

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Факультет информатики и информационных технологий

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Информационные технологии и программирование

Кафедра информационных систем и технологий программирования

Образовательная программа
09.03.03 Прикладная информатика

Направленность (профиль) программы
Информационные системы и программирование
Прикладная информатика в экономике и управлении
Прикладная информатика в юриспруденции

Уровень высшего образования
бакалавриат

Форма обучения

Очная

Статус дисциплины:
входит в обязательную часть ОПОП

Махачкала, 2022

Рабочая программа дисциплины «Информационные технологии и программирование» составлена в 2022 году в соответствии с требованиями ФГОС ВО – бакалавриат по направлению подготовки 09.03.03 Прикладная информатика от «19» сентября 2017г. № 922.

Разработчик(и): кафедра информационных систем и технологий программирования, доц. Баммаева Г.А.

Рабочая программа дисциплины одобрена:
на заседании кафедры ИСиТП от «01» марта 2022г., протокол № 8

Зав. кафедрой  Исмиханов З.Н.
(подпись)

на заседании Методической комиссии факультета ИиИТ
от «17» марта 2022г., протокол № 7

Председатель  Бакмаев А.Ш.
(подпись)

Рабочая программа дисциплины согласована с учебно-методическим управлением «31» марта 2022г.

Начальник УМУ  Гасангаджиева А.Г.

Аннотация рабочей программы дисциплины

Дисциплина входит в обязательную часть ОПОП образовательной программы бакалавриата по направлению 09.03.03 Прикладная информатика.

Дисциплина реализуется в факультете информатики и информационных технологий кафедрой информационных систем и технологий программирования.

Содержание дисциплины охватывает круг вопросов, связанных с систематизацией знаний в области информатики и информационных технологий, полученных в процессе изучения школьной программы, а так же углубление их с учетом профиля, приобретение компетенции в использовании информационных и коммуникационных технологий на уровне опытного пользователя, готовности к освоению на этой основе профильных профессиональных технологий; развитие основных навыков работы с информацией; получение практических навыков по основам алгоритмизации вычислительных процессов и программированию решения вычислительных и других задач; умение самостоятельно применять эти навыки соответственно учебным целям; знакомство с необходимым набором профессиональных инструментов.

Дисциплина нацелена на формирование следующих общепрофессиональных компетенций выпускника: ОПК-7.

Преподавание дисциплины предусматривает проведение следующих видов учебных занятий: лекции, лабораторные занятия, самостоятельная работа.

Рабочая программа дисциплины предусматривает проведение следующих видов контроля успеваемости: текущий контроль в форме опроса, тестов, контрольных работ и промежуточный контроль в форме экзамена.

Объем дисциплины 4 зачетные единицы, в том числе в 144 академических часах по видам учебных занятий

форма обучения - очная

| Семестр | Учебные занятия | | | | | | | СРС, в том числе экзамен | Форма промежуточной аттестации | |
|---------|-----------------|--|--------|----------------------|----------------------|-----|-----|--------------------------|--------------------------------|--------------|
| | в том числе: | | | | | | | | | |
| | Всего | Контактная работа обучающихся с преподавателем | | | | | КСП | | | Консультации |
| | | Всего | Лекции | Лабораторные занятия | Практические занятия | КСП | | | | |
| 1 | 144 | 54 | 18 | 36 | | | 90 | экзамен | | |

1. Цели освоения дисциплины

Целями освоения дисциплины являются формирование у студентов фундамента современной информационной культуры; обеспечение устойчивых навыков работы на персональном компьютере (ПК) в условиях локальных и глобальных вычислительных сетей и систем телекоммуникаций; применение программных средств (ПС) общего назначения; освоение основ современной методологии разработки компьютерных информационных систем и практической реализации ее основных элементов с использованием ПК и типовых программных продуктов; формирование навыков создания программных продуктов с использованием современных средств программирования, изучение технологии использования средств программирования.

2. Место дисциплины в структуре ОПОП бакалавриата

Дисциплина «Информационные технологии и программирование» входит в обязательную часть дисциплин учебного плана направления (специальности) 09.03.03 - Прикладная информатика (в экономике, в юриспруденции).

Знание дисциплины «Информационные технологии и программирование» является важной составляющей общей программистской культуры и навыков программирования выпускника. Эти знания необходимы при решении практических задач из разнообразных прикладных областей, таких, как объектно-ориентированное программирование, разработка программных приложений, информационные системы и технологии и т.д.

Для успешного освоения данного курса студент должен иметь элементарные знания по школьному курсу дисциплины «Информатика».

Список дисциплин, для изучения которых необходимы знания данного курса

1. Объектно-ориентированное программирование
2. Разработка программных приложений
3. Корпоративные информационные системы
4. Бухгалтерские информационные системы
5. Операционные системы
6. Программная инженерия

3. Компетенции обучающегося, формируемые в результате освоения дисциплины (перечень планируемых результатов обучения)

| Код и наименование компетенции из ОПОП | Код и наименование индикатора достижения компетенций (в соответствии с ОПОП) | Планируемые результаты обучения | Процедура освоения |
|---|---|--|---|
| ОПК-7. Способен разрабатывать алгоритмы и программы, пригодные для практического применения | ИД 7.1. ОПК-7.1. Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий. ИД 7.2. ОПК-7.2. Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ. ИД 7.3. ОПК-7.3. Владеет | <i>Знает:</i> методы разработки алгоритмов и программ, основы информатики и программирования, проектирования, конструирования и тестирования программных продуктов. <i>Умеет:</i> разрабатывать алгоритмы и программы, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов <i>Владеет:</i> способами разработки алгоритмов и программ, основами информатики и программирования, применять их к проектированию, конструированию и тестированию | Опрос, тестирование, контрольная работа |

| | | | |
|--|---|------------------------|--|
| | навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач. | программных продуктов. | |
|--|---|------------------------|--|

4. Объем, структура и содержание дисциплины

4.1. Объем дисциплины составляет 4 зачетных единиц, 144 академических часов.

4.2. Структура дисциплины

4.2.1. Структура дисциплины в очной форме

| № | Разделы и темы дисциплины | Семестр | Неделя семестра | Виды учебной работы и самостоятельная работа, в час | | | Самостоятельная работа | Формы контроля |
|--|--|---------|-----------------|---|----------------------|----------------------|------------------------|---|
| | | | | Лекции | Практические занятия | Лабораторные занятия | | |
| Модуль 1. Основные понятия и методы теории информации и кодирования | | | | | | | | |
| 1 | Введение в дисциплину. Понятие информации и информационных технологий. | 1 | | 2 | | 12 | 6 | отчеты выполнения лабораторных заданий, контрольная работа, опрос |
| 2 | Состав, архитектура и функционирование ПК | 1 | | 2 | | 12 | 6 | отчеты выполнения лабораторных заданий, контрольная работа, опрос |
| 3 | Данные. Представление данных в памяти компьютера | 1 | | 2 | | 12 | 6 | отчеты выполнения лабораторных заданий, контрольная работа, опрос |
| <i>Итого по модулю 1:</i> | | | | 6 | | 36 | 18 | |
| Модуль 2. Программное обеспечение ПЭВМ | | | | | | | | |
| 4 | Программное обеспечение ЭВМ. | 1 | | 2 | | 12 | 6 | отчеты выполнения лабораторных заданий, контрольная работа, опрос |
| 5 | Классы профессиональных | 1 | | 2 | | 12 | 6 | отчеты выполнения |

