

Документ подписан простой электронной подписью

Информация о владельце: **Министерство науки и высшего образования Российской Федерации**

ФИО: Кандрашина Елена **Федеральное государственное автономное образовательное учреждение**

Должность: И.о. ректора ФГАОУ ВО «Самарский государственный **экономический**

университет»

Дата подписания: 18.07.2024 14:31:41

Уникальный программный ключ:

2db64eb9605ce27edd3b8e8fdd32c70e0674ddd2

**«Самарский государственный экономический университет»**

~~Факультет среднего профессионального и предпрофессионального образования~~

**Кафедра** факультета среднего профессионального и предпрофессионального образования

**Утверждено**

Ученым советом университета  
(протокол №10 от 30 мая 2024г.)

## **РАБОЧАЯ ПРОГРАММА**

**Наименование дисциплины** ОП.04 Основы алгоритмизации и программирования

**Специальность** 09.02.07 Информационные системы и программирование

Квалификация (степень) выпускника специалист по информационным системам

## ***СОДЕРЖАНИЕ***

- 1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ**
- 2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ**
- 3. ОСОБЕННОСТИ РЕАЛИЗАЦИИ ДИСЦИПЛИНЫ В ОТНОШЕНИИ ЛИЦ ИЗ ЧИСЛА ИНВАЛИДОВ И ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ**
- 4. ФОРМЫ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ТЕМАМ УЧЕБНОЙ ДИСЦИПЛИНЫ**
- 5. ЗАДАНИЯ ДЛЯ ПРАКТИЧЕСКИХ ЗАНЯТИЙ**
- 6. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ**
- 7. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ**

# 1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ ОП.04 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

## 1.1. Место дисциплины в структуре основной образовательной программы:

Дисциплина ОП.04 Основы алгоритмизации и программирования является частью общепрофессионального цикла основной профессиональной образовательной программы в соответствии с ФГОС по специальности 09.02.07 «Информационные системы и программирование».

Дисциплина ОП.04 Основы алгоритмизации и программирования обеспечивает формирование общих и профессиональных компетенций по всем видам деятельности ФГОС по специальности 09.02.07 «Информационные системы и программирование». Особое значение дисциплина имеет при формировании и развитии следующих компетенция: ОК 01, ОК 02, ПК 2.4, ПК 2.5.

### Перечень общих компетенций

Код	Наименование общих компетенций
ОК 01	Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам
ОК 02	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности.

### Перечень профессиональных компетенций

Код	Наименование видов деятельности и профессиональных компетенций
ПК 2.4	Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.
ПК 2.5	Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.

## 1.1.2. В результате освоения профессионального модуля студент должен:

<b>уметь</b>	выполнять базовые операции над объектами: цепочками символов, числами, списками, деревьями; проверять свойства этих объектов; выполнять и строить простые алгоритмы; оперировать информационными объектами, используя графический интерфейс: открывать, именовать, сохранять объекты, архивировать и разархивировать информацию, пользоваться меню и окнами, справочной системой; предпринимать меры антивирусной безопасности; оценивать числовые параметры информационных объектов и процессов: объем памяти, необходимый для хранения информации; скорость передачи информации; создавать информационные объекты; искать информацию с применением правил поиска (построения запросов) в базах данных, компьютерных сетях, некомпьютерных источниках информации (справочниках и словарях, каталогах, библиотеках) при выполнении заданий и
--------------	--

	<p>проектов по различным учебным дисциплинам; пользоваться персональным компьютером и его периферийным оборудованием (принтером, сканером, модемом, мультимедийным проектором, цифровой камерой, цифровым датчиком); следовать требованиям техники безопасности, гигиены, эргономики и ресурсосбережения при работе со средствами информационных и коммуникационных технологий</p>
<b>знать</b>	<p>виды информационных процессов; примеры источников и приемников информации; единицы измерения количества и скорости передачи информации; принцип дискретного (цифрового) представления информации; основные свойства алгоритма, типы алгоритмических конструкций: следование, ветвление, цикл; понятие вспомогательного алгоритма; программный принцип работы компьютера; назначение и функции используемых информационных и коммуникационных технологий</p>
<b>Иметь практический опыт</b>	<p>эксплуатации и модификации информационных систем; участие в разработке информационных систем</p>

## 2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

### 2.1. Объем учебной дисциплины и виды учебной работы

<b>Вид учебной работы</b>	<b>Объем часов</b>
<b>Объем образовательной программы учебной дисциплины</b>	<b>118</b>
в том числе:	
теоретическое обучение	<b>28</b>
практические занятия	<b>48</b>
курсовая работа (проект) <i>(не предусмотрено)</i>	
<i>Самостоятельная работа</i>	<b>36</b>
<b>Консультация</b>	<b>4</b>
<b>Промежуточная аттестация (экзамен)</b>	<b>6</b>

## 2.2. Тематический план и содержание учебной дисциплины

Наименование разделов и тем	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная учебная работа обучающихся	Объем в часах	Формируемые компетенции	
1	2	3	4	
<b>Раздел 1. Основы алгоритмизации и программирования</b>		<b>92</b>	ОК 01, ОК 02, ПК 2.4, ПК 2.5.	
<b>Тема 1.1. Устройство языка Python. Среда разработки IDLE. Сохранение Python-программ</b>	<b>Содержание</b>	<b>34</b>	ОК 01, ОК 02, ПК 2.4, ПК 2.5.	
	<b>Теоретическое обучение</b>	<b>14</b>		
	1. Основы программирования на языке Python. Переменные. Условия if, else. Операторы сравнения. Библиотека math.	2		
	2. Циклы.	2		
	3. Строки. Методы. Срезы.	2		
	4. Функции и рекурсия.	2		
	5. Списки. Словари. Кортежи.	2		
	6. Множества. Массивы.	2		
	7. Работа с файловой системой и файлами. Утилита pip: установка дополнительных библиотек. Анонимные функции. Функции генераторы.	2		
	<b>В том числе, практических занятий.</b>	<b>20</b>		ОК 01, ОК 02, ПК 2.4, ПК 2.5.
	<b>1. Практическое занятие «Задачи с использованием условных операторов»</b>	2		
	<b>2. Практическое занятие «Задачи с использованием циклов»</b>	2		
	<b>3. Практическое занятие «Задачи с использованием строковых операторов»</b>	2		
	<b>4. Практическое занятие «Задачи с использованием Функции и рекурсий»</b>	2		
<b>5. Практическое занятие «Задачи с использованием Списков. Словарей. Кортежей»</b>	4			
<b>6. Практическое занятие «Задачи с использованием Множеств и Массивов»</b>	2			
<b>7. Практическое занятие «Работа с файловой системой и файлами»</b>	4			
<b>8. Практическое занятие «Задачи с использованием анонимных функций. Функции генераторы»</b>	2			
<b>Тема 1.2 Введение в ООП</b>	<b>Содержание</b>	<b>42</b>	ОК 01, ОК 02, ПК 2.4,	
	<b>Теоретическое обучение</b>	<b>14</b>		

