

Документ подписан квалифицированной электронной подписью

Сертификат: 023E519200DAAC0FA374E9329E4F1A569EE

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

Действителен до: 2022-01-01

**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 и сферы обслуживания

**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 «22» января 2019, протокол № 5/1.

Заведующий кафедрой Прикладной экономики и сферы обслуживания  
(название кафедры)

к.п.н., доцент Гнездилова Н.А.

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

Елец  
2019 год

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

Учебная дисциплина «Информатика и программирование» изучается обучающимися, осваивающими образовательную программу «Прикладная информатика» по профилю Прикладная информатика в экономике в соответствии с Федеральным государственным образовательным стандартом высшего образования по направлению 09.03.03 Прикладная информатика (уровень бакалавриата), утвержденным приказом Министерства образования и науки РФ от 19.09.2017 N 922 (ФГОС ВО 3++).

Цель 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, развитие у студентов навыков программирования, способностей к самостоятельной творческой работе, умения применять методы программирования для решения различных задач прикладных дисциплин. Обучение алгоритмам обработки числовой и текстовой информации, способам записи алгоритмов, средствам реализации алгоритма на языке программирования высокого уровня.

Изучение учебной дисциплины направлено на подготовку обучающихся к осуществлению деятельности по концептуальному, функциональному и логическому проектированию систем среднего и крупного масштаба и сложности, планированию разработки или восстановления требований к системе, анализу проблемной ситуации заинтересованных лиц, разработке бизнес-требований заинтересованных лиц, постановки целей создания системы, разработки концепции системы и технического задания на систему, организации оценки соответствия требованиям существующих систем и их аналогов, представлению концепции, технического задания на систему и изменений в них заинтересованным лицам, организации согласования требований к системе, разработке шаблонов документов требований, постановке задачи на разработку требований к подсистемам и контроль их качества, сопровождению приемочных испытаний и ввода в эксплуатацию системы, обработке запросов на изменение требований к системе, определенных профессиональным стандартом «Системный аналитик», утвержденного приказом Министерства труда и социальной защиты Российской Федерации от 28.10.2014 N 809н (Регистрационный номер №34882).

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

Учебная дисциплина «Информатика и программирование» изучается студентами, осваивающими образовательную программу бакалавров по направлению 09.03.03 «Прикладная информатика», она относится к обязательной части учебного плана, изучается по заочной форме обучения в ходе 1 сессии и 2 сессии 1 курса.

Для успешного усвоения курса студент должен знать основы информатики и математики в объеме школьного курса. Параллельно с учебной дисциплиной «Информатика и программирование» изучаются дисциплины: «Математика», «Математическая логика и дискретная математика», «Визуальное программирование».

Курс создает предпосылки для освоения таких дисциплин, как «Программная инженерия», «Проектирование информационных систем».

## **3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ), СООТНЕСЕННЫЕ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОП**

В процессе изучения учебной дисциплины студенты должны овладеть универсальной компетенцией - Способен осуществлять поиск, критический анализ и

синтез информации, применять системный подход для решения поставленных задач (УК-1).

Формируемая компетенция	Планируемые результаты обучения	Соотнесение показателей обучения дисциплины с индикаторами достижения компетенций	
		Код показателя результатов обучения	Код индикатора компетенции
<p><i>УК-1</i> Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач</p>	<b><u>Знать:</u></b>		
	основные понятия критического анализа, системного подхода для решения поставленных задач	УК-1-31	И-УК-1.1
	базовые алгоритмы обработки числовой и текстовой информации	УК-1-32	И-УК-1.1
	технологии и алгоритмы решения типовых задач	УК-1-33	И-УК-1.1
	основные понятия информатики	УК-1-34	И-УК-1.1
	назначение и основные характеристики устройств компьютера	УК-1-35	И-УК-1.1
	способы кодирования и представления информации в компьютере	УК-1-36	И-УК-1.1
	основные понятия и методы алгоритмизации процессов обработки информации	УК-1-37	И-УК-1.1
	классификацию, основные свойства и специализацию языков программирования	УК-1-38	И-УК-1.1
	<b><u>Уметь</u></b>		
	анализировать информацию и применять системный подход для решения поставленных задач	УК-1-У1	И-УК-1.2
	использовать методы по решению типовых задач	УК-1-У2	И-УК-1.2
	использовать алгоритмы обработки информации для решения поставленных задач	УК-1-У3	И-УК-1.2
	работать в качестве пользователя персонального компьютера	УК-1-У4	И-УК-1.2
	разрабатывать алгоритмы решения экономических и управленческих задач	УК-1-У5	И-УК-1.2
	использовать языки и системы программирования	УК-1-У6	И-УК-1.2
	выбрать метод решения прикладной задачи в соответствии с особенностями программного средства	УК-1-У7	И-УК-1.2
выбрать инструментальные средства обработки экономических данных в соответствии с поставленной задачей	УК-1-У8	И-УК-1.2	
<b><u>Владеть</u></b>			