| № | Разделы и темы дисциплины | Семестр | Неделя семестра | Виды учебной работы и самостоятельная работа, в час | | | Самостоятельная работа | Формы контроля |
|--|---|---------|-----------------|---|----------------------|----------------------|------------------------|---|
| | | | | Лекции | Практические занятия | Лабораторные занятия | | |
| | компьютерных программ. | | | | | | | лабораторных заданий, контрольная работа, опрос |
| 6 | Системы компьютерной графики | 1 | | 2 | | 12 | 6 | отчеты выполнения лабораторных заданий, контрольная работа, опрос |
| | <i>Итого по модулю 2:</i> | | | 6 | | 36 | 18 | |
| Модуль 3. Алгоритм, алгоритмизация и программирование. Основы языка программирования Python | | | | | | | | |
| 7 | Алгоритмы и их свойства, формализация понятия алгоритм | 1 | | 2 | | 12 | 6 | отчеты выполнения лабораторных заданий, контрольная работа, опрос |
| 8 | Языки программирования и их назначение. Краткая история языков программирования | 1 | | 2 | | 12 | 6 | отчеты выполнения лабораторных заданий, контрольная работа, опрос |
| 9 | Знакомство с Python. Ввод и вывод данных. Логические выражения и операторы | 1 | | 2 | | 12 | 6 | отчеты выполнения лабораторных заданий, контрольная работа, опрос |
| | <i>Итого по модулю 3:</i> | | | 6 | | 36 | 18 | |
| Модуль 4. Подготовка к экзамену | | | | | | | | |
| | <i>Итого по модулю 4:</i> | | | | | | 36 | экзамен |
| | <i>Итого:</i> | | | 18 | | 36 | 90 | |
| | | | | | | | | |

4.3. Содержание дисциплины, структурированное по темам (разделам)

4.3.1. Содержание лекционных занятий по дисциплине

| № п/п | Наименование темы | Трудоемкость | Содержание | Формируемые компетенции | Результаты освоения (знать, уметь, владеть) | Технологии обучения |
|------------------------|--|--------------|--|-------------------------|---|----------------------------------|
| 1 семестр | | | | | | |
| Лекционный курс | | | | | | |
| 1 | Введение в дисциплину. Понятие информации и информационных технологий. | 2 | Понятие «информатика», информационные технологии. Составные части информатики. Определение информации. Формы представления и свойства информации. Классификация информации. | ОПК 7.1 | Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий. | Интерактивная лекция, обсуждение |
| 2 | Состав, архитектура и функционирование ПК | 2 | Общее понятие об аппаратном обеспечении. | ОПК 7.1 | Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий. | Интерактивная лекция, обсуждение |
| 3 | Данные. Представление данных в памяти компьютера | 2 | Данные. Операции над данными. Основные принципы кодирования информации: представление текстовой, графической, других видов информации. Двоичная система кодирования символов. Таблица ASCII кодов. Кодировка Unicode. Понятие файловой структуры | ОПК 7.1 | Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий. | Интерактивная лекция, обсуждение |

| № п/п | Наименование темы | Трудоемкость | Содержание | Формируемые компетенции | Результаты освоения (знать, уметь, владеть) | Технологии обучения |
|-------|--|--------------|---|-------------------------|---|----------------------------------|
| 4 | Программное обеспечение ЭВМ. | 2 | Программный принцип управления компьютером. Классификация программного обеспечения. | ОПК 7.1 | Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий. | Интерактивная лекция, обсуждение |
| 5 | Классы профессиональных компьютерных программ. | 2 | Роль и место программ профессионального уровня в современном обществе. Классификация и краткая характеристика компьютерных программ | ОПК 7.1 | Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий. | |
| 6 | Системы компьютерной графики | 2 | Виды компьютерной графики. Растровая и векторная графика. Трехмерная графика. Обзор цветowych моделей. Обзор основных графических пакетов | ОПК 7.1 | Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий. | |
| 7 | Алгоритмы и их свойства, формализация понятия алгоритм | 2 | Этапы решения задач. Понятие алгоритма. Свойства и формы записи алгоритмов. Алгоритмические машины Поста и Тьюринга | ОПК 7.1 | Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки | Интерактивная лекция, обсуждение |

| № п/п | Наименование темы | Трудоемкость | Содержание | Формируемые компетенции | Результаты освоения (знать, уметь, владеть) | Технологии обучения |
|-------|---|--------------|--|-------------------------|---|----------------------------------|
| | | | | | информационных систем и технологий. | |
| 8 | Языки программирования и их назначение. Краткая история языков программирования | 2 | История развития языков программирования. Сравнительный обзор высокоуровневых языков программирования и общие понятия высокоуровневых языков программирования. Движущие силы развития языков программирования. Классификация языков программирования. | ОПК-7.1 | Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий. | Интерактивная лекция, обсуждение |
| 9 | Знакомство с Python. Ввод и вывод данных. Логические выражения и операторы | 2 | Краткая историческая справка и основные особенности языка. Дзен Питона. Режимы работы с интерпретатором. Создание скриптов. Данные и их типы. Операции в программировании. Изменение типов данных. Логические выражения и логический тип данных. Логические операторы. Сложные логические выражения. | ОПК-7.1 | Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий. | Интерактивная лекция, обсуждение |

Лабораторные занятия

| | | | | | | |
|---|---|---|--|---------|---|---|
| 1 | Основы работы с операционным и системами семейства Windows Объекты Windows | 4 | Автоматический запуск приложений. Редактирование свойств типов файлов. Настройка периферийных устройств | ОПК 7.3 | Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач. | Защита лабораторных заданий, выполнение индивидуального задания |
| 2 | Стандартные настройки Windows | 2 | Прямой код, обратный (инверсный) код, дополнительный код | ОПК 7.3 | Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач. | Защита лабораторных заданий, выполнение индивидуального задания |
| 3 | Системы счисления. Позиционные системы счисления | 2 | Операции над данными. Методы кодирования данных. Понятие файловой структуры | ОПК 7.3 | Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач. | Защита лабораторных заданий, выполнение индивидуального задания |
| 4 | Данные. Операции над данными | 4 | Создание текстовых файлов, автоматизация работы с текстом, применение редактора формул и графических объектов, создание таблиц и списков | ОПК 7.3 | Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач. | Защита лабораторных заданий, выполнение индивидуального задания |
| 5 | Работа с текстовым процессором MS Word | 4 | Редактирование рабочей книги, построение диаграмм, работа с формулами, использование логических формул | ОПК 7.3 | Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач. | Защита лабораторных заданий, выполнение индивидуального задания |

| | | | | | | |
|---|---|---|---|---------|---|---|
| 6 | Работа с табличным процессором MS Excel | 4 | Технология создания баз данных и работы с ними в среде | ОПК 7.3 | Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач. | Защита лабораторных заданий, выполнение индивидуального задания |
| 7 | Работа с СУБД MS Access | 8 | Режимы работы с интерпретатором. Создание скриптов. | ОПК-7.3 | Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач. | Защита лабораторных заданий, выполнение индивидуального задания |
| 8 | Знакомство с Python. | 4 | Данные и их типы. Операции в программировании. Изменение типов данных. Переменные. Вывод данных. Функция print(). Ввод данных. Функция input(). | ОПК-7.3 | Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач. | Защита лабораторных заданий, выполнение индивидуального задания |
| 9 | Ввод и вывод данных. Логические выражения и операторы | 4 | Данные и их типы. Операции в программировании. Изменение типов данных. Переменные. Вывод данных. Функция print(). Ввод данных. Функция input(). | ОПК-7.3 | Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач. | Защита лабораторных заданий, выполнение индивидуального задания |

4.3.2. Содержание лабораторно-практических занятий по дисциплине

Лабораторная работа. Использование стилей для создания документов

Цель работы: Оформление структуры документа и построение оглавления с использованием стилей Word.

