упорядочивания массива. 44. Написать программу, порождающую исключения различных типов при вычислениях частного двух вводимых с клавиатуры чисел, корня из числа и попытке открытия на чтение не существующего файла. Описать обработчики исключений соответствующих типов. Добавить абсолютный обработчик. 45. Создать приложение Windows типа “Hello World” 46. Создать приложение Windows – калькулятор. 47. Создать приложение Windows – текстовый редактор.</p>	УК-1– В8

**7. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ  
ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ  
ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**

## 7.1. Средства оценивания в ходе текущего контроля

- письменные краткие опросы в ходе аудиторных занятий на знание категорий учебной дисциплины;
- задания и упражнения, рекомендованные для самостоятельной работы;
- практические работы;
- ответы на вопросы при подготовке к экзамену.

## 7.2. ФОС для текущего контроля

<b>№</b>	<b>Код результата обучения</b>	<b>ФОС текущего контроля</b>
1.	УК-1-31	вопросы 001 – 004
2.	УК-1-31	вопросы 005 – 008
3.	УК-1-32	вопросы 009 – 016
4.	УК-1-32	вопросы 017 – 024
5.	УК-1-33	вопросы 025 – 032
6.	УК-1-33	вопросы 033 – 040
7.	УК-1-34	вопросы 041 – 060
8.	УК-1-34	вопросы 061 – 072
9.	УК-1-35	вопросы 073 – 080
10.	УК-1-35	вопросы 081 – 097
11.	УК-1-36	вопросы 098– 103
12.	УК-1-36	вопросы 104 – 108
13.	УК-1-37	вопросы 109 – 130
14.	УК-1-37	вопросы 131 – 141
15.	УК-1-38	вопросы 142 – 150
16.	УК-1-38	вопросы 151 – 160
17.	УК-1-У1	Задание 01
18.	УК-1-У1	Задание 02
19.	УК-1-У2	Задание 03
20.	УК-1-У2	Задание 04
21.	УК-1-У3	Задание 05
22.	УК-1-У3	Задание 06
23.	УК-1-У4	Задание 07
24.	УК-1-У4	Задание 08
25.	УК-1-У5	Задание 09
26.	УК-1-У5	Задание 10
27.	УК-1-У6	Задание 11
28.	УК-1-У6	Задание 12-13
29.	УК-1-У7	Задание 14
30.	УК-1-У7	Задание 15
31.	УК-1-У8	Задание 16
32.	УК-1-У8	Задание 17
33.	УК-1-В1	Задание 18
34.	УК-1-В1	Задание 19
35.	УК-1-В2	Задание 20
36.	УК-1-В2	Задание 21-22
37.	УК-1-В3	Задание 23
38.	УК-1-В3	Задание 24
39.	УК-1-В4	Задание 25
40.	УК-1-В4	Задание 26
41.	УК-1-В5	Задание 26

42.	УК-1-В5	Задание 27-28
43.	УК-1-В6	Задание 29
44.	УК-1-В6	Задание 30
45.	УК-1-В7	Задание 31
46.	УК-1-В7	Задание 32-33
47.	УК-1-В8	Задание 34-40
48.	УК-1-В8	Задание 41-47