	навыками для применения системных подходов	УК-1-В1	И-УК-1.3
	навыками решения практических задач	УК-1-В2	И-УК-1.3
	методологией решения научных и практических задач	УК-1-В3	И-УК-1.3
	навыками работы на ПК	УК-1-В4	И-УК-1.3
	навыками разработки алгоритмов решения экономических и управленческих задач	УК-1-В5	И-УК-1.3
	навыками использования языков и систем программирования	УК-1-В6	И-УК-1.3
	технологией создания документации с помощью текстового процессора MicrosoftWord	УК-1-В7	И-УК-1.3
	технологией решения типовых информационных и вычислительных задач с помощью табличного процессора MicrosoftExcel	УК-1-В8	И-УК-1.3

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

**4.1.Общий объем учебной дисциплины (модуля)**

№	Форма обучения	Семестр	Общая трудоемкость		В том числе контактная работа с преподавателем					СР	контроль	
			В з. е.	В часов	Всего	Лекции	Пр	КоР	Конс			Экзамен
2	Заочная	Установочная сессия, 3курс	1	36	4	4					32	
		1 сессия, 1 курс	3	108	12	4	4	1,6	2	0,4	89,4	6,6
		2 сессия, 1 курс	4	144	12	4	4	1,6	2	0,4	125,4	6,6
	<b>ИТОГО</b>		<b>8</b>	<b>288</b>	<b>28</b>	<b>12</b>	<b>8</b>	<b>3,2</b>	<b>4</b>	<b>0,8</b>	<b>246,8</b>	<b>13,2</b>

Дисциплина предполагает изучение 2 разделов, 30 тем. Общая трудоемкость дисциплины составляет 8 зачетных единиц (288 часов). Форма промежуточной аттестации – экзамены.

**4.2. Распределение учебного времени по темам и видам учебных занятий**

**а) заочная форма обучения**

№	Наименование	Всего	Контактная работа с	СР	Контр	Формиру
---	--------------	-------	---------------------	----	-------	---------

	разделов, тем учебных занятий	о часов	преподавателем							о ль	-емые результаты обучения
			Всего	Л	Пр	Ко Р	Ко н с	Экз			
1	2	3	4	5	6	7	8	9	10	11	12
1.	Принципы представления информации в ЭВМ	8	1	1					7		УК-1-32 УК-1-32
2.	Реализация информационных процессов	8	1	1					7		УК-1-32 УК-1-33
3.	Программное обеспечение (ПО) ПЭВМ	8	1	1					7		УК-1-31 УК-1-В1
4.	Управление персональным компьютером	12	1	1					11		УК-1-32 УК-1-У4
	<b>Итого</b>	<b>36</b>	<b>4</b>	<b>4</b>					<b>32</b>		

5.	Программные средства электронного офиса	20	2	1	1				18		УК-1-31 УК-1-В3
6.	Основы работы в текстовом процессоре Word	20	2	1	1				18		УК-1-В3 УК-1-У1
7.	Основы работы в табличном процессоре Excel	20	2	1	1				18		УК-1-В2 УК-1-У1 УК-1-32
8.	Алгоритмизация и основы программирования.	23	2	1	1				21		УК-1-36 УК-1-У1
	<i>Промежуточная аттестация (экзамен)</i>	25	4			1,6	2	0,4	14,4	6,6	
	<b>Итого</b>	<b>108</b>	<b>12</b>	<b>4</b>	<b>4</b>	<b>1,6</b>	<b>2</b>	<b>0,4</b>	<b>89,4</b>	<b>6,6</b>	
9.	Языки и современные среды программирования	6	1	1					5		УК-1-31 УК-1-33 УК-1-37
10.	Язык программирования С. Линейные алгоритмы	6	1	1					5		УК-1-32 УК-1-У2

11.	Реализация разветвляющихся алгоритмов.	6	1	1					5		УК-1-33 УК-1-У2
12.	Реализация циклических алгоритмов.	6	1	1					5		УК-1-33 УК-1-В8
13.	Статические и динамические массивы.	4							4		УК-1-32 УК-1-В1
14.	Организация интерфейса пользователя.	4							4		УК-1-У1 УК-1-В2
15.	Принципы работы с файлами.	5							5		УК-1-У3 УК-1-В3
16.	Графика.	5							5		УК-1-В4 УК-1-У3
17.	Пользовательские функции.	5							5		ОК-8-32 УК-1-У3
18.	Типы данных, определяемые пользователем.	5							5		УК-1-У5 УК-1-В6
19.	Архитектура языков высокого уровня.	5							5		УК-1-У1 УК-1-В3
20.	Основные понятия объектно-ориентированного подхода к программированию.	6							6		УК-1-37 УК-1-В8
21.	Инициализация объектов. Конструкторы и деструкторы.	6							6		УК-1-У3 УК-1-31
22.	Перегрузка операций.	6							6		УК-1-В3 УК-1-В2

23.	Стандартная библиотека.	6							6		УК-1-У3 УК-1-В2
24.	Наследование, его реализация средствами языка программирования.	6	1		1				5		УК-1-В4 УК-1-36
25.	Шаблоны. Библиотека STL.	6							6		УК-1-В2 УК-1-В1
26.	Обработка исключительных ситуаций.	6							6		УК-1-У5 УК-1-У3
27.	Проектирование программных комплексов.	6	1		1				5		УК-1-В7 УК-1-У8
28.	Разработка Windows - приложений.	6	1		1				5		УК-1-В2 УК-1-В2
29.	Графика в Windows - приложениях.	6	1		1				5		УК-1-У4 УК-1-В2
30.	Обзор средств и технологий программирования	6							6		УК-1-33 УК-1-В1
	<b>Промежуточная аттестация (экзамен)</b>	25	4			1,6	2	0,4	14,4	6,6	
	<b>Итого</b>	144	12	4	4	1,6	2	0,4	125,4	6,6	
	<b>ИТОГО</b>	<b>288</b>	<b>28</b>	<b>12</b>	<b>8</b>	<b>3,2</b>	<b>4</b>	<b>0,8</b>	<b>246,8</b>	<b>13,2</b>	