Задание:

Создать с использованием стилей документ (отчет по научной работе) со следующей структурой:

Введение

Глава 1

1.1.

1.2.

1.3.

Глава 2

2.1.

2.2.

Глава 3

3.1.

3.2.

3.3.

3.4.

Заключение

Литература

Приложение

1. Цвет шрифта - черный, шрифт текста документа –TimesNewRoman, размер 14 pt, полужирный шрифт не используется. Выравнивание текста - по ширине страницы.

2. Отчет следует печатать, соблюдая следующие размеры полей: правое- 10 мм, верхнее и нижнее - 20 мм, левое -30мм.

3. «Введение», «Заключение», «Литература», «Приложения» служат заголовками структурных элементов отчета. Заголовки структурных элементов следует располагать в середине строки без точки в конце и печатать прописными (большими) буквами, не подчеркивая, шрифт -Arial, размер 14pt.

4. Разделы отчета (главы) должны быть пронумерованы арабскими цифрами в пределах всего отчета и записываться с абзацного отступа. После номера ставится точка и пишется название раздела (главы) прописными буквами без точки в конце, шрифт -Arial, размер 14pt.

5. Структурные элементы отчета и главы начинаются с новой страницы, параграфы продолжают на той же странице. Параграфы на составные части не подразделяются.

6. Параграфы следует нумеровать арабскими цифрами в пределах каждого раздела отчета (главы). Номер параграфа должен состоять из номера раздела (главы) и номера параграфа, разделенных точкой. Заголовки параграфов печатаются с абзацного отступа 15 мм строчными буквами (кроме первой прописной). Точка в конце не ставится.

7. Переносы слов в заголовках не допускаются. Количество слов в заголовках - от двух до 14 (не более двухстрок).

8. Заголовок параграфа не должен быть последней строкой на странице.

9. После заголовков структурных элементов и параграфов устанавливается интервал - одна пустая строка. Расстояние между последней строкой текста параграфа и заголовком следующего параграфа составляет двестроки.

10. Каждый абзац текста начинается с красной строки, при этом отступ составляет 5 печатных знаков от начала текста или 15мм.

Методика выполнения работы

1. Создаем структуродокумента:

- Копируем заданную структуру в свой отчет по работе.

• Выбираем на главной вкладке Стили и изменяем стиль Заголовок1: кликаем правой кнопкой на Заголовок1, выбираем пункт Изменить, устанавливаем цвет шрифта - черный, шрифт –TimesNewRoman, размер 14 pt, обычный, по центру, межстрочный интервал - одинарный. В меню Формат выбираем пункт Шрифт и выбираем все прописные. В меню Формат выбираем пункт Абзац и выбираем в пункте Положение на странице - с новой страницы.

• Выделяем по очереди все структурные элементы и кликаем затем на стильЗаголовок1.

• Изменяем стиль Заголовок2 и Заголовок3 (подобным образом в соответствии с заданием).

• Выделяем по очереди все главы и кликаем на стильЗаголовок2.

• Выделяем по очереди все параграфы и кликаем на стильЗаголовок3.

2. Делаем нумерацию страниц: Вставка → Номер страницы → Внизу страницы.

3. Чтобы посмотреть созданную структуру, выберите вкладку Вид и отметьте Область навигации. Слева появится созданная структура.

4. Вставляем текст в указанном формате в соответствующие разделы документа (не более одной страницы в каждом разделе). Названия структурных заголовков, параграфов, их текст либо придумать, либо взять, например, из данного лабораторного практикума.

5. Создаем стиль Текст по заданному для него формату и применяем его для текста документа. Стиль Текст основан на стиле Красная строка.

6. С помощью стилей сформируем оглавление в конце созданного документа:

• Устанавливаем курсор в конец документа.

• На вкладке Ссылки кликаем мышкой по кнопке Оглавление. Из выпадающего списка выбираем Автособираемое оглавление 1. Появится оглавление, которое затем необходимо привести к заданному формату.

Лабораторная работа: Работа со списками. Операции над списками в Python

Цель работы: Изучение одномерных массивов в Python.

Массивы (списки) в Python- это определенное количество элементов одного типа, которые имеют общее имя, и каждый элемент имеет свой индекс - порядковый номер.

Часто для работы с массивами используются списки.

Список (list) - это структура данных для хранения объектов различных типов.

Списки являются упорядоченными последовательностями, которые состоят из различных типов данных, заключающихся в квадратные скобки [] и отделяющиеся друг от друга с помощью запятой.

Создание списков на Python.

Создать список можно несколькими способами

1. Получение списка через присваивание конкретных значений.

Так выглядит в коде Python пустой список:

```
s = [] # Пустой список
```

Примеры создания списков со значениями:

```

l=[5, 75, -4, 7, -51] # список целых чисел
l=[1.13, 5.34, 12.63, 4.6, 34.0, 12.8] # список из вещественных чисел
l=["Оля", "Владимир", "Михаил", "Дарья"] # список из строк
l=["Москва", "Иванов", 12, 124] # смешанный список
l=[[0, 0, 0], [1, 0, 1], [1, 1, 0]] # список, состоящий из списков
l=['s', 'p', ['isok'], 2] # список из значений и списка

```

Списки можно складывать (конкатенировать) с помощью знака «+»:

```

l=[1, 3]+[4, 23]+[5]
print('l=[1, 3]+[4, 23]+[5] =', l)

```

Результат:

```

>>>
l=[1, 3]+[4, 23]+[5] = [1, 3, 4, 23, 5]
>>> |

```

2. Создание списка при помощи функции Split().

Используя функцию split в Python можно получить из строки список.

```
stroka = "Привет, страна"
```

```
lst=stroka.split(",")
```

```

stroka = "Здравствуй, Дедушка Мороз" #stroka - строка
lst=stroka.split(",") #lst - список
print('stroka = ', stroka)
print('lst=stroka.split(","):', lst)

```

Результат

```

===== RESTART: C:/Users/maxim/Desktop/ex_list_
stroka = Здравствуй, Дедушка Мороз
lst=stroka.split(","): ['Здравствуй', ' Дедушка Мороз']

```

3. Генераторы списков.

В Python создать список можно также при помощи генераторов.

Первый способ.

Сложение одинаковых списков заменяется умножением:

список из 10 элементов, заполненный единицами

```
l = [1]*10
```

Второй способ.

Пример 1.

```
l = [i for i in range(10)]
```

Пример 2.

```

c = [c * 3 for c in 'list']
print(c) # ['lll', 'iii', 'sss', 'ttt']

```

```

Создание списка из строки.
l = list (строка):
['c', 'т', 'р', 'о', 'к', 'а']

Создание списка при помощи функции Split().
stroka=" Hello, friend "
lst=stroka.split(","):
['Hello', ' friend']

Генераторы списков.
Первый способ.
l = [1]*10:
[1, 1, 1, 1, 1, 1, 1, 1, 1, 1]

Второй способ. Пример 1.
l = [i for i in range(10)]:
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

Второй способ. Пример 2.
c=[c*3 for c in "list"]:
['lll', 'iii', 'sss', 'ttt']

```

Примеры использования генераторов списка.

Пример 1.

Заполнить список квадратами чисел от 0 до 9, используя генератор списка.

Решение:

```
l = [i*i for i in range(10)]
```

Пример 2.