	1. Определение класса и создание экземпляра класса. Метод <code>init_()</code> , <code>del_()</code> .	2	ПК 2.5.
	2. Статистические методы и методы класса. Абстрактные методы	2	
	3. Обработка исключений. Инструкции <code>try,except,else</code> .	2	
	4. Основы SQLite. Создание Базы данных.	4	
	5. Библиотека <code>MySQLClient</code>	4	
	6. Библиотека <code>PyODBC</code>	2	
	<b>В том числе, практических занятий</b>	<b>28</b>	ОК 01, ОК 02, ПК 2.4, ПК 2.5.
	<b>1. Практическое занятие</b> «Выполнение запросов. Вложенные запросы».	6	
	<b>2. Практическое занятие</b> «Управление транзакциями».	6	
	<b>3. Практическое занятие</b> «Трассировка выполняемых запросов».	6	
	<b>4. Практическое занятие</b> «Библиотека <code>MySQLClient</code> ».	4	
	<b>5. Практическое занятие</b> «Библиотека <code>PyODBC</code> ».	6	
<b>Тематика самостоятельной учебной работы при изучении дисциплины</b>		<b>36</b>	ОК 01, ОК 02, ПК 2.4, ПК 2.5.
1. Работа с конспектами, учебной и специальной литературой;			
2. Доработка разрабатываемых проектов;			
3. Подготовка отчетов по практическим занятиям;			
4. Написание рефератов и докладов.			
<b>Курсовой проект (работа) не предусмотрен</b>			
<b>Самостоятельная учебная работа обучающегося над курсовым проектом (работой) (не предусмотрена)</b>			
<b>Консультация</b>		<b>4</b>	
<b>Промежуточная аттестация (экзамен)</b>		<b>6</b>	
<b>Всего</b>		<b>118</b>	

### **3. ОСОБЕННОСТИ РЕАЛИЗАЦИИ ДИСЦИПЛИНЫ В ОТНОШЕНИИ ЛИЦ ИЗ ЧИСЛА ИНВАЛИДОВ И ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ**

Студент с ограниченными возможностями здоровья, в отличие от остальных обучающихся, имеют свои специфические особенности восприятия, переработки материала.

Подбор и разработка учебных материалов должны производиться с учетом того, чтобы предоставлять этот материал в различных формах так, чтобы инвалиды с нарушениями слуха получали информацию визуально, с нарушениями зрения – аудиально (например, с использованием программ-синтезаторов речи) или с помощью тифлоинформационных устройств.

Выбор средств и методов обучения осуществляется самим преподавателем. При этом в образовательном процессе рекомендуется использование социально-активных и рефлексивных методов обучения, технологий социокультурной реабилитации с целью оказания помощи в установлении полноценных межличностных отношений с другими студентами, создании комфортного психологического климата в студенческой группе.

Согласно требованиям, установленным Минобрнауки России к порядку реализации образовательной деятельности в отношении инвалидов и лиц с ограниченными возможностями здоровья, необходимо иметь в виду, что:

1) инвалиды и лица с ограниченными возможностями здоровья по зрению имеют право присутствовать на занятиях вместе с ассистентом, оказывающим обучающемуся необходимую помощь.

2) инвалиды и лица с ограниченными возможностями здоровья по слуху имеют право на использование звукоусиливающей аппаратуры.

При проведении промежуточной аттестации по дисциплине обеспечивается соблюдение следующих общих требований:

- проведение аттестации для инвалидов в одной аудитории совместно с студентами, не являющимися инвалидами, если это не создает трудностей для инвалидов и иных обучающихся при прохождении государственной итоговой аттестации;

- присутствие в аудитории ассистента (ассистентов), оказывающего студенту инвалидам необходимую техническую помощь с учетом их индивидуальных особенностей (занять рабочее место, передвигаться, прочесть и оформить задание, общаться с экзаменатором);

- пользование необходимыми студенту инвалидам техническими средствами при прохождении аттестации с учетом их индивидуальных особенностей;

- обеспечение возможности беспрепятственного доступа обучающихся инвалидов в аудитории, туалетные и другие помещения, а также их пребывания в указанных помещениях.

По письменному заявлению обучающегося инвалида продолжительность сдачи студенту инвалидом экзамена может быть увеличена по отношению к установленной продолжительности его сдачи:

- продолжительность сдачи экзамена, проводимого в письменной форме, - не более чем на 90 минут;

- продолжительность подготовки обучающегося к ответу на экзамене, проводимом в устной форме, - не более чем на 20 минут;

В зависимости от индивидуальных особенностей обучающихся с ограниченными возможностями здоровья организация обеспечивает выполнение следующих требований при проведении аттестации:

а) для слепых:

- задания и иные материалы для сдачи экзамена оформляются рельефно-точечным шрифтом Брайля или в виде электронного документа, доступного с помощью компьютера со специализированным программным обеспечением для слепых, либо зачитываются ассистентом;

- письменные задания выполняются студентами на бумаге рельефно-точечным шрифтом Брайля или на компьютере со специализированным программным обеспечением для слепых, либо надиктовываются ассистенту;

- при необходимости студенту предоставляется комплект письменных принадлежностей и бумага для письма рельефно-точечным шрифтом Брайля, компьютер со специализированным программным обеспечением для слепых;

б) для слабовидящих:

- задания и иные материалы для сдачи экзамена оформляются увеличенным шрифтом;
- обеспечивается индивидуальное равномерное освещение не менее 300 люкс;
- при необходимости студенту предоставляется увеличивающее устройство, допускается использование увеличивающих устройств, имеющихся у обучающихся;

в) для глухих и слабослышащих, с тяжелыми нарушениями речи:

- обеспечивается наличие звукоусиливающей аппаратуры коллективного пользования, при необходимости студенту предоставляется звукоусиливающая аппаратура индивидуального пользования;

- по их желанию испытания проводятся в письменной форме;

г) для лиц с нарушениями опорно-двигательного аппарата (тяжелыми нарушениями двигательных функций верхних конечностей или отсутствием верхних конечностей):

- письменные задания выполняются студентами на компьютере со специализированным программным обеспечением или надиктовываются ассистенту;

- по их желанию испытания проводятся в устной форме.