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

### ТЕМА 1. Принципы представления 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.

#### Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

### 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.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**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.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**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.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

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

Понятие электронного офиса. **MicrosoftOffice**. 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 **MicrosoftOffice**.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМА6. Основы работы в текстовом процессоре 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.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМА 7. Основы работы в табличном процессоре 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.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМА 8. 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.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**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.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМА10. Язык программирования Си. Реализация линейных алгоритмов.**

Алфавит языка, лексемы. Ключевые слова и идентификаторы. Типы данных. Директивы препроцессора *include* и *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.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМА11. Реализация разветвляющихся алгоритмов.**

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

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМА12. Реализация циклических алгоритмов.**

Циклы с предусловием и постусловием. Реализация циклов с помощью операторов ветвления и передачи управления. Операторы цикла **while**, **dowhile**, **for**. Взаимное приведение циклов *for* и *while*. Оператор продолжения **continue**. Прерывание циклов. Вложенные циклы. Понятие об итерации. Рекурсивные и циклические алгоритмы.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**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.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**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.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМА15. Принципы работы с файлами.**

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

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**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.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**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.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМА19. Архитектура и возможности языков высокого уровня.**

Принципы объектно-ориентированного программирования. Функциональное программирование. Визуальное программирование. Программирование баз данных. Программирование в компьютерных сетях. Программирование мультимедиа-сред.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМА20. 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.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМА21. 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.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМА22. 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.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМА 23. Стандартная библиотека.**

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

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМА 24. Принцип наследования и его реализация средствами языка программирования.**

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.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМА 25. Шаблоны. Библиотека STL.** Перегрузка функций. Шаблоны функций. Создание шаблонов функций. Классы-шаблоны. Создание и использование шаблонов классов. Библиотека STL. Алгоритмы и контейнеры. Вектор, список, стек, очередь.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМА26. Обработка исключительных ситуаций.**

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

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМА27. Проектирование программных комплексов.**

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

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМА28. 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.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМА29. 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.

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМА30. Обзор современных средств и технологий 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).

Литература:

- а) основная: 1-3.
- б) дополнительная: 4-7.

**ТЕМЫ ПРАКТИЧЕСКИХ ЗАНЯТИЙ****Тема 1. Принципы представления информации в ЭВМ**Основные вопросы:

1. Изучение единиц измерения количества информации.
2. Представление текстовой информации. Таблица ASCII. Кодировки ANSI иUnicode.
3. Двоичная система кодирования символов.
4. Арифметика в двоичной системе кодирования символов.
5. Шестнадцатеричная система кодирования символов.
6. Арифметика в шестнадцатеричной системе.
7. Представление чисел с плавающей запятой в формате double, X<sub>16</sub>.

**Тема 2. Средства реализации информационных процессов**Основные вопросы:

1. Изучение устройства внешней памяти.
2. Накопители на жестких и гибких магнитных дисках.

3. Другие накопители и носители информации.
4. Изучение устройства видеосистемы.
5. Изучение устройств ввода-вывода информации.
6. Изучение коммуникационного оборудования(модемы, сетевыеадаптеры).

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

#### Основные вопросы:

1. Основные функции операционных систем.
2. Настройка интерфейса ОС.
3. Изучение основных элементов рабочего стола ОС.
4. Использование средств программной среды ОС.
5. Использование справочной системы.
6. Правила составления имен файлов, типы файлов, шаблоны (маски) имени файлов.
7. Приемы работы с исполняемыми файлами (программами) и файлами документов.
8. Операции с файлами и каталогами: создание, копирование, переименование, удаление.
9. Архивирование файлов. Работа с архивами.

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

#### Основные вопросы:

1. Защита от компьютерныхвирусов.
2. Антивирусные средства, их классификация ииспользование.
3. Изучение проблемы информационной безопасности информационныхсистем.
4. Угрозы безопасности информации и ихклассификация.
5. Технические, организационные и программные средства защитыинформации.

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

#### Основные вопросы:

1. Усвоение понятия электронного офиса MicrosoftOffice.
2. Состав программных средств - текстовый редактор Word, электронные таблицы Excel, система управления базами данных Access, система подготовки презентаций PowerPoint.
3. Примеры обмена данными между приложениями и ихвзаимодействия.
4. Организация совместной работы пользователей средствами MicrosoftOffice.