Заполнить список числами, где каждое последующее число больше на 2.

```
l = [(i+1)+i for i in range(10)]
```

```
print(l)
```

```

Заполнить список квадратами чисел от 0 до 9, используя генератор списка.
l = [i*i for i in range(10)]:
[0, 1, 4, 9, 16, 25, 36, 49, 64, 81]

```

```

Заполнить список числами, где каждое последующее число больше на 2.
l = [(i+1)+i for i in range(10)]:
[1, 3, 5, 7, 9, 11, 13, 15, 17, 19]

```

Модуль random предоставляет функции для генерации случайных чисел, букв, случайного выбора элементов последовательности.
random.randint(A, B) - случайное целое число N, $A \leq N \leq B$.
random.random() - случайное число от 0 до 1.

Случайные числа в списке:

10 чисел, сгенерированных случайным образом в диапазоне (10,80)

```
from random import randint
```

```
l = [randint(10,80) for x in range(10)]
```

10 чисел, сгенерированных случайным образом в диапазоне (0,1)

```
l = [random() for i in range(10)]
```

```

from random import *
l = [randint(10,80) for i in range(10)]
print('10 чисел, сгенерированных случайным образом в диапазоне (10,80).')
print('l = [randint(10,80) for x in range(10)]:')
print(l)
print()

```

```

l = [random() for i in range(10)]
print('10 чисел сгенерированных в диапазоне от 0 до 1.')
print('l = [random() for i in range(10)]:')
for i in range(len(l)):
    print ('{:0.2f}'.format(l[i]), end = " ")

```

```

10 чисел, сгенерированных случайным образом в диапазоне (10,80).
l = [randint(10,80) for x in range(10)]:
[70, 33, 79, 61, 34, 27, 11, 55, 52, 31]

```

```

10 чисел сгенерированных в диапазоне от 0 до 1.
l = [random() for i in range(10)]:
0.66 0.97 0.87 0.57 0.54 0.83 0.57 0.65 0.04 0.07

```

4. Ввод списка (массива) в языке Python.

Для ввода элементов списка используется цикл for и команда range ():

```

for i in range(N):
    x[i] = int(input())

```

Более простой вариант ввода списка:

```

x = [ int(input()) for i in range(N) ]

```

```

print('Ввод списка. Пример 1:')
x=[]
for i in range(4):
    x.append(int(input()))
print(x)

x=[]
print('Ввод списка. Пример 2:')
x = [ int(input()) for i in range(4) ]
print(x)

```

```

Ввод списка. Пример 1:

```

```

45
4
85
2
[45, 4, 85, 2]

```

```

Ввод списка. Пример 2:

```

```

4
5
7
8
[4, 5, 7, 8]

```

Функция int здесь используется для того, чтобы строка, введенная пользователем, преобразовывалась в целые числа.

5. Вывод списка (массива) в языке Python.

Вывод целого списка (массива):

```

print(L)

```


Поэлементный вывод списка (массива):

```
for i in range(N):  
    print( L[i], end = " " )
```

```
....  
Вывод целого списка (массива)  
[1, 56, 6, 3, 6, 7, 3, 37, 7, 37, 37]
```

```
Поэлементный вывод списка (массива)  
1 56 6 3 6 7 3 37 7 37 37
```

Методы списков

| Метод | Что делает |
|---------------------------------------|---|
| list.append(x) | Добавляет элемент в конец списка |
| list.extend(L) | Расширяет список list, добавляя в конец все элементы списка L |
| list.insert(i, x) | Вставляет перед i-ым элементом значение x |
| list.remove(x) | Удаляет первый элемент в списке, имеющий значение x. ValueError, если такого элемента не существует |
| list.pop([i]) | Удаляет i-ый элемент и возвращает его. Если индекс не указан, удаляется последний элемент |
| list.index(x, [start [, end]]) | Возвращает положение первого элемента со значением x (при этом поиск ведется от start до end) |
| list.count(x) | Возвращает количество элементов со значением x |
| list.reverse() | Разворачивает список |
| list.copy() | Поверхностная копия списка |
| list.clear() | Очищает список |

Ниже приведена программа, демонстрирующая методы работы списков.

```
a=[0, 2, 2, 2, 4] #список a  
b=[5, 6, 7, 2, 9] #список b  
print ('Исходный список a:', a)  
print ('Исходный список b:', b)  
x=99  
y=5  
  
a.append(x)  
print ('a.append(x):', a)  
  
a.extend(b)  
print ('a.extend(b):', a)  
  
a.insert(3, x)  
print ('a.insert(3, x):', a)  
  
a.remove(x)  
print ('a.remove(x):', a)  
  
print ('a.pop(5):', a.pop(5))  
print (a)  
  
print ('a.index(y, 0, len(a)):', a.index(y, 0, len(a)))  
  
print ('a.count(2):', a.count(2))  
  
a.reverse()  
print ('a.reverse():', a)  
  
z=a.copy()  
print ('z=a.copy():', z)  
  
z.clear()  
print ('z.clear():')  
print ('z =', z)
```

Пример программы на Python

```
Исходный список a: [0, 2, 2, 2, 4]
Исходный список b: [5, 6, 7, 2, 9]
a.append(x): [0, 2, 2, 2, 4, 99]
a.extend(b): [0, 2, 2, 2, 4, 99, 5, 6, 7, 2, 9]
a.insert(3,x): [0, 2, 2, 99, 2, 4, 99, 5, 6, 7, 2, 9]
a.remove(x): [0, 2, 2, 2, 4, 99, 5, 6, 7, 2, 9]
a.pop(5): 99
[0, 2, 2, 2, 4, 5, 6, 7, 2, 9]
a.index(y,0,len(a)): 5
a.count(2): 4
a.reverse(): [9, 2, 7, 6, 5, 4, 2, 2, 2, 0]
z=a.copy(): [9, 2, 7, 6, 5, 4, 2, 2, 2, 0]
z.clear():
z = []
```

Результат выполнения программы

Пример выполнения задания

Вариант 0

1. Из массива X длиной n, среди элементов которого есть положительные, отрицательные и равные нулю, сформировать новый массив Y, взяв в него только те элементы из X, которые больше по модулю заданного числа M. Вывести на экран число M, данный и полученные массивы.

Решение:

```
n=int(input('Введите длину массива\n'))
m=int(input('Введите число M\n'))
x=[]
y=[]
for i in range(n):
    print('Введите ',i,'элемент:')
    x.append(int(input()))
for i in range(n):
    if abs(x[i])>m:
        y.append(x[i])
print('Введённое число M:',m)
print('Массив X:',x)
print('Массив Y:',y)
```

```
Введите длину массива
5
Введите число M
20
Введите 0 элемент:
21
Введите 1 элемент:
22
Введите 2 элемент:
5
Введите 3 элемент:
6
Введите 4 элемент:
8
Введённое число M: 20
Массив X: [21, 22, 5, 6, 8]
Массив Y: [21, 22]
```

Задания для самостоятельного выполнения

Вариант 1

1. Дан одномерный массив, состоящий из N целочисленных элементов. Ввести массив с клавиатуры. Найти максимальный элемент. Вывести массив на экран в обратном порядке.

2. В массиве действительных чисел все нулевые элементы заменить на среднее арифметическое всех элементов массива.

Вариант 2

1. Дан одномерный массив, состоящий из N целочисленных элементов. Ввести массив с клавиатуры. Найти минимальный элемент. Вывести индекс минимального элемента на экран.

2. Дан массив целых чисел. Переписать все положительные элементы во второй массив, а остальные - в третий.

Вариант 3

1. В одномерном числовом массиве D длиной n вычислить сумму элементов с нечетными индексами. Вывести на экран массив D , полученную сумму.