О необходимости обеспечения специальных условий для проведения аттестации студент должен сообщить письменно не позднее, чем за 10 дней до начала аттестации. К заявлению прилагаются документы, подтверждающие наличие у обучающегося индивидуальных особенностей (при отсутствии указанных документов в организации).

#### 4. ФОРМЫ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ТЕМАМ УЧЕБНОЙ ДИСЦИПЛИНЫ

При планировании самостоятельной внеаудиторной работы студенту могут быть рекомендованы следующие виды заданий:

– для овладения знаниями: чтение текста (учебника, первоисточника, дополнительной литературы); составление плана текста; графическое изображение структуры текста; конспектирование текста; выписки из текста; работа со словарями и справочниками; ознакомление с нормативными документами; учебно-исследовательская работа; использование аудио- и видеозаписей, компьютерной техники и Интернета и др.;

– для закрепления и систематизации знаний: работа с конспектом лекций (обработка текста); повторная работа над учебным материалом (учебника, первоисточника, дополнительной литературы, аудио- и видеозаписей); составление плана и тезисов ответа; составление таблиц для систематизации учебного материала; изучение нормативных материалов; ответы на контрольные вопросы; аналитическая обработка текста (аннотирование, рецензирование, реферирование и др.); подготовка сообщений к выступлению на семинаре, конференции; подготовка докладов, докладов; составление библиографии, тематических кроссвордов; тестирование и др.;

– для формирования умений: решение задач и упражнений по образцу; решение вариантов задач и упражнений; выполнение чертежей, схем; выполнение расчётно-графических работ; решение ситуационных производственных (профессиональных) задач; подготовка к деловым играм; проектирование и моделирование разных видов и компонентов профессиональной деятельности; подготовка курсовых и дипломных работ (проектов); экспериментально-конструкторская работа; опытно-экспериментальная работа; упражнения на тренажёре; упражнения спортивно-оздоровительного характера; рефлексивный анализ профессиональных умений с использованием аудио- и видеотехники и др.

Наиболее распространенной формой самостоятельной работы является подготовка докладов.

Доклады и рефераты должны быть выполнены в соответствии с методическими указаниями, утвержденными Ученым советом Университета (протокол №9 от 28 марта 2018 г.).

##### 4.1. Вопросы для самостоятельной работы

Наименование разделов и тем Общепрофессиональный цикл (ОП) Самостоятельная учебная работа обучающихся	Формируемые компетенции
1	2
<b>ОП.4 Основы алгоритмизации и программирования</b>	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
1. Работа с конспектами, учебной и специальной литературой	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
2. Подготовка отчетов по практическим занятиям	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
3. Написание рефератов и докладов	ОК 01. ОК 02., ПК 2.4. ПК 2.5.

## **4.2. Примерная тематика докладов/рефератов**

### **Раздел 1. Основы алгоритмизации и программирования**

#### **ОП.04 Основы алгоритмизации и программирования**

##### **Тема 1.1. Устройство языка Python. Среда разработки IDLE. Сохранение Python-программ**

1. Обработка и анализ растровых материалов программирования Python
2. Язык программирования Python.
3. Программирование Perl, Python
4. Исследование языка Python
5. Сравнение Python и других языков по синтаксису языка
6. Unity или Blender совместно с языком программирования Java или Python

##### **Тема 1.2 Введение в ООП**

1. Алгоритмические языки.
2. Исполнители алгоритмов. Компьютер как формальный исполнитель алгоритмов (программ).
3. Линейная алгоритмическая конструкция. Команда присваивания. Примеры.
4. Алгоритмическая структура «ветвление». Команда ветвления. Примеры полного и неполного ветвления.
5. Алгоритмическая структура «цикл». Циклы со счетчиком и циклы по условию.
6. Технология решения задач с помощью компьютера (моделирование, формализация, алгоритмизация, программирование).
7. Передовые технологии и популярные средства разработки программного обеспечения.
8. Графический инструментарий.
9. Алгоритмы и методы обработки данных.
10. Объектно-ориентированное программирование.

## 5. ЗАДАНИЯ ДЛЯ ПРАКТИЧЕСКИХ ЗАНЯТИЙ

По дисциплине предусмотрены практические занятия с использованием активных и интерактивных форм проведения занятий (компьютерных симуляций) в сочетании с внеаудиторной работой для формирования и развития общих и профессиональных компетенций обучающихся.

Наименование разделов и тем Общепрофессионального цикла (ОП), / лабораторные работы и/или практические занятия	Формируемые компетенции
1	2
<b>Раздел 1. Основы алгоритмизации и программирования</b>	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
<b>ОП.04 Основы алгоритмизации и программирования</b>	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
<b>Тема 1.1. Устройство языка Python. Среда разработки IDLE. Сохранение Python-программ</b>	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
1. Основы программирования на языке Python. Переменные. Условия if, else. Операторы сравнения. Библиотека math.	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
2. Циклы.	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
3. Строки. Методы. Срезы.	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
4. Функции и рекурсия.	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
5. Списки. Словари. Кортежи.	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
6. Множества. Массивы.	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
7. Работа с файловой системой и файлами.	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
8. Утилита pip: установка дополнительных библиотек.	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
9. Анонимные функции. Функции генераторы.	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
<b>В том числе, практических занятий.</b>	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
<b>1. Практическое занятие «Задачи с использованием условных операторов»</b>	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
<b>2. Практическое занятие «Задачи с использованием циклов»</b>	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
<b>3. Практическое занятие «Задачи с использованием строковых операторов»</b>	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
<b>4. Практическое занятие «Задачи с использованием Функции и рекурсий»</b>	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
<b>5. Практическое занятие «Задачи с использованием Списков. Словарей. Кортежей»</b>	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
<b>6. Практическое занятие «Задачи с использованием Множеств и Массивов»</b>	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
<b>7. Практическое занятие «Работа с файловой системой и файлами»</b>	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
<b>8. Практическое занятие «Задачи с использованием анонимных функций. Функции генераторы»</b>	ОК 01. ОК 02., ПК 2.4. ПК 2.5.