## 7.3. ФОС для промежуточной аттестации

**Задания для оценки знаний**

№	Код результата обучения	Задания
1.	УК-1-31	1. Что такое информация и данные? 2. Какие системы счисления используются в ПЭВМ?
2.	УК-1-31	3. Какие единицы информации используются для измерения емкости жестких дисков? 4. Назовите основные составные части персонального компьютера.
3.	УК-1-32	5. От чего зависит быстродействие персонального компьютера? 6. Какие виды памяти Вы знаете?
4.	УК-1-32	7. Что такое внешняя память ПЭВМ? 8. Что такое программное обеспечение ПЭВМ?
5.	УК-1-33	9. Что входит в состав системного (базового) программного обеспечения? 10. Каковы основные функции операционной системы?
6.	УК-1-33	11. Что такое файловая система? 12. Какие существуют ограничения на имена файлов в DOS?
7.	УК-1-34	13. Для чего используются шаблоны (маски)? 14. Что такое дефрагментация файлов и каково ее назначение?
8.	УК-1-34	15. В каких случаях нужна архивация данных, и какие существуют программы архивации? 16. Что такое компьютерный вирус и как он проявляется?
9.	УК-1-35	вопросы к экзамену 17-20 (1)
10.	УК-1-35	17. Какие мероприятия необходимы для защиты от компьютерных вирусов? 18. Как проверить дискету, диск, оперативную память на наличие вируса? 19. Назовите основные угрозы безопасности экономической информации? 20. Что такое операционные оболочки и их назначение?
11.	УК-1-36	вопросы к экзамену 21-24 (1)
12.	УК-1-36	21. Какие операции можно осуществлять с файлами в Norton Commander? 22. Что такое фильтрация файлов? Как она реализуется в Norton Commander? 23. Чем отличается перемещение от копирования файлов? 24. Как отыскать файл средствами Norton Commander?
13.	УК-1-37	вопросы к экзамену 25-28 (1)
14.	УК-1-37	25. В чем заключается процесс отладки программы? 26. Какие существуют способы копирования файлов в Windows? 27. Что такое многозадачный режим работы? 28. Можно ли восстановить удаленный в

		Windows файл?
15.	УК-1-38	вопросы к экзамену 29-32 (1)
16.	УК-1-38	29. Как в Windows определить размер свободного дискового пространства? 30. Как просмотреть содержимое буфера обмена Windows? 31. Функциональные возможности Проводника и системной папки Мой компьютер. 32. Как можно осуществить настройку Word для потребностей конкретного пользователя?

### Задания для оценки умений

№	Код результата обучения	Задания
1.	УК-1-У1	В качестве фондов оценочных средств для оценки умений обучающегося используются задания 1, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.2).
2.	УК-1-У1	В качестве фондов оценочных средств для оценки умений обучающегося используются задания 2, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.2).
3.	УК-1-У2	В качестве фондов оценочных средств для оценки умений обучающегося используются задания 3, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.2).
4.	УК-1-У2	В качестве фондов оценочных средств для оценки умений обучающегося используются задания 4, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.2).
5.	УК-1-У3	В качестве фондов оценочных средств для оценки умений обучающегося используются задания 5, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.2).
6.	УК-1-У3	В качестве фондов оценочных средств для оценки умений обучающегося используются задания 6, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.2).
7.	УК-1-У4	В качестве фондов оценочных средств для оценки умений обучающегося используются задания 7, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.2).
8.	УК-1-У4	В качестве фондов оценочных средств для оценки умений обучающегося используются задания 8, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.2).
9.	УК-1-У5	В качестве фондов оценочных средств для оценки умений обучающегося используются задания 9, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.2).
10.	УК-1-У5	В качестве фондов оценочных средств для оценки умений обучающегося используются задания 10, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.2).
11.	УК-1-У6	В качестве фондов оценочных средств для оценки

		умений обучающегося используются задания 11, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.2).
12.	УК-1-У6	В качестве фондов оценочных средств для оценки умений обучающегося используются задания 12, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.2).
13.	УК-1-У7	В качестве фондов оценочных средств для оценки умений обучающегося используются задания 13, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.2).
14.	УК-1-У7	В качестве фондов оценочных средств для оценки умений обучающегося используются задания 14, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.2).
15.	УК-1-У8	В качестве фондов оценочных средств для оценки умений обучающегося используются задания 15, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.2).
16.	УК-1-У8	В качестве фондов оценочных средств для оценки умений обучающегося используются задания 16-17, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.2).