2. Дан одномерный массив из 8 элементов. Заменить все элементы массива меньшие 15 их удвоенными значениями. Вывести на экран монитора преобразованный массив.

Вариант 4

1. Дан массив целых чисел. Найти максимальный элемент массива и его порядковый номер.

2. Дан одномерный массив целого типа. Получить другой массив, состоящий только из нечетных чисел исходного массива или сообщить, что таких чисел нет. Полученный массив вывести в порядке убывания элементов.

Вариант 5

1. Дан одномерный массив из 10 целых чисел. Вывести пары отрицательных чисел, стоящих рядом.

2. Дан целочисленный массив размера 10. Создать новый массив, удалив все одинаковые элементы, оставив их 1 раз.

Вариант 6

1. Дан одномерный массив из 10 целых чисел. Найти максимальный элемент и сравнить с ним остальные элементы. Вывести количество меньших максимального и больших максимального элемента.

2. Одномерный массив из 10-и целых чисел заполнить с клавиатуры, определить сумму тех чисел, которые >5 .

Вариант 7

1. Дан массив целых чисел. Найти сумму элементов с четными номерами и произведение элементов с нечетными номерами. Вывести сумму и произведение.

2. Переставить в одномерном массиве минимальный элемент и максимальный.

Вариант 8

1. Найдите сумму и произведение элементов списка. Результаты вывести на экран.

2. В массиве действительных чисел все нулевые элементы заменить на среднее арифметическое всех элементов массива.

Вариант 9

1. Дан одномерный массив, состоящий из N вещественных элементов. Ввести массив с клавиатуры. Найти и вывести минимальный по модулю элемент. Вывести массив на экран в обратном порядке.

2. Даны массивы A и B одинакового размера 10. Вывести исходные массивы. Поменять местами их содержимое и вывести в начале элементы преобразованного массива A , а затем - элементы преобразованного массива B .

Вариант 10

1. Определите, есть ли в списке повторяющиеся элементы, если да, то вывести на экран это значение, иначе сообщение об их отсутствии.

2. Дан одномерный массив из 15 элементов. Элементам массива меньше 10 присвоить нулевые значения, а элементам больше 20 присвоить 1. Вывести на экран монитора первоначальный и преобразованный массивы в строчку.

Вариант 11

1 Найти наибольший элемент списка, который делится на 2 без остатка и вывести его на экран.

2. Дан одномерный массив целого типа. Получить другой массив, состоящий только из четных чисел исходного массива, меньше 10, или сообщить, что таких чисел нет. Полученный массив вывести в порядке возрастания элементов.

Вариант 12

1. Найти наименьший нечетный элемент списка и вывести его на экран.

2. Даны массивы А и В одинакового размера 10. Поменять местами их содержимое и вывести вначале элементы преобразованного массива А, а затем - элементы преобразованного массива В.

Вариант 13

1. Дан одномерный массив целых чисел. Проверить, есть ли в нем одинаковые элементы. Вывести эти элементы и их индексы.

2. Дан одномерный массив из 8 элементов. Заменить все элементы массива меньшие 15 их удвоенными значениями. Вывести на экран монитора преобразованный массив.

Вариант 14

1. Найти максимальный элемент численного массива и поменять его местами с минимальным.

2. Программа заполняет одномерный массив из 10 целых чисел числами, считанными с клавиатуры. Определить среднее арифметическое всех чисел массива. Заменить элементы массива большие среднего арифметического на 1.

Вариант 15

1. Определите, есть ли в списке повторяющиеся элементы, если да, то вывести на экран эти значения.

2. Дан одномерный массив целого типа. Получить другой массив, состоящий только из нечетных чисел исходного массива или сообщить, что таких чисел нет. Полученный массив вывести в порядке убывания элементов.

5. Образовательные технологии

Образовательные технологии, используемые при изучении курса, предусматривают применение инновационных методов обучения. Использование персональных компьютеров при выполнении лабораторных работ и сдаче итогового экзамена. Чтение лекций с использованием компьютера и проектора, проведение лабораторных работ в компьютерном классе с доступом к интернету.

При реализации учебной дисциплины используются электронные практикумы, электронные учебники, презентации средства диагностики и контроля разработанные специалистами кафедры т.д.

Удельный вес занятий, проводимых в интерактивной форме, составляет 20% аудиторных занятий.

6. Учебно-методическое обеспечение самостоятельной работы студентов

Форма контроля и критерий оценок

В соответствии с учебным планом предусмотрен экзамен .

Формы контроля: текущий контроль, промежуточный контроль по модулю, итоговый контроль по дисциплине предполагают следующее распределение баллов.

Примерное распределение времени самостоятельной работы студентов

| Вид самостоятельной работы | Примерная трудоёмкость, в.ч. | Формируемые компетенции |
|--|------------------------------|-------------------------|
| | очная | |
| Текущая СРС | | |
| работа с лекционным материалом, с учебной литературой | 16 | ОПК-7 |
| опережающая самостоятельная работа (изучение нового материала до его изложения на занятиях) | 12 | ОПК-7 |
| самостоятельное изучение разделов дисциплины | 28 | ОПК-7 |
| выполнение домашних заданий, домашних контрольных работ | 16 | ОПК-7 |
| подготовка к лабораторным работам, к практическим и семинарским занятиям | 16 | ОПК-7 |
| подготовка к контрольным работам, коллоквиумам, зачётам | 16 | ОПК-7 |
| Творческая проблемно-ориентированная СРС | | |
| выполнение расчётно-графических работ | 8 | ОПК-7 |
| поиск, изучение и презентация информации по заданной проблеме, анализ научных публикаций по заданной теме | 8 | ОПК-7 |
| исследовательская работа, участие в конференциях, семинарах, олимпиадах | 12 | ОПК-7 |
| анализ данных по заданной теме, выполнение расчётов, составление схем и моделей на основе собранных данных | 12 | ОПК-7 |
| Итого СРС: | 144 | |

7. Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

7.1. Типовые контрольные задания

Примерный перечень вопросов к промежуточному контролю или экзамену по всему изучаемому курсу:

1. Определение информации. Формы представления и свойства информации
2. Классификация информации
3. Сущность понятия «правовая информация», ее виды
4. Общее понятие об аппаратном обеспечении
5. Компоненты системного блока
6. Периферийные устройства
7. Позиционные и непозиционные системы счисления. Двоичная, восьмеричная, шестнадцатеричная системы счисления
8. Алгоритмы перевода чисел из одной системы счисления в другие. Перевод чисел в системах счисления с кратными основаниями
9. Арифметические действия в двоичной системе счисления
10. Данные. Операции над данными
11. Методы кодирования данных
12. Понятие файловой структуры
13. Программный принцип управления компьютером. Классификация программного обеспечения
14. Системное программное обеспечение
15. Сервисное программное обеспечение

16. Прикладное программное обеспечение
17. Этапы решения задач. Понятие алгоритма
18. Свойства и формы записи алгоритмов
19. Линейные алгоритмы. Ветвления в алгоритмах
20. Циклические алгоритмы
21. История развития языков программирования
22. Движущие силы развития языков программирования
23. Классификация языков программирования
24. Основные понятия алгоритмических языков программирования
25. Встроенный тип str. Методы объекта str.
26. print() и форматирование вывода.
27. яя плавающей запятой. Целые числа с произвольной точностью.
28. Множества. Встроенные типы set и frozenset.
29. Инструкции и синтаксис. Составные конструкции и обработка исключений
30. Инструкции if/else/elif, логические операторы и выражения сравнения
31. Циклы while и for в Python
32. Функции в Python. Основные понятия
33. Области видимости и пространство имен в Python.
34. Передача аргументов в функцию. Специальные режимы сопоставления аргументов.
35. Парадигма объектно-ориентированного программирования. Поддержка в Python функционального программирования.