### **Тема 6. Основы работы в текстовом процессоре Word**

#### Основные вопросы:

1. Основные понятия и терминология работы в текстовомредакторе.
2. Структура окна, состав меню, панелиинструментов.
3. Настройка и создание новых панелей инструментов. Настройкапараметров.
4. Технология ввода и редактирования текста. Форматирование абзацев и списков.
5. Создание и применениестилей.
6. Таблицы в текстовых документах.Вычисления.
7. Диаграммы. Графические объекты в текстовыхдокументах.
8. Создание составного документа, режим главногодокумента.
9. Использование форм и полей **Word**. Автоматизация работы с помощью кодовполей.
10. Изучение параметров страницы, абзаца при форматированиидокумента.
11. Организация различныхсписков.

12. Оформление дополнительных элементов (сноски, примечания, колонтитулы, ит.д.)
13. Применение различных стилей, контроль в структуредокумента.
14. Формирование оглавления многостраничного документа

### **Тема 7. Основы работы в табличном процессоре Excel**

#### Основные вопросы:

1. Структура окна, состав меню, панели инструментов Excel.
2. Настройка и создание новых панелей инструментов в Excel. Настройка параметров.
3. Изучение файла - книга Excel, листы.
4. Понятие - ячейка, ввод, редактирование и форматы данных.
5. Автоматическое заполнение таблицы прогрессией и функцией.
6. Абсолютные и относительные адреса ячеек.
7. Построение диаграмм. Тип и вид диаграммы. Мастер диаграмм.
8. Форма. Организация полей.
9. Подведение промежуточных итогов. Сводные таблицы. Мастер сводных таблиц.
10. Вставка и применение формул и функций в табличном процессоре Excel.
11. Построение диаграмм и графиков в Excel.
12. Инструменты обработки данных (сортировка, фильтрация, импорт, экспорт).
13. Связь листов в многостраничных документах.
14. Контроль расчетов в табличном процессоре Excel.

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

#### Основные вопросы:

1. Понятие алгоритма. Виды алгоритмов (линейные, с ветвлением, циклические).
2. Представление алгоритмов в графическом виде (блок-схема) и в псевдокоде.
3. Представление алгоритмов в псевдокоде.
4. Исследование типовых алгоритмов. Решение задач по алгоритмизации.
5. Методология разработки алгоритма. Оценка эффективности алгоритма

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

#### Основные вопросы:

1. Сравнительный обзор высокоуровневых языков программирования.
2. Общие понятия высокоуровневых языков программирования – типы данных, переменные, выражения, операторы ветвления, циклы, функции.
3. Трансляция программы. Виды трансляторов. Этапы создания программы.
4. Работа в интегрированной среде программирования, инструментальной среде.
5. Создание проекта в интегрированной среде программирования.

### **Тема 10. Язык программирования С. Реализация линейных алгоритмов**

#### Основные вопросы:

1. Типы данных. Переменные и константы.
2. Оператор присваивания.
3. Арифметические операции. Математические функции.
4. Операции отношения. Логические операции.
5. Логические выражения.
6. Сокращенная запись операций.

### **Тема 11. Реализация разветвляющихся алгоритмов**

Основные вопросы:

1. Логические выражения. Ветвление.
2. Блок-схемы.
3. Оператор множественного выбора.
4. Метки и безусловный переход
5. Простые и сложные логические выражения.
6. Организация ветвления.
7. Вложенность условия.
8. Оператор множественного выбора.
9. Программирование условных операторов в задачах. **Тема 12.**

**Реализация циклических алгоритмов**Основные вопросы:

1. Циклы.
2. Циклические алгоритмы.
3. Блок-схемы циклов с постусловием и предусловием.
4. Циклы в Си/Си++: while do-while.
5. Цикл for.
6. Стандартные алгоритмы: суммирование и поиск max/min.
7. Организация различных видов циклов в языках C/C++:
8. Инкрементные циклы, особенности программирования
9. Программирование циклов в прикладных задачах.

**Тема 13. Статические и динамические структуры массивы**Основные вопросы:

1. Одномерные массивы.
2. Указатели. Операции на указателями.
3. Указатели и массивы.
4. Динамические массивы.
5. Многомерные массивы.
6. Обработка векторов и матриц.

**Тема 14. Организация интерфейса пользователя**Основные вопросы:

1. Горячие клавиши.
2. Пассивное меню.
3. Активное меню.

**Тема 15. Принципы работы с файлами**Основные вопросы:

1. Работа с файлами. Текстовые файлы.
2. Бинарные файлы.
3. Задачи на циклы, массивы и файлы.

**Тема 16. Графика**Основные вопросы:

1. Графический режим.
2. Инструменты – перо, кисть и т.д.
3. Графически примитивы.

**Тема 17. Пользовательские функции**Основные вопросы:



1. Тип функции. Возвращаемое значение.
2. Параметры функции. Передача параметров по значению.
3. Указатели как параметры функции.
4. Параметры функции. Передача параметров по ссылке.
5. Рекурсия.
6. Значения параметров по умолчанию.
7. Перегрузка.

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

#### Основные вопросы:

1. Пользовательские типы данных.
2. Определение структур и массивов структур.
3. Обработка данных структур.
4. Директива **typedef**.
5. Перечисления, объединения полей битов.

### **Тема 19. Архитектура и возможности языков высокого уровня**

#### Основные вопросы:

1. Процедурное программирование.
2. Принципы объектно-ориентированного программирования.
3. Функциональное программирование.
4. Визуальное программирование.