**Задания, направленные на формирование профессиональных навыков, владений.**

№	Код результата обучения	Задания
1.	УК-1-В1	В качестве фондов оценочных средств для оценки навыков, владений, опыта, деятельности обучающегося используются задания 18-19, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.3), а также практическая работа: чтение лекций, проведение различных видов практических занятий с использованием активных методов обучения.
2.	УК-1-В1	В качестве фондов оценочных средств для оценки навыков, владений, опыта, деятельности обучающегося используются задания 20-21, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.3), а также практическая работа: чтение лекций, проведение различных видов практических занятий с использованием активных методов обучения.
3.	УК-1-В2	В качестве фондов оценочных средств для оценки навыков, владений, опыта, деятельности обучающегося используются задания 22-23, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.3), а также практическая работа: чтение лекций, проведение различных видов практических занятий с использованием активных методов обучения.
4.	УК-1-В2	В качестве фондов оценочных средств для оценки навыков, владений, опыта, деятельности

		обучающегося используются задания 24-25, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.3), а также практическая работа: чтение лекций, проведение различных видов практических занятий с использованием активных методов обучения.
5.	УК-1-В3	В качестве фондов оценочных средств для оценки навыков, владений, опыта, деятельности обучающегося используются задания 26-27, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.3), а также практическая работа: чтение лекций, проведение различных видов практических занятий с использованием активных методов обучения.
6.	УК-1-В3	В качестве фондов оценочных средств для оценки навыков, владений, опыта, деятельности обучающегося используются задания 28-29, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.3), а также практическая работа: чтение лекций, проведение различных видов практических занятий с использованием активных методов обучения.
7.	УК-1-В4	В качестве фондов оценочных средств для оценки навыков, владений, опыта, деятельности обучающегося используются задания 30-31, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.3), а также практическая работа: чтение лекций, проведение различных видов практических занятий с использованием активных методов обучения.
8.	УК-1-В4	В качестве фондов оценочных средств для оценки навыков, владений, опыта, деятельности обучающегося используются задания 32-33, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.3), а также практическая работа: чтение лекций, проведение различных видов практических занятий с использованием активных методов обучения.
9.	УК-1-В5	В качестве фондов оценочных средств для оценки навыков, владений, опыта, деятельности обучающегося используются задания 34-35, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.3), а также практическая работа: чтение лекций, проведение различных видов практических занятий с использованием активных методов обучения.
10.	УК-1-В5	В качестве фондов оценочных средств для оценки навыков, владений, опыта, деятельности обучающегося используются задания 36-37, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.3), а также практическая работа: чтение лекций, проведение различных видов практических занятий с использованием активных методов обучения.
11.	УК-1-В6	В качестве фондов оценочных средств для оценки навыков, владений, опыта, деятельности

		обучающегося используются задания 38-39, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.3), а также практическая работа: чтение лекций, проведение различных видов практических занятий с использованием активных методов обучения.
12.	УК-1-В6	В качестве фондов оценочных средств для оценки навыков, владений, опыта, деятельности обучающегося используются задания 40-41, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.3), а также практическая работа: чтение лекций, проведение различных видов практических занятий с использованием активных методов обучения.
13.	УК-1-В7	В качестве фондов оценочных средств для оценки навыков, владений, опыта, деятельности обучающегося используются задания 42-43, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.3), а также практическая работа: чтение лекций, проведение различных видов практических занятий с использованием активных методов обучения.
14.	УК-1-В7	В качестве фондов оценочных средств для оценки навыков, владений, опыта, деятельности обучающегося используются задания 44-45, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.3), а также практическая работа: чтение лекций, проведение различных видов практических занятий с использованием активных методов обучения.
15.	УК-1-В8	В качестве фондов оценочных средств для оценки навыков, владений, опыта, деятельности обучающегося используются задания 46, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.3), а также практическая работа: чтение лекций, проведение различных видов практических занятий с использованием активных методов обучения.
16.	УК-1-В8	В качестве фондов оценочных средств для оценки навыков, владений, опыта, деятельности обучающегося используются задания 47, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.3), а также практическая работа: чтение лекций, проведение различных видов практических занятий с использованием активных методов обучения.