Примерные тестовые задания для проведения текущего контроля и промежуточной аттестации по итогам освоения дисциплины

1. Информатика - это наука...
 - а) о преобразовании информации
 - б) защите информации
 - в) способах получения и передачи информации
 - г) компьютерах
2. Информацию, достаточную для решения поставленной задачи, называют...
 - а) полезной
 - б) актуальной
 - в) полной
 - г) достоверной
3. Выберите вариант, в котором объемы памяти расположены в порядке возрастания
 - а) 10 бит, 2 байта, 20 бит, 1010 байт, 1 Кбайт
 - б) 10 бит, 20 бит, 2 байта, 1 Кбайт, 1010 байт
 - в) 10 бит, 20 бит, 2 байта, 1010 байт, 1 Кбайт
 - г) 10 бит, 2 байта, 20 бит, 1 Кбайт, 1010 байт
4. Файловая система определяет ...
 - а) способ организации данных на диске
 - б) физические особенности носителя
 - в) емкость диска
 - г) число пикселей на диске
5. Операторы ... являются простой конструкцией условия
 - а) If-Then
 - б) Select Case
 - в) DoWhile
 - г) DoUntil
6. Базовые структуры алгоритма
 - а) следование

- б) переключатель
 - в) ветвление
 - г) безусловный переход
 - д) цикл
 - е) условный переход
7. Среда разработки программного обеспечения - это ...
- а) компилятор кода
 - б) система программных средств, используемая для разработки программного обеспечения
 - в) программа, предназначенная для запуска других программ
 - г) программа, предназначенная для написания кода программ
8. Переменная типа Boolean может принимать значения ...
- а) 0, 1
 - б) True, False
 - в) 1, -1
 - г) -1, 0 1
9. Прикладное программное обеспечение общего назначения
- а) текстовые и графические редакторы
 - б) системы управления базами данных (СУБД)
 - в) программы сетевого планирования и управления
 - г) оболочки экспертных систем и систем искусственного интеллекта
- д) средства разработки приложений
- е) бухгалтерские программы
10. Прикладные программы называют ...
- а) утилитами
 - б) приложениями
 - в) драйверами
 - г) браузерами
11. Программные комплексы проблемно-ориентированного прикладного программного обеспечения:
- а) система «Галактика»
 - б) система автоматизированного проектирования AutoCAD
 - в) пакет офисных программ MS Office
 - г) программы оценки эффективности инвестиций ProjectExpert
 - д) комплекс программ OpenOffice
 - е) программы решения уникальных прикладных программ
 - ж) справочно-правовая система «КОНСУЛЬТАНТПЛЮС»
12. Программные средства методо-ориентированного прикладного программного обеспечения
- а) математической статистики
 - б) математического программирования (линейного, динамического, статистического);
 - в) системы управления базами данных (СУБД);
 - г) графические редакторы
 - д) теории массового обслуживания
 - е) текстовые редакторы
13. Прикладное программное обеспечение - это
- а) программы, написанные для пользователей или самими пользователями, для задания компьютеру конкретной работы
 - б) совокупность программ, необходимых для функционирования аппаратных средств компьютера

в) все программы, необходимые для организации диалога пользователя с компьютером

г) комплекс программ, с помощью которых пользователь может решать свои информационные задачи из самых разных предметных областей, не прибегая к программированию

14. Функции в электронной таблице представляют собой ...

а) программы с уникальным именем, для которой пользователь должен задать конкретные значения аргументов

б) объекты, предназначенные для выполнения математических операций; не в) содержат алфавитных и специальных символов

г) объекты, предназначенные для выполнения логических операций

д) объекты, предназначенные для выполнения статистических операций

15. Файловая система необходима...

а) для управления аппаратными средствами

б) для тестирования аппаратных средств

в) для организации структуры хранения

г) для организации структуры аппаратных средств

16. Шины персонального компьютера обеспечивают...

а) соединение между собой его элементов и устройств

б) устранение излучения сигналов

в) устранение теплового излучения

г) применение общего источника питания

17. Постоянно запоминающее устройство (ПЗУ) является ... памятью

а) энергонезависимой

б) энергозависимой

в) динамической

г) оперативной с произвольным доступом

7.2. Методические материалы, определяющие процедуру оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

а) Критерии оценивания компетенций (результатов).

Программой дисциплины в целях проверки прочности усвоения материала предусматривается проведение различных форм контроля:

1. Текущий контроль - это проверка полноты знаний по основному материалу дисциплинарного модуля (ДМ).

2. Промежуточный контроль - итоговая проверка уровня знаний студента по данной дисциплине в конце семестра (в форме устного или письменного экзамена, сетевого компьютерного тестирования.) Промежуточной формой контроля знаний, умений и навыков по дисциплине является экзамен.

Общий результат выводится как интегральная оценка, складывающаяся из текущего контроля - 50% и промежуточного контроля - 50%.

Текущий контроль по дисциплине включает:

- посещение занятий - 10 баллов,

- участие на практических занятиях (устный опрос, решение задач) - 25 баллов,

- выполнение лабораторных заданий - 25 баллов,

- выполнение домашних контрольных работ (самостоятельная работа) - 10 баллов.

Текущий контроль по ДМ:

письменная контрольная работа - 15 баллов;

тестирование - 15 баллов;

Промежуточный контроль по дисциплине включает:

- устный экзамен (тестирование) - 30 баллов,

Критерии оценки посещения занятий- оценка выставляется по 100 бальной системе и соответствует проценту занятий, которые посетил студент из всего количества аудиторных занятий предусмотренных ДМ.

Критерии оценки участия на практических занятиях

Устный опрос. Развернутый ответ студента должен представлять собой связное, логически последовательное сообщение на заданную тему, показывать его умение применять определения, правила в конкретных случаях.

Показатели оценивания:

- 1) полнота и правильность ответа;
- 2) степень осознанности, понимания изученного;
- 3) языковое оформление ответа.

Критерии оценивания устного опроса:

86-100 баллов ставится, если студент полно излагает материал (отвечает на вопрос), дает правильное определение основных понятий; обнаруживает понимание материала, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только из учебника, но и самостоятельно составленные; излагает материал последовательно и правильно с точки зрения норм литературного языка.

66-85 баллов ставится, если студент дает ответ, удовлетворяющий тем же требованиям, что и для 86-100 баллов, но допускает 1-2 ошибки, которые сам же исправляет, и 1-2 недочета в последовательности и языковом оформлении излагаемого.

51-65 балл ставится, если студент обнаруживает знание и понимание основных положений данной темы, но излагает материал неполно и допускает неточности в определении понятий или формулировке правил; не умеет достаточно глубоко и доказательно обосновать свои суждения и привести свои примеры; излагает материал непоследовательно и допускает ошибки в языковом оформлении излагаемого.

0-50 баллов ставится, если студент обнаруживает незнание большей части соответствующего вопроса, допускает ошибки в формулировке определений и правил, искажающие их смысл, беспорядочно и неуверенно излагает материал.

Решение задач.

86-100 баллов выставляется, если студент демонстрирует знания теоретического и практического материала по теме практической работы, определяет взаимосвязи между показателями задачи, даёт правильный алгоритм решения, определяет междисциплинарные связи по условию задания.