<b>Тема 1.2 Введение в ООП</b>	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
1. Определение класса и создание экземпляра класса. Метод <code>init ()</code> , <code>del ()</code> .	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
2. Статистические методы и методы класса. Абстрактные методы	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
3. Обработка исключений. Инструкции <code>try,except,else</code> .	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
4. Основы SQLite. Создание Базы данных.	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
5. Библиотека MySQLClient	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
6. Библиотека PyODBC	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
<b>В том числе, практических занятий</b>	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
<b>1. Практическое занятие</b> «Выполнение запросов. Вложенные запросы».	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
<b>2. Практическое занятие</b> «Управление транзакциями».	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
<b>3. Практическое занятие</b> «Трассировка выполняемых запросов».	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
<b>4. Практическое занятие</b> «Библиотека MySQLClient».	ОК 01. ОК 02., ПК 2.4. ПК 2.5.
<b>5. Практическое занятие</b> «Библиотека PyODBC».	ОК 01. ОК 02., ПК 2.4. ПК 2.5.

## 6. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

**6.1. Для реализации программы дисциплины предусмотрены:** лаборатория компьютерных сетей, учебная аудитория для проведения занятий лекционного типа, занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенная набором демонстрационного оборудования и учебно-наглядными пособиями; учебная аудитория для текущего контроля и промежуточной аттестации, оснащенная набором демонстрационного оборудования и учебно-наглядными пособиями; библиотека, читальный зал с выходом в интернет; помещение для хранения и профилактического обслуживания учебного оборудования; актовый зал; помещение для самостоятельной работы, оснащенные в соответствии с ОПОП по специальности 09.02.07 Информационные системы и программирование).

### 6.2. Информационное обеспечение реализации программы

Для реализации программы библиотечный фонд Университет имеет электронные образовательные и информационные ресурсы, рекомендуемые для использования в образовательном процессе

#### 6.2.1. Электронные издания:

1. Трофимов, В. В. Основы алгоритмизации и программирования : учебник для среднего профессионального образования / В. В. Трофимов, Т. А. Павловская ; под редакцией В. В. Трофимова. — 4-е изд. — Москва : Издательство Юрайт, 2024. — 119 с. — (Профессиональное образование). — ISBN 978-5-534-17498-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/539994>

### 6.2.2. Электронные ресурсы

1. Научная электронная библиотека eLIBRARY.RU <https://elibrary.ru/>
2. Электронная библиотечная система Юрайт Издательство Юрайт <https://biblio-online.ru/>
3. Платформа «Библиокомплектатор» <http://www.bibliocomplectator.ru/>
4. Справочно-правовая система «Консультант Плюс» <http://konsultant.ru/>

### 6.2.3. Дополнительные источники

1. Казанский, А. А. Прикладное программирование на Excel 2019 : учебное пособие для среднего профессионального образования / А. А. Казанский. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 171 с. — (Профессиональное образование). — ISBN 978-5-534-12461-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/537736>

2. Подбельский, В. В. Программирование. Базовый курс C# : учебник для среднего профессионального образования / В. В. Подбельский. — Москва : Издательство Юрайт, 2024. — 369 с. — (Профессиональное образование). — ISBN 978-5-534-11467-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/542233>

### 6.3. Обязательное программное обеспечение

1. Astra Linux Special Edition «Смоленск», «Орел»; РедОС
2. МойОфис Стандартный 2, МойОфис Образование, Р7-Офис Профессиональный

## 7. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ ОП.04 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ

### 7.1 Паспорт фонда оценочных средств по дисциплине

Фонд оценочных средств предназначен для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины ОП. 04 Основы алгоритмизации и программирования по специальности 09.02.07 «Информационные системы и программирование».

Фонд оценочных средств разработан в соответствии с требованиями ФГОС СПО 09.02.07 «Информационные системы и программирование» и рабочей программой ОП.04 Основы алгоритмизации и программирования.

В результате освоения учебной дисциплины студент должен:  
- освоить общие компетенции:

Код	Наименование общих компетенций
ОК 01	Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам
ОК 02	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности.

Перечень профессиональных компетенций:

Код	Наименование видов деятельности и профессиональных компетенций
ПК 2.4	Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.

ПК 2.5	Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.
--------	--

- получить умения и знания:

<b>уметь</b>	- разрабатывать программы на языке программирования Python, создавая собственные классы, а также использовать классы и модули из библиотек этого языка; - создавать удобный интерфейс для использования созданных программных средств с помощью библиотеки TkInter или др.
<b>знать</b>	- синтаксис языка программирования Python; - основные принципы объектно-ориентированного программирования. - основные классы из библиотеки классов языка программирования Python для создания объектно-ориентированных приложений.
<b>иметь практический опыт</b>	эксплуатации и модификации информационных систем; участие в разработке информационных систем

## 7.2. Перечень контролирующих мероприятий для проведения текущего контроля и промежуточной аттестации

Перечень контролирующих мероприятий для проведения текущего контроля по дисциплине ОП.04 «Основы алгоритмизации и программирования» представлен в таблице 1.

Таблица 1

Номер семестра	Текущий контроль				
	Тестирование	Опрос	Практические задания	Реферат/ доклад	Формирование портфолио
3	+	+	+	+	
4	+	+	+	+	

Перечень контролирующих мероприятий для проведения промежуточной аттестации по дисциплине ОП.04 «Основы алгоритмизации и программирования» представлен в таблице 2.

Таблица 2

Номер семестра	Промежуточная аттестация			
	Курсовая работа	Промежуточное тестирование	Зачет	Экзамен
3			+	
4				+

### 7.3 Результаты освоения междисциплинарных курсов, подлежащие оцениванию

<i>Результаты обучения</i>	<i>Критерии оценки</i>	<i>Методы оценки</i>
<p><b>Уметь</b>  Анализировать проектную и техническую документацию.  Выполнять тестирование интеграции.  Организовывать постобработку данных.  Использовать приемы работы в системах контроля версий.  Оценивать размер минимального набора тестов.  Разрабатывать тестовые пакеты и тестовые сценарии.  Выполнять ручное и автоматизированное тестирование программного модуля.  Выявлять ошибки в системных компонентах на основе спецификаций.  Использовать выбранную систему контроля версий.  Использовать методы для получения кода с заданной функциональностью и степенью качества.  Анализировать проектную и техническую документацию.  Организовывать постобработку данных.  Приемы работы в системах контроля версий.  Выявлять ошибки в системных компонентах на основе спецификаций.</p>	<p>- разрабатывать программы на языке программирования Python, создавая собственные классы, а также использовать классы и модули из библиотек этого языка;  - создавать удобный интерфейс для использования созданных программных средств с помощью библиотеки TkInter или др.</p>	<p>Устный опрос, тестирование, практические задания, доклад, реферат</p>
<p><b>Знать</b>  - основные источники информации и ресурсы для решения задач и проблем в профессиональном и/или социальном контексте  Модели процесса разработки программного обеспечения.  Основные принципы процесса разработки программного обеспечения.  Основные подходы к интегрированию программных модулей.  Основы верификации и аттестации программного обеспечения.  Методы и способы идентификации сбоев и ошибок при интеграции приложений.  Методы и схемы обработки исключительных ситуаций.  Основные методы и виды тестирования программных продуктов.  Приемы работы с инструментальными средствами тестирования и отладки.  Стандарты качества программной документации.  Основы организации инспектирования и верификации.  Встроенные и основные специализированные инструменты анализа качества программных</p>	<p>- синтаксис языка программирования Python;  - основные принципы объектно-ориентированного программирования.  - основные классы из библиотеки классов языка программирования Python для создания объектно-ориентированных приложений.</p>	<p>Устный опрос, тестирование, практические задания, доклад, реферат</p>