### **Тема 20. Объектно-ориентированный подход к программированию. Основные вопросы:**

1. Понятие класса и объекта.
2. Данные-члены (свойства) и функции-члены (методы).
3. Доступ к членам класса: открытые, закрытые и защищенные члены класса.
4. Объекты. Обращение к членам объекта.
5. Дружественные классы и функции.

### **Тема 21. Инициализация объектов. Конструкторы и деструкторы**

#### Основные вопросы:

1. Конструкторы. Свойства конструкторов.
2. Конструктор по умолчанию. Параметры конструкторов.
3. Конструктор преобразования.
4. Деструктор.

### **Тема 22. Перегрузка операций**

#### Основные вопросы:

1. Понятие перегрузки. Перегрузка функций. Перегрузка операций.
2. Перегружаемые операции. Операции, не допускающие перегрузки.
3. Перегрузка унарных операций. Перегрузка бинарных операций.
4. Перегрузка операций как методов класса и как дружественных функций.

### **Тема 23. Стандартная библиотека**

#### Основные вопросы:

1. Строки C++. Класс `string`.
2. Конструкторы, операции и методы классов строк.
3. Поточковые классы. Форматирование потоков. Методы обмена потоками.
4. Файловые потоки.

### **Тема 24. Принцип наследования и его реализация**

**средствами языка программирования**Основные вопросы:

1. Базовый и производный классы.
2. Режимы доступа к членам базового класса.
3. Конструкторы и деструкторы производных классов.
4. Полиморфизм. Виртуальные методы.
5. Множественное наследование.

**Тема 25. Шаблоны. Библиотека STL**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 26. Обработка исключительных ситуаций**Основные вопросы:

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

**Тема 27. Проектирование программных комплексов**Основные вопросы:

1. Этапы создания объектно-ориентированных программ.
2. Понятие проекта. Понятие компонента.
3. Реализация проекта. Отладка проектов.
4. Средства проектирования

**Тема 28. Разработка Windows****- приложений. Основные**вопросы:

1. Структура и выполнение Windows-программ.
2. Обработка сообщений. Управление памятью.
3. Классы приложений. Инициализация окон.
4. Типы проектов.
5. Виды компонентов.

**Тема 29. Графика в Windows - приложениях.**Основные вопросы:

1. Общие принципы создания графических программ.
2. Вывод графических образов на экран.
3. Хранение графических образов.
4. Программирование графики для Windows.
5. Графические объекты Windows.

**Тема 30. Обзор современных средств и технологий программирования. Основные вопросы:**

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

## 6. ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

### 6.1.2. Задания для приобретения, закрепления и углубления знаний

#### УК-1-31

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?
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-1-32

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 оператор **break**?
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-1-33

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 (битовые) операции Вы знаете?
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-1-34**

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 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?
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 информации?
72. Перечислите методы и средства защиты информации.

#### **УК-1-35**

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 вDOS?
85. Для чего используется шаблоны(маски)?
86. Что такое дефрагментация файлов и каково ее значение?
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 значение?
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-1-36**

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?
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-1-37**

109. В чем состоит концепция объектно-ориентированного программирования?
110. Перечислите различия между структурой и 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?
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-1-38

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?
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 свойство компонента?
157. Что такое событие?
158. Чем отличается визуальный компонент от не визуального?
159. Какие стандартные диалоги Вам известны?
160. Что такое модальный диалог?

### 6.2. Задания для повторения и углубления приобретаемых умений

#### УК-1-У1

1. Предложить студентам самостоятельно перевести числа из десятичной системы счисления в двоичную, восьмеричную и шестнадцатеричную системы счисления,
2. Выполнить обратные преобразования. Представить число в дополнительном коде и в формате с плавающей запятой.

#### УК-1-У2

3. Структура программы на языке C выглядит следующим образом:

```
#include <conio.h> #include <stdio.h>
    //добавить объявление и инициализацию
    переменных a и b. void fun(){
    //добавить объявление и инициализацию переменных a и b .
    //с помощью функции printf вывести значения локальных переменных a и b .
    //с помощью функции printf вывести значения глобальных переменных a и b .
    //вывести значения адресов локальных и глобальных переменных a и b .
}
```

```

voidmain(){
    //добавить объявление и инициализацию переменных a и b .
    //вызвать функцию fun
    // с помощью функции printf вывести значения локальных переменных a и b .
    // с помощью функции printf вывести значения глобальных переменных a и b
    .
    //вывести значения адресов локальных и глобальных
    переменных a и b . getch();
}

```

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:

- зачем нужна директива препроцессора `#include<stdio.h>`;
- почему адреса локальных и глобальных переменных сильно различаются;
- почему адреса локальных переменных расположены в обратном порядке;
- зачем при работе в среде Borland C++ 5 нужна инструкция `getch()`;

### УК-1-У3

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.

6.

1) Набрать предложенный текст в MS Word. Вставить рисунок из Clipart, задав режим обтекания текстом. Добавить еще один рисунок, поместив его затекстом.