66-85 баллов выставляется, если студент демонстрирует знания теоретического и практического материала по теме практической работы, допуская незначительные неточности при решении задач, имея неполное понимание междисциплинарных связей при правильном выборе алгоритма решения задания.

51-65 балл выставляется, если студент затрудняется с правильной оценкой предложенной задачи, даёт неполный ответ, требующий наводящих вопросов преподавателя, выбор алгоритма решения задачи возможен при наводящих вопросах преподавателя.

0-50 баллов выставляется студенту, если он даёт неверную оценку ситуации, неправильно выбирает алгоритм решения.

Критерии оценки выполнения лабораторных заданий.

Основными показателями оценки выполненной студентом и представленной для проверки работы являются:

1. Степень соответствия выполненного задания поставленным целям, задачам и требованиям;
2. Оформление, структурирование и комментирование лабораторной работы;
3. Уникальность выполнения работы (отличие от работ коллег);
4. Успешные ответы на контрольные вопросы.

Критерии оценки лабораторной работы.

86-100 баллов - оформление соответствует требованиям, критерии выдержаны, защита всего перечня контрольных вопросов.

66-85 баллов - оформление соответствует требованиям, критерии выдержаны, защита только до 85 % контрольных вопросов.

51-65 балл - оформление соответствует требованиям, критерии выдержаны, защита только до 51 % контрольных вопросов.

0-50 баллов - оформление не соответствует требованиям, критерии не выдержаны, защита только менее 51 % контрольных вопросов.

Критерии оценки выполнения домашних контрольных работ (самостоятельная работа).

Основными показателями оценки выполненной студентом и представленной для проверки домашней контрольной работы являются:

1. Степень соответствия выполненного задания поставленным целям, задачам и требованиям;

2. Оформление, структурирование и комментирование лабораторной работы;

3. Уникальность выполнения работы (отличие от работ коллег);

4. Успешные ответы на контрольные вопросы.

Критерии оценки домашней контрольной работы.

86-100 баллов - студент правильно выполнил индивидуальное самостоятельное задание. Показал отличное владение навыками применения полученных знаний и умений при решении профессиональных задач в рамках усвоенного учебного материала. Ответил на все дополнительные вопросы на защите.

66-85 баллов - студент выполнил индивидуальное самостоятельное задание с небольшими неточностями. Показал хорошие владения навыками применения полученных знаний и умений при решении профессиональных задач в рамках усвоенного учебного материала. Ответил на большинство дополнительных вопросов на защите.

51-65 балл - студент выполнил индивидуальное самостоятельное задание с существенными неточностями. Показал удовлетворительное владение навыками применения полученных знаний и умений при решении профессиональных задач в рамках усвоенного учебного материала. При ответах на дополнительные вопросы на защите было допущено много неточностей.

0-50 баллов - при выполнении индивидуального самостоятельного задания студент продемонстрировал недостаточный уровень владения умениями и навыками при решении профессиональных задач в рамках усвоенного учебного материала. При ответах на дополнительные вопросы на защите было допущено множество неточностей.

Критерии оценки текущего контроля по ДМ (письменная контрольная работа и тестирование).

Письменная контрольная работа состоит из двух типов вопросов:

1. Теоретические вопросы из курса лекций и практических работ. - 40 баллов.

2. Практические вопросы и задачи по лекционному и практическому материалу. - 60 баллов.

86-100 баллов - студент, показал всестороннее, систематическое и глубокое знание учебного материала, самостоятельно ответил на вопросы, ответ отличается богатством и точностью использованных терминов, материал излагается последовательно и логично; показал отличное владение навыками применения полученных знаний и умений при решении профессиональных задач.

66-85 баллов - студент, показал полное знание учебного материала, не допускающий в ответе существенных неточностей, самостоятельно выполнивший ответивший на вопросы; показал хорошие владения навыками применения полученных знаний и умений при решении профессиональных задач

51-65 балл - студент, обнаруживший знание основного учебного материала в

объёме, необходимом для дальнейшей учебы самостоятельно выполнивший задания, однако допустивший некоторые погрешности при ответе на вопросы; показал удовлетворительное владение навыками применения полученных знаний и умений при решении профессиональных задач.

0-50 баллов - выставляется студенту, обнаружившему пробелы в знаниях или отсутствие знаний по значительной части основного учебного материала, не выполнившего задания, допустившему принципиальные ошибки при ответе на вопросы, продемонстрировавший недостаточный уровень владения умениями и навыками при решении профессиональных задач.

Критерии выставления оценок за *тестирование* Тестовое задание состоит из пятнадцати вопросов. Время выполнения работы: 15-20 мин.

86-100 баллов - оценка «отлично» - 13-15 правильных ответов;

66-85 баллов - оценка «хорошо» - 10-12 правильных ответов;

51-65 балл - оценка «удовлетворительно» - 8-9 правильных ответов;

0-50 баллов - оценка «неудовлетворительно» - менее 8 правильных ответов.

Критерии оценки устного экзамена

Экзаменационные билеты включают три типа заданий:

1. Теоретические вопросы из курса лекций и практических работ. - 30 баллов.

2. Практические вопросы по лекционному и практическому материалу. - 40 баллов.

3. Проблемные вопросы и расчетные задачи. - 40 баллов.

Проверка качества подготовки студентов на экзаменах заканчивается выставлением отметок по принятой пятибалльной шкале («отлично», «хорошо», «удовлетворительно», «неудовлетворительно»).

86-100 баллов - оценка «отлично» - студент владеет знаниями по дисциплине «Информационные технологии и программирование» в полном объеме учебной программы, достаточно глубоко осмысливает дисциплину; самостоятельно, в логической последовательности и исчерпывающе отвечает на все вопросы билета, подчеркивая при этом самое существенное, умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделять в нем главное; устанавливать причинно-следственные связи; четко формирует ответы, свободно решает ситуационные задачи повышенной сложности; хорошо знаком с основной литературой; увязывает теоретические аспекты дисциплины с прикладными задачами исследования операций и методов оптимизации; владеет современными информационными технологиями решения прикладных задач.

66-85 баллов - оценка «хорошо» - студент владеет знаниями дисциплины «Информационные технологии и программирование» почти в полном объеме программы (имеются пробелы знаний только в некоторых, особенно сложных разделах); самостоятельно и отчасти при наводящих вопросах дает полноценные ответы на вопросы билета; не всегда выделяет наиболее существенное, не допускает вместе с тем серьезных ошибок в ответах; умеет решать легкие и средней тяжести ситуационные задачи; умеет трактовать выбор тех или иных методов и средств решения прикладных задач.

51-65 балл - оценка «удовлетворительно» - студент владеет основным объемом знаний по дисциплине «Информационные технологии и программирование»; проявляет затруднения в самостоятельных ответах, оперирует неточными формулировками; в процессе ответов допускаются ошибки по существу вопросов. Студент способен решать лишь наиболее легкие задачи, владеет только обязательным минимумом при решении задач исследования операций.

0-50 баллов - оценка «неудовлетворительно» - студент не освоил обязательного минимума знаний дисциплины «Информационные технологии и программирование», не способен ответить на вопросы билета даже при дополнительных наводящих вопросах экзаменатора.

Критерии оценки экзамена в форме тестирования

Тестовое задание состоит из тридцати вопросов. Время выполнения работы: 60 мин.

86-100 баллов - оценка «отлично» - 26-30 правильных ответов;

66-85 баллов - оценка «хорошо» - 20-25 правильных ответов;

51-65 баллов - оценка «удовлетворительно» - 16-19 правильных ответов;

0-50 баллов - оценка «неудовлетворительно» - менее 16 правильных ответов.