продуктов. Методы организации работы в команде разработчиков. Модели процесса разработки программного обеспечения. Основные принципы процесса разработки программного обеспечения. Основные подходы к интегрированию программных модулей. Основы верификации и аттестации программного обеспечения. Стандарты качества программной документации. Основы организации инспектирования и верификации. Встроенные и основные специализированные инструменты анализа качества программных продуктов. Методы организации работы в команде разработчиков		
--	--	--

#### 7.4. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания.

**Текущий контроль** знаний представляет собой контроль освоения программного материала учебной дисциплины, с целью своевременной коррекции обучения, активизации самостоятельной работы и проверки уровня знаний и умений обучающихся, сформированности компетенций.

Наименования разделов ОП/ Общепрофессиональный цикл	Типы контрольных заданий		
1	2		
<b>Раздел 1. Основы алгоритмизации и программирования</b>	Вопросы к устному опросу	Тестирование, решение задач	доклад, реферат
<b>Тема 1.1. Устройство языка Python. Среда разработки IDLE. Сохранение Python-программ</b>	Вопросы к устному опросу	Тестирование, решение задач	доклад, реферат
<b>Тема 1.2 Введение в ООП</b>	Вопросы к устному опросу	Тестирование, решение задач	доклад, реферат

##### 7.4.1. Комплект оценочных средств для текущего контроля

Текущий контроль знаний представляет собой контроль освоения программного материала учебной дисциплины, с целью своевременной коррекции обучения, активизации самостоятельной работы и проверки уровня знаний и умений обучающихся, сформированности компетенций. Результаты текущего контроля заносятся в журналы учебных занятий.

Формы текущего контроля знаний:

- устный опрос;
- практические задания;
- тестирование;
- написание докладов/рефератов.

Проработка конспекта лекций и учебной литературы осуществляется студентами в течение всего семестра, после изучения новой темы.

Преподаватель контролирует знание студентом пройденного материала с помощью контрольных вопросов или тестирования.

### **Вопросы для текущего контроля знаний (устный опрос)**

***Формируемые компетенции – ОК 01. ОК 02. ПК 2.4. ПК 2.5.***

1. Типы данных.
2. Переменные.
3. Числовые типы данных.
4. Операции над числовыми типами данных.
5. Строки.
6. Строки `unicod`.
7. Вывод данных.
8. Ввод данных.
9. Форматированный ввод/вывод.
10. Списки.
11. Выражения в списках.
12. Оператор `del`.
13. Использование списков, как стеков.
14. Использование списков, как очередей.
15. Операции сравнения для списков.
16. Диапазоны.
17. Кортежи. Отличие кортежей от словарей
18. Словари.
19. Оператор `if`. Особенности операторов сравнения.
20. Операторы цикла. Оператор `for`. Оператор `while`. Завершение цикла.
21. Продолжение цикла. Оператор `pass`.
22. Определение функции.
23. Пространство имен функции.
24. Передача параметров. Ключи.
25. Передача в функцию переменного числа аргументов.
26. Элементы функционального программирования.
27. Использование лямбда функций.
28. Функции работы со структурами данных.
29. Функция `map()`. Примеры применения
30. Функция `filter()`. Примеры применения
31. Функция `reduce()`. Примеры применения
32. Документирование функций.
33. Создание модулей.
34. Указание кодировки.
35. Поиск модулей.

### **Примерная тематика докладов/рефератов**

***Формируемые компетенции - ОК 01. ОК 02. ПК 2.4. ПК 2.5.***

1. Обработка и анализ растровых материалов программирования Python
2. Язык программирования Python.
3. Программирование Perl, Python
4. Исследование языка Python
5. Сравнение Python и других языков по синтаксису языка
6. Unity или Blender совместно с языком программирования Java или Python
7. Алгоритмические языки.
8. Исполнители алгоритмов. Компьютер как формальный исполнитель алгоритмов (программ).
9. Линейная алгоритмическая конструкция. Команда присваивания. Примеры.
10. Алгоритмическая структура «ветвление». Команда ветвления. Примеры полного и неполного ветвления.
11. Алгоритмическая структура «цикл». Циклы со счетчиком и циклы по условию.
12. Технология решения задач с помощью компьютера (моделирование, формализация, алгоритмизация, программирование).
13. Передовые технологии и популярные средства разработки программного обеспечения.
14. Графический инструментарий.
15. Алгоритмы и методы обработки данных.
16. Объектно-ориентированное программирование.

**Примерный перечень практических занятий по дисциплине  
Формируемые компетенции - ОК 01. ОК 02. ПК 2.4. ПК 2.5.**

**1. Практическое занятие «Задачи с использованием условных операторов»**

**Решение задач**

**Задача 1**

Расставьте скобки в выражении

`a and b or not a and not b`

в соответствии с порядком вычисления выражения (приоритетом операций). Всего потребуется 5 пар скобок (внешние скобки входят в их число).

**Задача 2**

Выполните код в интерпретаторе Python 3 и вставьте в поле ответа результат вычисления. Постарайтесь разобраться, почему интерпретатор выдал именно такой ответ. Помните, что любые арифметические операции выше по приоритету операций сравнения и логических операторов.

Помните, что в Python логические значения пишутся с большой буквы: `True`, `False`.

```
x = 5
```

```
y = 10
```

```
y > x * x or y >= 2 * x and x < y
```

**Задача 3**

Найдите результат выражения для заданных значений `aa` и `bb`. Учитывайте регистр символов при ответе.

a = True

b = False

a and b or not a and not b

## 2. Практическое занятие «Задачи с использованием циклов»

### Решение задач

#### Задача 1

Какое значение будет у переменной i после выполнения фрагмента программы?

```
i = 0
while i <= 10:
    i = i + 1
    if i > 7:
        i = i + 2
        Задача 2
```

Сколько итераций цикла будет выполнено в этом фрагменте программы?

```
i = 0
while i <= 10:
    i = i + 1
    if i > 7:
        i = i + 2
        Задача 3
```

Сколько всего знаков \* будет выведено после исполнения фрагмента программы:

```
i = 0
while i < 5:
    print('*')
    if i % 2 == 0:
        print('*')
    if i > 2:
        print('***')
    i = i + 1
```

## 3. Практическое занятие «Задачи с использованием строковых операторов»

### Решение задач

#### Задача 1

Укажите результат выражения:

"123" + "42"

Попробуйте получить ответ без использования интерпретатора.