2) На новой странице поместить таблицу:

№:	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-У4

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.
8. **Практическое занятие по изучению NortonCommander** 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-У5

#### 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);
intiSum(int a,
int b);
```

В головной функции модуля (*main*) задать значение переменных и подсчитать соответствующие суммы, используя все варианты функции *Sum*. Вывести результаты.

2) Написать программу, вычисляющую выражение:

$$\frac{ax + \sin b}{4a + be^{-a}}$$

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.

10. **Написать пользовательские функции**, предназначенные для ввода переменной, передаваемой через параметр, вычисления факториала ( $n! = 1*2*3*...*n$ , указание - использовать рекурсию), вычисления ряда

$$f(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} \dots$$

(приближение функции  $\sin(x)$ ):

и функцию, осуществляющую вывод значений переменной  $x$ , ряда  $f(x)$  и  $\sin(x)$ . В функции *main* помимо декларации переменных должны содержаться

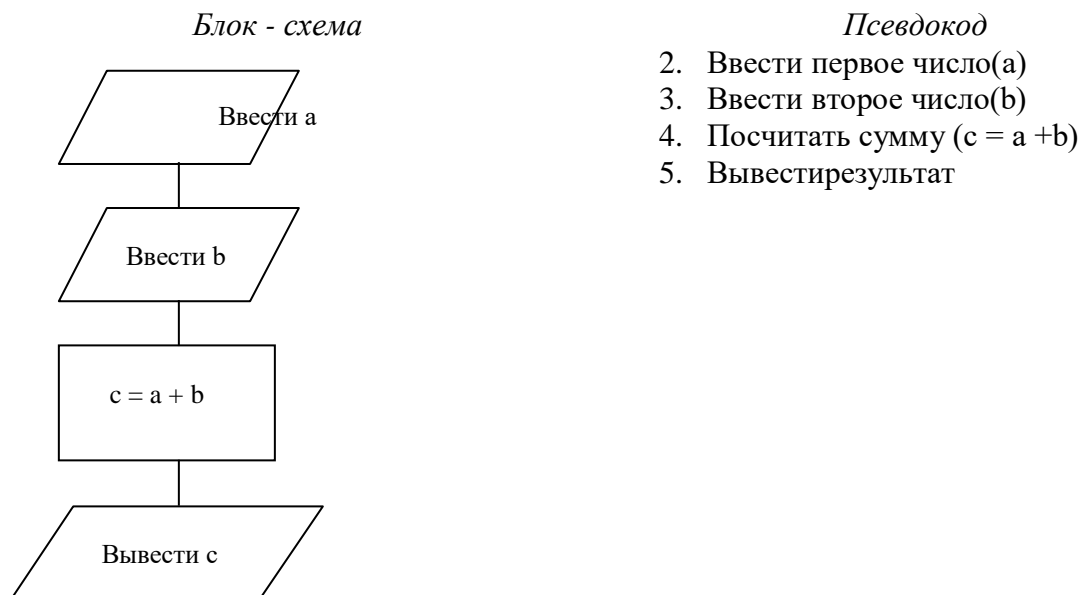


только вызовы перечисленных выше функций. Глобальных переменных не использовать!

**11. Варианты:** приближенное вычисление  $\cos(x)$ ,  $e^x$ ,  $\ln(1+x)$  ит.п.

### УК-1-У6

**12.** Практические упражнения включают в себя предложение нарисовать блок-схему алгоритма по заданному псевдокоду, и наоборот. Например, ввести два числа и вывести сумму:



**13.** Придумайте и решите аналогичную задачу

### УК-1-У7

**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 «goout!». 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.

**15. Написать программу, запрашивающую значения трех чисел и выводящую  $f(x)$ .**

### УК-1-У8

**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.

### 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 себя).
- 4\*) Написать программу расчета определителей произвольного порядка. *Указание:* использовать разложение определителя по строке.

## 6.3. Задания для повторения и углубления приобретаемых владений

### УК-1-В1

### 18. В данном задании для организации циклов использовать операторы *if* и *goto*.

- 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: *oncemore?(y/n)*

и в случае, если пользователь нажмет клавишу “y”, возврат к пункту 1) задания.

- 4) Оформить вывод меню и обработку выбранного значения отдельными функциями.

### 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-В2****20. В данном задании для организации циклов использовать операторы *while* и *dowhile*.**

- 1) Написать программу проверки пароля. В случае если пароль не верен, приложение должно повторно запрашивать пароль.
- 2) Переписать программы предыдущего задания с помощью инструкций *while* и *dowhile*.
- 3) Написать программу суммирования и подсчета арифметического среднего вводимых с клавиатуры *n* чисел. Число *n* вводится с клавиатуры пользователем.
- 4) Написать программу выбора максимального или минимального значения из вводимых с клавиатуры *n* чисел. Число *n* 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 (*n* 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.

**21. Написать пользовательские функции работы со строками.**

Прототипы функций: `int strlen(char*s);` // длина строки  
`char* strcpy(char*out, char*in);`

/ копирование `char*`  
`strcat(char*out, char*in);`

/ конкатенация  
`char* strchr(char*s, char c);` //  
поиск символа в строке `char* strstr(char*`  
`s, char*str);` //  
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 (*string.h*). Проверить 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,  $12+3*4-(2+8)/2$ . Проверить корректность выражения. Посчитать результат.

**УК-1-В3**

**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
1	Язык C/C++	программирование	10	13	104	2912
2	Конан	фэнтези	4	120	336	9408
3	Сварог	фэнтези	5	150	525	14700
4	Математика	учебная	3	1	3	84
5	VBasic	программирование	8	11	79,2	2217,6
6	Экономика	учебная	2	0	0	0

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.

**24. Excel - 2.** 1) На первом листе создать таблицу:

номер	фамилия	должность	стаж	цвет волос	пол	оклад
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

2) Пользуясь пунктом меню «Данные» добавить в список несколько записей. Упорядочить записи по различным полям. Применяя фильтр оставить записи только по программистам. Отобрать 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.

**УК-1-В4**

**25.** Задание должно включать в себя изучение систем навигации – «Проводник» и «Мой компьютер», работу с файлами, ярлыками и папками.

**26.** Рассмотреть процедуры настройки «Панели задач» и меню кнопки «Пуск», задачи поиска файлов и папок и стандартные программы, входящие в состав **Windows**.

**УК-1-В5**

**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, 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.

**УК-1-В6**

**28.** При нажатии клавиш клавиатуры функция *getch()* 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 *getch()*. Конструкция