## 8. ПЕРЕЧЕНЬ ОСНОВНОЙ И ДОПОЛНИТЕЛЬНОЙ УЧЕБНОЙ ЛИТЕРАТУРЫ, НЕОБХОДИМОЙ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

### 8.1. Основная литература:

- 1) Зыков, С. В. Программирование : учебник и практикум для вузов / С. В. Зыков. — Москва : Издательство Юрайт, 2021. — 320 с. — (Высшее образование). — ISBN 978-5-

534-02444-9. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/469579>

2) Жилко, Е. П. Информатика и программирование. Часть 1 : учебное пособие / Е. П. Жилко, Л. Н. Титова, Э. И. Дямина. — Москва : Ай Пи Ар Медиа, 2020. — 195 с. — ISBN 978-5-4497-0567-9 (ч. 1), 978-5-4497-0566-2. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/95153.html>

### **8.2. Дополнительная литература:**

1) Иноземцева С.А. Информатика и программирование [Электронный ресурс] : лабораторный практикум / С.А. Иноземцева. — Электрон.текстовые данные. — Саратов: Вузовское образование, 2018. — 68 с. — 978-5-4487-0260-0. — Режим доступа: <http://www.iprbookshop.ru/75691.html>

2) Зоткин С.П. Программирование на языке высокого уровня С/С++ [Электронный ресурс] : конспект лекций / С.П. Зоткин. — 3-е изд. — Электрон.текстовые данные. — М. : МИСИ-МГСУ, ЭБС АСВ, 2018. — 140 с. — 978-5-7264-1810-0. — Режим доступа: <http://www.iprbookshop.ru/76390.html>

3) Смоленцев Н.К. MATLAB. Программирование на Visual C#, Borland C#, JBuilder, VBA [Электронный ресурс] : учебный курс / Н.К. Смоленцев. — Электрон. текстовые данные. — Саратов: Профобразование, 2017. — 456 с. — 978-5-4488-0066-5. — Режим доступа: <http://www.iprbookshop.ru/63595.html>

## **9. ПЕРЕЧЕНЬ КОМПЛЕКТОВ ЛИЦЕНЗИОННОГО И СВОБОДНО РАСПРОСТРАНЯЕМОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ, ИСПОЛЬЗУЕМОГО ПРИ ИЗУЧЕНИИ УЧЕБНОЙ ДИСЦИПЛИНЫ**

При изучении учебной дисциплины (в том числе в интерактивной форме) предполагается применение современных информационных технологий. Комплект программного обеспечения для их использования включает в себя: операционная система Microsoft Windows 7 Pro, офисный пакет программ Microsoft Office Professional Plus 2010, офисный пакет программ Microsoft Office Professional Plus 2007, антивирусная программа Dr. Web Desktop Security Suite, архиватор 7-zip, аудиопроигрыватель AIMP, просмотр изображений FastStone Image Viewer, ПО для чтения файлов формата PDF Adobe Acrobat Reader, ПО для сканирования документов NAPS2, ПО для записи видео и проведения видеотрансляций OBS Studio, ПО для удалённого администрирования Aspiа, правовой справочник Гарант Аэро, онлайн-версия Консультант Плюс: Студент, электронно-библиотечная система IPRBooks, электронно-библиотечная система Юрайт, математические вычисления Mathcad 14 University, версия 1С для использования типовых конфигураций в учебных целях: 1С: Предприятие 8. Комплект для обучения в высших и средних учебных заведениях, моделирование бизнес-процессов CA ERwin Process Modeler 7.3, версия 1С для обучения программированию: 1С: Предприятие 8.2 Версия для обучения программированию

## **10. ПЕРЕЧЕНЬ РЕСУРСОВ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННОЙ СЕТИ «ИНТЕРНЕТ», НЕОБХОДИМЫХ ДЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)**

### **10.1. Интернет-ресурсы**

1. ЭБС IPRbooks (АйПиАрбукс) <http://www.iprbookshop.ru>
2. Образовательная платформа ЮРАЙТ <https://urait.ru>
3. <https://cyberleninka.ru> – научная электронная библиотека «КИБЕРЛЕНИНКА»
4. <https://elibrary.ru> – научная электронная библиотека
5. <http://www.gpntb.ru/> - Государственная публичная научно-техническая библиотека России



6. [http://www.math.spbu.ru/user/ant/all\\_articles/057\\_Terekhov\\_Technology\\_programming.pdf](http://www.math.spbu.ru/user/ant/all_articles/057_Terekhov_Technology_programming.pdf)- авторский учебник по программированию
7. [http://citforum.ru/programming/cpp\\_ref/index.shtml](http://citforum.ru/programming/cpp_ref/index.shtml) - справочное руководство по C++.