Заметьте, что сложение строк производится одинаково и не зависит от того, какие данные содержит строка.

## Задача 2

напишите программу, вычисляющую площадь треугольника по переданным длинам трёх его сторон по формуле Герона:

$$S = \sqrt{p(p-a)(p-b)(p-c)}$$

$$p = \frac{a + b + c}{2}$$

где

– полупериметр треугольника. На вход программе подаются целые числа, выводом программы должно являться вещественное число, соответствующее площади треугольника.

Sample Input:

3  
4  
5

Sample Output:

6.0

## 4. Практическое занятие «Задачи с использованием Функции и рекурсий»

### Решение задач

#### Задача 1

Напишите программу, которая считывает с консоли числа (по одному в строке) до тех пор, пока сумма введённых чисел не будет равна 0 и сразу после этого выводит сумму квадратов всех считанных чисел.

Гарантируется, что в какой-то момент сумма введённых чисел окажется равной 0, после этого считывание продолжать не нужно.

В примере мы считываем числа 1, -3, 5, -6, -10, 13; в этот момент замечаем, что сумма этих чисел равна нулю и выводим сумму их квадратов, не обращая внимания на то, что остались ещё не прочитанные значения.

Sample Input:

1  
-3  
5  
-6  
-10  
13  
4  
-8

Sample Output:

340

#### Задача 2

Напишите программу, которая выводит часть последовательности 1 2 2 3 3 3 4 4 4 4 5 5 5 5 5

... (число повторяется столько раз, чему равно). На вход программе передаётся неотрицательное целое число  $n$  — столько элементов последовательности должна отобразить программа. На выходе ожидается последовательность чисел, записанных через пробел в одну строку.

Например, если  $n = 7$ , то программа должна вывести 1 2 2 3 3 3 4.

### Задача 3

Напишите программу, которая считывает список чисел `lstlst` из первой строки и число `xx` из второй строки, которая выводит все позиции, на которых встречается число `xx` в переданном списке `lstlst`.

Позиции нумеруются с нуля, если число `xx` не встречается в списке, вывести строку "Отсутствует" (без кавычек, с большой буквы).

Позиции должны быть выведены в одну строку, по возрастанию абсолютного значения.

## 4. Практическое занятие «Задачи с использованием Списков. Словарей. Кортежей»

### Решение задач

#### Задача 1

Сколько элементов будет содержать список `students` после следующих операций?

```
students = ['Ivan', 'Masha', 'Sasha']
students += ['Olga']
students += 'Olga'
```

Введите в поле ответа одно число.

#### Задача 2

Имеется программа, код которой указан ниже.

Укажите, какие значения будут содержать списки в помеченных участках:

```
a = [1, 2, 3]
b = a
# значения списка b?
```

```
a[1] = 10
# значения списка b?
```

```
b[0] = 20
# значения списка a?
```

```
a = [5, 6]
# значения списка b?
```

Запишите значения списков в одну строку, разделяя списки точкой с запятой, а элементы внутри списка — пробелом, например:

```
1 1 1; 2 2 2; 3 3 3; 4 4 4
```

## 6. Практическое занятие «Задачи с использованием Множеств и Массивов»

### Решение задач

### Задача 1

Напишите программу, на вход которой подаётся прямоугольная матрица в виде последовательности строк. После последней строки матрицы идёт строка, содержащая только строку "end" (без кавычек, см. Sample Input).

Программа должна вывести матрицу того же размера, у которой каждый элемент в позиции  $i$ ,  $j$  равен сумме элементов первой матрицы на позициях  $(i-1, j)$ ,  $(i+1, j)$ ,  $(i, j-1)$ ,  $(i, j+1)$ . У крайних символов соседний элемент находится с противоположной стороны матрицы.

В случае одной строки/столбца элемент сам себе является соседом по соответствующему направлению.

Sample Input 1:

```
9 5 3
0 7 -1
-5 2 9
end
```

Sample Output 1:

```
3 21 22
10 6 19
20 16 -1
```

Sample Input 2:

```
1
end
```

Sample Output 2:

```
4
```

## 7. Практическое занятие «Работа с файловой системой и файлами»

### Решение задач

#### Задача 1

Напишите программу, которая считывает из файла строку, соответствующую тексту, сжатою с помощью кодирования повторов, и производит обратную операцию, получая исходный текст.

Запишите полученный текст в файл и прикрепите его, как ответ на это задание.

В исходном тексте не встречаются цифры, так что код однозначно интерпретируем.

## 8. Практическое занятие «Задачи с использованием анонимных функций. Функции генераторы»

### Решение задач

#### Задача 1

Напишите функцию  $f(x)$ , которая возвращает значение следующей функции, определённой на всей числовой прямой:

Требуется реализовать только функцию, решение не должно осуществлять операций ввода-вывода.

$$f(x) = \begin{cases} 1 - (x + 2)^2, & \text{при } x \leq -2 \\ -\frac{x}{2}, & \text{при } -2 < x \leq 2 \\ (x - 2)^2 + 1, & \text{при } 2 < x \end{cases}$$

Sample Input 1:

4.5

Sample Output 1:

7.25

Sample Input 2:

-4.5

Sample Output 2:

-5.25

Sample Input 3:

1

Sample Output 3:

-0.5

**Примерные тестовые задания по дисциплине  
Формируемые компетенции - ОК 01. ОК 02. ПК 2.4. ПК 2.5.**

1. Укажите вариант записи выражения, истинного при выполнении указанного условия и ложного в противном случае:

х принадлежит интервалу (2, 10) или (-2, 2)

(x<2) and (x<= 10) or abs (x) > 2

(x>2) or (x<=10) and abs (x) < 2

(x>2) or (x<= 10) and (x) < 2

(x>2) and (x< 10) or abs (x) < 2

Е) (x>2) and (x<=10) or (x) < 2

2. Объем памяти, требуемый для размещения записи, складывается из сумм значений полей записи