```
int c;
... while((c=getch())!=0);
```

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 *switch*, 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

```
void CharToOem(char* источник_ANSI, char*
результат_ASCII), определенную в заголовочном файле
windows.h или утилиту Fconvert в Borland C++.
```

3) Написать программу, выводящую в левый верхний угол окна *активное* вертикальное меню. Выбор пунктов меню должен осуществляться курсорными клавишами ↑ и ↓. 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 *clrscr*, *clreol*, *textcolor*, *textbackground*, *gotoxy*, *cputs* и т.п., определенные в *conio.h*. 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 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→.

### 30. 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

$1+2+\dots+n$ . Число  $n$  является параметром функции. *Указание:* применить рекурсию.

- 2) Написать функцию, вычисляющую сумму  $n$  элементов массива начиная с элемента  $n0$ . Переменные  $n$  и  $n0$  – параметры функции, причем параметр  $n0$  должен иметь значение по умолчанию, равное 0.

- 3) Написать несколько перегруженных функций, выдающих сообщение о типе параметра функции. Например,

```
void message (char* str) { printf("Эта строка - %s",str);
}
```

- 4) Написать шаблон функции, возвращающей сумму значений двух параметров функции. В случае если параметрами являются строки, результатом считать конкатенацию.

### 31. Графика в консольном приложении (*graphics.h*).

- 1) Используя графические примитивы, нарисовать картинку, сопроводив её подписью. Например,



Bad student

- 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, организующем перерисовку шарика использовать временную задержку *delay(dos.h)* параметр – время в мсек). Опрос клавиатуры производится, если функция *kbhit()* возвращает значение «истина».

**УК-1-В8**

- 32.** 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.
- 33.** 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.
- 34.** 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.
- 35.** 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».
- 36.** В классе Студент перегрузить операции «меньше», «равно», помещения в потоки и извлечение из потока (<< и >>). 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.
- 37.** Описать класс комплексных чисел. В классе определить арифметические операции.
- 38.** 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».
- 39.** 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.
- 40.** Описать класс Аспирант, являющийся наследником класса Студент. В наследнике добавит поле – шеф и перегрузить операции ввода вывода родительского класса.
- 41.** Написать шаблоны функций поиска в массиве максимума, искомого значения и упорядочивания массива.
- 42.** 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.
- 43.** Создать приложение Windows типа “Hello World”
- 44.** Создать приложение Windows – калькулятор.
- 45.** Создать приложение Windows – текстовый редактор.

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

### 7.1. Средства оценивания текущего контроля

- письменные краткие опросы в ходе аудиторных занятий по учебной дисциплине;
- выполнение заданий и упражнений в ходе практических занятий;
- ответы на вопросы при проведении зачета и экзамена.

### 7.2. ФОС для текущего контроля

№	Формируемая компетенция	Показатели сформированности компетенции	ФОС текущего контроля
1.	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач (УК-1)	УК-1-31	раздел 6.1, вопросы 01 – 08
2.		УК-1-32	раздел 6.1, вопросы 9 – 24
3.		УК-1-33	раздел 6.1, вопросы 25 – 40
4.		УК-1-34	раздел 6.1, вопросы 41 – 72
5.		УК-1-35	раздел 6.1, вопросы 73 – 97
6.		УК-1-36	раздел 6.1, вопросы 98 – 108
7.		УК-1-37	раздел 6.1, вопросы 109 – 141
8.		УК-1-38	раздел 6.1, вопросы 142 – 160
9.		УК-1-У1	раздел 6.2, задания 01 – 02
10.		УК-1-У2	раздел 6.2, задания 03 – 04
11.		УК-1-У3	раздел 6.2, задания 05 – 06
12.		УК-1-У4	раздел 6.2, задания 07 – 08
13.		УК-1-У5	раздел 6.2, задания 09 – 10
14.		УК-1-У6	раздел 6.2, задания 11 – 13
15.		УК-1-У7	раздел 6.2, задания 14 – 15
16.		УК-1-У8	раздел 6.2, задания 16 – 17
17.		УК-1-В1	раздел 6.3, задания 18 – 19
18.		УК-1-В2	раздел 6.3, задания 20 – 22
19.		УК-1-В3	раздел 6.3, задания 23 – 24
20.		УК-1-В4	раздел 6.3, задания 25 – 26
21.		УК-1-В5	раздел 6.3, задания 26 – 28
22.		УК-1-В6	раздел 6.3, задания 29 – 30
23.		УК-1-В7	раздел 6.3, задания 31 – 33
24.		УК-1-В8	раздел 6.3, задания 34 – 47

### 7.3. ФОС для промежуточной аттестации

#### 7.3.1 Задания для оценки знаний

№	Формируемая компетенция	Показатели результата обучения	ФОС для оценки знаний
1	Способен осуществлять поиск, критический анализ и синтез информации,	УК-1-31	Вопросы к экзамену (1) 01 – 15
2		УК-1-32	Вопросы к экзамену (1) 16 – 30
3		УК-1-33	Вопросы к экзамену (1) 31 – 50
4		УК-1-34	Вопросы к экзамену (1) 51 – 70



5	применять системный	УК-1-35	Вопросы к экзамену (2) 01 – 20
6	подход для решения поставленных задач (УК-1)	УК-1-36	Вопросы к экзамену (2) 21 – 40