## **11. ОСОБЕННОСТИ ОБУЧЕНИЯ ИНВАЛИДОВ И ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ**

Изучение учебной дисциплины «Информатика и программирование» обучающимися инвалидами и лицами с ограниченными возможностями здоровья осуществляется в соответствии с Приказом Министерства образования и науки РФ от 9 ноября 2015 г. № 1309 «Об утверждении Порядка обеспечения условий доступности для инвалидов объектов и предоставляемых услуг в сфере образования, а также оказания им при этом необходимой помощи» (с изменениями и дополнениями), Методическими рекомендациями по организации образовательного процесса для обучения инвалидов и лиц с ограниченными возможностями здоровья в образовательных организациях высшего образования, в том числе оснащенности образовательного процесса, утвержденными Министерством образования и науки РФ 08.04.2014г. № АК-44/05вн, Положением об организации обучения студентов – инвалидов и лиц с ограниченными возможностями здоровья, утвержденным приказом ректора Университета от 6 ноября 2015 года №60/о, Положением о Центре инклюзивного образования и психологической помощи АНО ВО«Российский новый университет», утвержденного приказом ректора от 20 мая 2016 года №187/о.

Лица с ограниченными возможностями здоровья и инвалиды обеспечиваются электронными образовательными ресурсами, адаптированными к состоянию их здоровья.

Предоставление специальных технических средств обучения коллективного и индивидуального пользования, подбор и разработка учебных материалов для обучающихся с ограниченными возможностями здоровья производится преподавателями с учетом индивидуальных психофизиологических особенностей обучающихся и специфики приема- передачи учебной информации на основании просьбы, выраженной в письменной форме.

С обучающимися по индивидуальному плану или индивидуальному графику проводятся индивидуальные занятия и консультации.

## **12. ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**

Учебная аудитория для проведения занятий лекционного типа, занятий семинарского типа, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля успеваемости и промежуточной аттестации

Ауд.305 (компьютерный класс № 3)

Специализированная мебель:

- столы студенческие;
- стулья студенческие;
- стол для преподавателя;
- стул для преподавателя;
- столы компьютерные;
- кресла компьютерные;
- шкаф для хранения раздаточного материала;
- доска (меловая);
- маркерная доска (переносная).

Технические средства обучения:

- проектор;
- ПК для преподавателя с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду вуза;
- ПК для с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду вуза;
- веб-камера;
- экран;
- колонки;
- микрофон.

Специализированное оборудование:

- наглядные пособия (плакаты)
- информационный стенд

Составитель: С.В. Толоконников

  
(подпись)

## Аннотация рабочей программы учебной дисциплины

**ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ****Код и направление подготовки 09.03.03 Прикладная информатика****Прикладная информатика в экономике**

Настоящая программа предназначена для изучения курса «ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ» студентами, обучающимися по направлению «Прикладная информатика» (код направления 09.03.03).

Цель курса – формирование у студентов основ современной информационной культуры; выработка и закрепление навыков работы на персональном компьютере; выработка умения построения информационных моделей; развитие у студентов навыков программирования; развитие умения применять методы программирования для решения различных прикладных задач; обучение студентов алгоритмам обработки численной и текстовой информации, способам записи и реализации алгоритма на языке программирования высокого уровня.

Задачи курса – формирование у студентов основ современной информационной культуры; выработка и закрепление устойчивых навыков работы на персональном компьютере; выработка умения построения информационных моделей, анализа полученных результатов; обучение студентов основам программирования; обучение методам создания и реализации алгоритма, методам отладки и тестирования программы; формирование представлений о технологиях программирования, проектировании программных продуктов и применении их для разработки программного и информационного обеспечения.

Учебная дисциплина Информатика и программирование относится к обязательной части учебного плана и изучается на 1 курсе.

В результате освоения дисциплины обучающийся по программе бакалавриата должен овладеть:

***- Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач (УК-1)***