Специально для типа данных записи объем памяти не отводится

из объема памяти, занимаемых полями записи

из объема памяти, отводимой переменной, определяемой от записи

из объема памяти, занимаемых полями записи

3 Выберите на какие группы делятся методы:  
Статистические, Динамические, Наследовательные  
Объектные, Перегружаемые. Классические  
Виртуальные, Динамические, Статистические  
Виртуальные, Перегружаемые, Статические  
Динамические. Перегружаемые. Статистические

4. Выберите стандартную функцию, которая используется для вычисления квадратного корня числа X.  
trunc(X)  
sqrt(X)  
sqr(X)  
exp(X)  
inc(X)

5 Укажите неправильно записанный оператор среди перечисленных вариантов.  
if a<b then a:=a\*a else b:=b\*b;  
if(a<b) or c then c:=false;  
if x and y then s:=s+1; else s:=s-1;  
if b then a:=d else a:=c;  
if k <>m then k:=m;

6. При определении массива указывается его  
имя, размерность и тип элементов  
размер и значение элемента  
размер и назначение элементов  
размер и свойство элемента  
размер и вид массива

7. Если множество  $M1 \supseteq M2$ , то это означает  
значения элементов множества  $M2$  больше значений элементов  $M1$ .  
все элементы множества  $M1$  содержатся в  $M2$   
значения элементов множества  $M1$  больше значений элементов  $M2$   
количество элементов множества  $M1$  больше количества элементов множества  $M2$   
все элементы множества  $M2$  содержатся в  $M1$

8. Количество элементов множества - это  
набор множества  
мощность множества;  
диапазон множества;  
размер множества  
объект множества;

9. Возможность изменять обработчики событий в любой момент выполнения программы называется  
делегирование  
агрегирование  
наследование  
автоматизирование  
генерирование

10. Формальные параметры подпрограммы делятся на:  
параметры переменные и параметры значения и параметры константы  
фактические параметры  
локальные параметры  
глобальные и локальные параметры  
фактические и локальные параметры

11. Файл - это ...  
программа, которая служит для подключения устройств ввода/вывода  
поименованная область на диске  
программа, переводящая текст на языке программирования в машинный код  
специальное место на диске, в котором хранятся характеристики файлов  
слово, имеющее до 8-ми символов до точки и 3-х после точки

12. SELECT COUNT(yearpub) FROM titles - функция возвращает:  
минимальное из всех значений данного поля  
число записей  
максимальное из всех значений данного поля  
сумму всех значений данного поля  
среднее по всем значениям данного поля

13. Какое предложение языка SQL позволяющее вносить изменения в записи таблицы  
Append  
Update  
Select  
Delete  
Insert

14. Добавить новую запись в таблицу  
INSERT INTO  
DROP INDEX  
DELETE FROM  
UPDATE  
CREATE DATABASE

15. Массивы бывают  
Одномерные и многомерные  
Строчные и числовые  
Одномерные и многомерные  
Числовые и вещественные  
Символьные или файловые

16. Основные операции с массивами - это  
Ввод и вывод значений в элементы массива, поиск максимального или минимального элемента,  
нахождение суммы элементов массива  
Подсчет индекса, количества элементов  
Определение типа элемента массива, сравнение элементов массива,  
Суммирование индексов, ввод значений в элементы массива

Перебор элементов массива, нахождение номера индекса,

17. Выберите оператор цикла с предусловием:

while ... do

case ... of

repeat... until

for... do

if... then

18. Какое утверждение неверно

Переменная может входить в состав логического выражения

Переменные и константы должны быть описаны перед первым использованием

Выражение может стоять в виде отдельного оператора в программе

Константа может входить в состав арифметического выражения

Выражение может стоять справа от знака присвоения

### Критерии и шкала оценивания (устный опрос)

Оценка			
«отлично»	«хорошо»	«удовлетворительно»	«неудовлетворительно»
Тема раскрыта в полном объеме, высказывания связанные и логичные, использована научная лексика, приведены примеры. Ответы даны в полном объеме.	Тема раскрыта не в полном объеме, высказывания в основном связанные и логичные, использована научная лексика, приведены примеры. Ответы на вопросы даны не в полном объеме.	Тема раскрыта недостаточно, высказывания несвязанные и нелогичные. Научная лексика не использована, не приведены примеры. Ответы на вопросы зависят от помощи со стороны преподавателя.	Тема не раскрыта. Логика изложения, примеры, выводы и ответы на вопросы отсутствуют.

### Критерии и шкала оценивания (выполнение практических заданий)

Оценка			
«отлично»	«хорошо»	«удовлетворительно»	«неудовлетворительно»
По решению задачи дан правильный ответ и развернутый вывод	По решению задачи дан правильный ответ, но не сделан вывод	По решению задачи дан частичный ответ, не сделан вывод	Задача не решена полностью

### Критерии и шкала оценивания (тестирование)

Число правильных ответов	Оценка
90-100% правильных ответов	Оценка «отлично»
70-89% правильных ответов	Оценка «хорошо»
51-69% правильных ответов	Оценка «удовлетворительно»
Менее 51 % правильных ответов	Оценка «неудовлетворительно»

### Критерии и шкала оценивания (доклады/рефераты)

Оценка	Критерии оценки доклада/реферата
<b>«отлично»</b>	<ol style="list-style-type: none"> <li>1. Соблюдение формальных требований к реферату</li> <li>2. Грамотное и полное раскрытие темы;</li> <li>3. Самостоятельность в работе над рефератом (использование рефератов из сети Интернет запрещается).</li> <li>4. Умение работать с учебной, профессиональной литературой.</li> <li>5. Умение работать с периодической литературой.</li> <li>6. Умение обобщать, делать выводы.</li> <li>7. Умение оформлять библиографический список к реферату в соответствии с требованиями ГОСТ Р 7.1.- 2003 «Библиографическая запись. Библиографическое описание. Общие требования и правила составления».</li> <li>8. Соблюдение требований к оформлению реферата.</li> <li>9. Умение кратко изложить основные положения реферата при его защите.</li> <li>10. Иллюстрация защиты реферата презентацией.</li> </ol>
<b>«хорошо»</b>	<ol style="list-style-type: none"> <li>1. Соблюдение формальных требований к реферату</li> <li>2. Грамотное и полное раскрытие темы;</li> <li>3. Самостоятельность в работе над рефератом (использование рефератов из сети Интернет запрещается).</li> <li>4. Умение работать с учебной, профессиональной литературой.</li> <li>5. Умение работать с периодической литературой.</li> <li>6. Не полно обобщен и сделан вывод.</li> <li>7. Не точно оформлен библиографический список к реферату в соответствии с требованиями ГОСТ Р 7.1.- 2003 «Библиографическая запись. Библиографическое описание. Общие требования и правила составления».</li> <li>8. Не полно соблюдены требования к оформлению реферата.</li> <li>9. Не четко сформированы краткие основные положения реферата при его защите.</li> <li>10. Иллюстрация защиты реферата презентацией.</li> </ol>
<b>«удовлетворительно»</b>	<ol style="list-style-type: none"> <li>1. Соблюдение формальных требований к реферату</li> <li>2. Грамотное и полное раскрытие темы;</li> <li>3. Самостоятельность в работе над рефератом (использование рефератов из сети Интернет запрещается).</li> <li>4. Не полно изучены учебная, профессиональная литература.</li> </ol>