7		УК-1-37	Вопросы к экзамену (2) 41 – 60
8		УК-1-38	Вопросы к экзамену (2) 61 – 92

**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 используются 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 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?

### *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 (побитовые) 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 свойство компонента?
89. Что такое событие?
90. Чем отличается визуальный компонент от не визуального?
91. Какие стандартные диалоги Вам известны?
92. Что такое модальный диалог?

### 7.3.2. Задания для оценки умений

В качестве фондов оценочных средств для оценки умений обучающегося используются задания 1-17, рекомендованные для выполнения в часы самостоятельной работы (раздел 6.2).

### 7.3.3. Задания для оценки навыков, владений, опыта деятельности

В качестве фондов оценочных средств для оценки навыков, владений, опыта, деятельности обучающегося используются задания 18-47, 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) 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>
- 2) Информатика [Электронный ресурс] : учебное пособие для студентов первого курса очной и заочной форм обучения / . — Электрон. текстовые данные. — Тамбов: Тамбовский государственный технический университет, ЭБС АСВ, 2015. — 158 с. — 978-

5-8265-1490-0. — Режим доступа: <http://www.iprbookshop.ru/64094.html>

3) Информатика [Электронный ресурс] : учебное пособие / . — Электрон. текстовые данные. — Ставрополь: Северо-Кавказский федеральный университет, 2016. — 178 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/66024.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 C/C++ [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) Истомина Е.П., Неклюдов С.Ю., Романченко В.И. Информатика и программирование: Учебник - 2-е изд. – М.: Андреевский ИД, 2008. (ГРИФ)

4) Гуда А.Н., Бутакова М.А., Нечитайло Н.М., Чернов А.В. Информатика и программирование: Компьютерный практикум. – М.: Дашков и К, 2009 (Гриф)

## 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 Aspia, 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 СА 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\\_programmin](http://www.math.spbu.ru/user/ant/all_articles/057_Terekhov_Technology_programmin)

- [g.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 оснащённости 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: А.С. Лабузов



(подпись)

## Аннотация рабочей программы учебной дисциплины

### ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ

Учебная дисциплина «Информатика и программирование» изучается обучающимися, осваивающими образовательную программу «Прикладная информатика» по профилю Прикладная информатика в экономике в соответствии с Федеральным государственным образовательным стандартом высшего образования по направлению 09.03.03 Прикладная информатика (уровень бакалавриата), утвержденным приказом Министерства образования и науки РФ от 19.09.2017 N 922 (ФГОС ВО 3++).

Цель 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, развитие у студентов навыков программирования, способностей к самостоятельной творческой работе, умения применять методы программирования для решения различных задач прикладных дисциплин. Обучение алгоритмам обработки числовой и текстовой информации, способам записи алгоритмов, средствам реализации алгоритма на языке программирования высокого уровня.

Учебная дисциплина «Информатика и программирование» изучается студентами, осваивающими образовательную программу бакалавров по направлению 09.03.03 «Прикладная информатика», она относится к обязательной части учебного плана, изучается по заочной форме обучения в ходе 1 сессии и 2 сессии 1 курса.

Изучение учебной дисциплины направлено на подготовку обучающихся к осуществлению деятельности по концептуальному, функциональному и логическому проектированию систем среднего и крупного масштаба и сложности, планированию разработки или восстановления требований к системе, анализу проблемной ситуации заинтересованных лиц, разработке бизнес-требований заинтересованных лиц, постановки целей создания системы, разработки концепции системы и технического задания на систему, организации оценки соответствия требованиям существующих систем и их аналогов, представлению концепции, технического задания на систему и изменений в них заинтересованным лицам, организации согласования требований к системе, разработке шаблонов документов требований, постановке задачи на разработку требований к подсистемам и контроль их качества, сопровождению приемочных испытаний и ввода в эксплуатацию системы, обработке запросов на изменение требований к системе, определенных профессиональным стандартом «Системный аналитик», утвержденного приказом Министерства труда и социальной защиты Российской Федерации от 28.10.2014 N 809н (Регистрационный номер №34882).

В процессе изучения учебной дисциплины студенты должны овладеть универсальной компетенцией - способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач (УК-1).