8. Учебно-методическое обеспечение дисциплины.

а) основная литература:

1. Макарова Н.В. Информатика: учеб. для вузов: [для бакалавров] / Макарова, Наталья Владимировна, В. Б. Волков [и др.] . - СПб.: Питер, 2013, 2011. - 573 с. - (Учебник для вузов). - Рекомендовано УМО. - ISBN 978-5-496-00001-7: 441-00.(Количество экз. - 80)

2. Давыдов И.С. Информатика [Электронный ресурс]: учебное пособие /И.С. Давыдов. - Электрон.текстовые данные. - СПб.: Проспект Науки, 2016. - 480 с. - 978-5903090-19-8. - Режим доступа: <http://www.iprbookshop.ru/35850.html> (дата обращения: 07.09.2022)

3. Роганов Е.А. Основы информатики и программирования [Электронный ресурс] / Е.А. Роганов. - 2-е изд. - Электрон.текстовые данные. - М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. - 392 с. - 2227-8397. - Режим доступа: <http://www.iprbookshop.ru/73689.html> (дата обращения: 07.06.2022)

б) дополнительная литература:

4. Васильев А.Н. Python на примерах [Электронный ресурс]: практический курс по программированию/ Васильев А.Н.- Электрон. текстовые данные- Санкт-Петербург: Наука и Техника, 2017.- 432 с.- Режим доступа: <http://www.iprbookshop.ru/73043.html>.- ЭБС «IPRbooks»

5. Сузи Р.А. Язык программирования Python [Электронный ресурс]: учебное пособие/ Сузи Р.А.- Электрон. текстовые данные- Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020.- 350 с.- Режим доступа: <http://www.iprbookshop.ru/97589.html>.- ЭБС «IPRbooks»

6. 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> (дата обращения: 07.06.2022)

7. Никифоров С.Н. Информатика. Часть 3. Прикладное программирование [Электронный ресурс]: учебное пособие / С.Н. Никифоров. - Электрон.текстовые данные. - СПб.: Санкт-Петербургский государственный архитектурно-строительный университет, ЭБС АСВ, 2016. - 128 с. - 978-5-9227-0743-5. - Режим доступа: <http://www.iprbookshop.ru/74384.html> (дата обращения: 07.06.2022)

8. Прохорова О.В. Информатика [Электронный ресурс]: учебник / О.В. Прохорова. - Электрон.текстовые данные. - Самара: Самарский государственный архитектурно-строительный университет, ЭБС АСВ, 2013. - 106 с. - 978-5-9585-0539-5. - Режим доступа: <http://www.iprbookshop.ru/20465.html> (дата обращения: 07.06.2022)

9. Шелудько В.М. Основы программирования на языке высокого уровня Python [Электронный ресурс]: учебное пособие/ Шелудько В.М.- Электрон.текстовые данные- Ростов-на-Дону, Таганрог: Издательство Южного федерального университета, 2017.- 146 с.- Режим доступа: <http://www.iprbookshop.ru/87461.html>.- ЭБС «IPRbooks»

10. Яшин В.Н. Информатика: аппаратные средства персонального компьютера: учеб.пособие / Яшин, Владимир Николаевич. - М.: ИНФРА-М, 2010. - 252,[4] с. - (Высшее образование: серия основана в 1996 г.). - Допущено УМО. - ISBN 978-5-16-003190-3: 153-23. (Количество экз. - 10)

9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»,

необходимых для освоения дисциплины

1. Портал доступа к электронным образовательным ресурсам ДГУ [Электронный ресурс] Режим доступа URL: <http://dgu.ru> (дата обращения: 1.07.2022)
2. Электронно-библиотечная система IPRbooks [Электронный ресурс] Режим доступа URL: <http://www.iprbookshop.ru> (дата обращения: 1.07.2022)
3. Электронно-библиотечная система «Университетская библиотека онлайн» [Электронный ресурс] Режим доступа URL: <http://biblioclub.ru> (дата обращения: 1.09.2022)
4. Электронно-библиотечная система издательства «Инфра» [Электронный ресурс] Режим доступа URL: <http://znanium.com> (дата обращения: 1.09.2022)
5. IT-портал [Электронный ресурс] Режим доступа URL: <http://citforum.ru>
6. Портал Национального открытого университета «Интуит» [Электронный ресурс] Режим доступа URL: <http://www.intuit.ru> (дата обращения: 1.09.2022)
7. Техническая документация фирмы Microsoft [Электронный ресурс] Режим доступа URL: <http://technet.microsoft.com/ru-ru/sysinternals>(дата обращения: 1.07.2022)
8. Moodle[Электронный ресурс]: система виртуального обучением: [база данных] / Даг.гос. ун-т. - Махачкала, г. - Доступ из сети ДГУ или, после регистрации из сети ун-та, из любой точки, имеющей доступ в интернет. -Режим доступа URL: <http://moodle.dgu.ru/>(дата обращения: 1.09.2022).
9. Учебный курс по программированию «Учите Питон» [Электронный ресурс] - Режим доступа: <http://pythontutor.ru/> (дата обращения: 1.07.2022)
10. Платформа учебных курсов Stepik [Электронный ресурс] - Режим доступа URL: <https://stepik.org/catalog?q=Python> (дата обращения: 1.07.2022)

10. Методические указания для обучающихся по освоению дисциплины

Для изучения теоретического курса студентам необходимо использовать лекционный материал, учебники и учебные пособия из списка основной и дополнительной литературы, интернет источники.

По дисциплине «Информационные технологии и программирование» в конце каждого модуля проводится контрольная работа.

В контрольную работу включаются теоретические вопросы и задачи тех типов, которые были разобраны на предшествующих практических занятиях.

Рабочей программой дисциплины «Информационные технологии и программирование» предусмотрена самостоятельная работа студентов в объеме 144 часа. Самостоятельная работа проводится с целью углубления знаний по дисциплине и предусматривает:

- чтение студентами рекомендованной литературы и усвоение теоретического материала дисциплины;
- подготовку к практическим занятиям;
- выполнение индивидуальных заданий;
- подготовку к контрольным работам, экзамену.

С самого начала изучения дисциплины студент должен четко уяснить, что без систематической самостоятельной работы успех невозможен. Эта работа должна регулярно начинаться сразу после лекционных и практических занятий, для закрепления только что пройденного материала.

После усвоения теоретического материала можно приступить к самостоятельному решению задач из учебников и пособий, входящих в список основной литературы.

11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем.

Интернет-ресурсы, мультимедиа, электронная почта для коммуникации со

студентами.

Обучающие интерактивные интернет-порталы, интернет-ресурсы, мультимедиа, электронная почта для коммуникации со студентами, системы программирования, MicrosoftOffice.

Справочно-правовые системы

1. Компьютерные классы с доступом к сети интернет и с набором лицензионного базового программного обеспечения для проведения лабораторных занятий;

2. Python IDE (Microsoft Visual Studio, Microsoft Visual Studio Code, PyCharm), для выполнения лабораторных заданий

3. Лекционная мультимедийная аудитория для чтения лекций с использованием мультимедийных материалов.

12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Компьютерные классы и лаборатории, оборудованные для проведения лекционных, практических и лабораторных занятий средствами оргтехники, персональными компьютерами, объединенными в сеть с выходом в Интернет; установленное лицензионное и свободное программное обеспечение.

Для выполнения лабораторных работ используется компьютерное оборудование с установленными программными продуктами MSOffice, Интерпретатор Python, Notebook. Аудиторные занятия проводятся в компьютерных классах с доступом к сети Интернет.