	<p>5. Не полно изучена периодическая литература.</p> <p>6. Не обобщены и не конкретизированы выводы.</p> <p>7. Не точно оформлен библиографический список к реферату в соответствии с требованиями ГОСТ Р 7.1.- 2003 «Библиографическая запись. Библиографическое описание. Общие требования и правила составления».</p> <p>8. Не соблюдены требования к оформлению реферата.</p> <p>9. Не четко сформированы краткие основные положения реферата при его защите.</p> <p>10. Иллюстрация защиты реферата презентацией отсутствует</p>
<b>«неудовлетворительно»</b>	Реферат не представлен по соответствующим критериям оценивания

## 7.4.2. КОМПЛЕКТ ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

Примерные вопросы для подготовки к дифференцированному зачету

*Формируемые компетенции - ОК 01. ОК 02. ПК 2.4. ПК 2.5.*

Зачет позволяет оценить степень сформированности компетенций ОК 01. ОК 02. ОК 04. ОК 05. ОК 09. ПК 2.4. ПК 2.5.

1. Типы данных.
2. Переменные.
3. Числовые типы данных.
4. Операции над числовыми типами данных.
5. Строки.
6. Строки unisod.
7. Вывод данных.
8. Ввод данных.
9. Форматированный ввод/вывод.
10. Списки.
11. Выражения в списках.
12. Оператор del.
13. Использование списков, как стеков.
14. Использование списков, как очередей.
15. Операции сравнения для списков.
16. Диапазоны.
17. Кортежи. Отличие кортежей от словарей
18. Словари.
19. Оператор if. Особенности операторов сравнения.
20. Операторы цикла. Оператор for. Оператор while. Завершение цикла.
21. Продолжение цикла. Оператор pass.
22. Определение функции.
23. Пространство имен функции.
24. Передача параметров. Ключи.
25. Передача в функцию переменного числа аргументов.
26. Элементы функционального программирования.
27. Использование лямбда функций.
28. Функции работы со структурами данных.
29. Функция map(). Примеры применения
30. Функция filter(). Примеры применения
31. Функция reduce(). Примеры применения

32. Документирование функций.
33. Создание модулей.
34. Указание кодировки.
35. Поиск модулей.

**Примерные вопросы для подготовки к экзамену  
Формируемые компетенции - ОК 01. ОК 02. ПК 2.4. ПК 2.5.**

1. Компиляция модулей на Python.
2. Стандартные модули Python.
3. Использование функции `__dir()`.
4. Структурирование модулей в пакеты.
5. Импорт модулей и их составляющих из пакета.
6. Ссылки в пакетах.
7. Пакеты и файловая система.
8. Класс File.
9. Открытие файла.
10. Методы класса для File ввода-вывода.
11. Взаимодействие с файловой системой.
12. Модуль path.
13. Объекты и файловый ввод-вывод.
14. Объявление класса
15. Управление атрибутами и методами класса
16. Объявление объектов
17. Множественное наследование
18. Заимствование свойств и методов у родительского класса

**Критерии и шкалы оценивания промежуточной аттестации**

**Шкала и критерии оценки (дифференцированный зачет)**

Отлично	Хорошо	Удовлетворительно	Неудовлетворительно
<p>1. Материал изложен грамотно, в определенной логической последовательности, правильно используется терминология;</p> <p>2. Показано умение иллюстрировать теоретические положения конкретными примерами, применять их в новой ситуации;</p> <p>3. Продемонстрировано освоение ранее изученных отсутствующих вопросов, сформированность</p>	<p>1. Ответ удовлетворяет в основном требованиям на оценку «5», но при этом может иметь следующие недостатки:</p> <p>2. В изложении допущены небольшие пробелы, не искажившие содержание ответа;</p> <p>3. Допущены один - два недочета при освещении основного содержания ответа,</p>	<p>1. Неполно или непоследовательно раскрыто содержание материала, но показано общее понимание вопроса и продемонстрированы умения, достаточные для дальнейшего усвоения материала.</p> <p>2. Имелись затруднения или допущены ошибки в определении понятий, использовании терминологии, исправленные после нескольких наводящих вопросов;</p>	<p>1. Содержание материала не раскрыто.</p> <p>2. Ошибки в определении понятий, не использовалась терминология в ответе.</p>

умений и знаний; 4. Ответ прозвучал самостоятельно, без наводящих вопросов.	исправленные по замечанию преподавателя.	3. При неполном знании теоретического материала выявлена недостаточная сформированность умений и знаний.	

### Шкала и критерии оценки (экзамен)

Отлично	Хорошо	Удовлетворительно	Неудовлетворительно
<p>1. Материал изложен грамотно, в определенной логической последовательности, правильно используется терминология;</p> <p>2. Показано умение иллюстрировать теоретические положения конкретными примерами, применять их в новой ситуации;</p> <p>3. Продемонстрировано усвоение ранее изученных сопутствующих вопросов, сформированность умений и знаний;</p> <p>4. Ответ прозвучал самостоятельно, без наводящих вопросов.</p>	<p>1. Ответ удовлетворяет в основном требованиям на оценку «5», но при этом может иметь следующие недостатки:</p> <p>2. В изложении допущены небольшие пробелы, не искажившие содержание ответа;</p> <p>3. Допущены один - два недочета при освещении основного содержания ответа, исправленные по замечанию преподавателя.</p>	<p>1. Неполно или непоследовательно раскрыто содержание материала, но показано общее понимание вопроса и продемонстрированы умения, достаточные для дальнейшего усвоения материала.</p> <p>2. Имелись затруднения или допущены ошибки в определении понятий, использовании терминологии, исправленные после нескольких наводящих вопросов;</p> <p>3. При неполном знании теоретического материала выявлена недостаточная сформированность умений и знаний.</p>	<p>1. Содержание материала не раскрыто.</p> <p>2. Ошибки в определении понятий, не использовалась терминология в ответе.</p>