

МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ
И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ
Северо-Кавказский филиал
ордена Трудового Красного Знамени федерального государственного
бюджетного образовательного учреждения высшего образования
«Московский технический университет связи и информатики»

УТВЕРЖДАЮ
Зам. директора по УР
Н.А. Андреева
«28» 04 2026 г.

Рабочая программа дисциплины

Б1.О.37 Основы программирования на Python

Кафедра
Направление подготовки
Профиль

Информатики и вычислительной техники
10.03.01 Информационная безопасность
Безопасность компьютерных систем (по отрасли или
в сфере профессиональной деятельности)

Формы обучения

очная

Распределение часов дисциплины по семестрам (для очной формы обучения (ОФО))

Вид учебной работы	ОФО	
	ЗЕ	часов/семестр
Общая трудоемкость дисциплины, в том числе (по семестрам, курсам):	3	108/4
Контактная работа, в том числе (по семестрам, курсам):		54/4
Лекции		18/4
Лабораторных работ		18/4
Практических занятий		18/4
Семинаров		
Самостоятельная работа		54/4
Контроль		
Число контрольных работ (по курсам)		
Число КР (по семестрам, курсам)		
Число КП (по семестрам, курсам)		
Число зачетов с разбивкой по семестрам		1/4
Число экзаменов с разбивкой по семестрам		

Программу составил:
ст. преподаватель кафедры ИВТ Осмоловская Н.С.

Рабочая программа дисциплины
«Основы программирования на Python»

Разработана в соответствии с ФГОС ВО:
направления подготовки 10.03.01 «Информационная безопасность», утвержденным приказом Министерства науки и высшего образования Российской Федерации от 17 ноября 2020 г. N 1427

Составлена на основании учебного плана
направления 10.03.01 Информационная безопасность профиль Безопасность компьютерных систем (по отрасли или в сфере профессиональной деятельности), одобренного Учёным советом СКФ МТУСИ, протокол № 8 от 31.03 2026г., и утвержденного директором СКФ МТУСИ 31.03 2026 г.

Рассмотрена и одобрена на заседании кафедры
Информатики и вычислительной техники

Протокол от «28» 04 2026 г. № 9.

Зав. кафедрой  Соколов С.В.

1. Цели изучения дисциплины

Целью освоения дисциплины «Основы программирования Python» является изучение конструкции языка программирования Python, развитие умений программирования на языке Python для решения профессиональных задач.

Задачи дисциплины состоят в следующем:

- изучение основных конструкций языка программирования Python;
- формирование умений самостоятельного создания и использования различных структур данных;
- автоматизация задач по сбору и обработке данных;
- развитие умений разработки эффективных алгоритмов и программ на основе языка программирования Python.

2. Планируемые результаты обучения

Изучение дисциплины направлено на формирование у обучающегося способности решать профессиональные задачи в соответствии с эксплуатационным видом деятельности.

Результатом освоения дисциплины являются сформированные у обучающегося компетенции:

ОПК-7: Способен использовать языки программирования и технологии разработки программных средств для решения задач профессиональной деятельности		
Знать:		
- теоретические основы алгоритмизации и программирования; - работу линейных, разветвляющихся и циклических алгоритмов; - синтаксис Python и типы данных; - работу с функциями и создание модулей объектно-ориентированного и событийно-ориентированного программирования.		
Уметь:		
- реализовывать работу линейных, разветвляющихся и циклических алгоритмов на Python; - преобразовывать типы данных; - создавать пользовательские функции; - реализовывать концепцию объектно-ориентированного и событийно-ориентированного программирования.		
Владеть:		
- основными понятиями, связанными с общим синтаксисом и созданием алгоритмов на языке Python; - навыками создания пользовательских функций, объектно-ориентированного программирования, событийно-ориентированного программирования на Python.		

3. Место дисциплины в структуре образовательной программы

Требования к предварительной подготовке обучающегося (предшествующие дисциплины, модули, темы):		
1	Б1.О.05	Введение в информационные технологии
2	Б1.О.14	Информационные технологии и программирование
Последующие дисциплины и практики, для которых освоение данной дисциплины необходимо:		
1	Б1.О.29	Методы и средства криптографической защиты информации
2	Б1.В.02	Комплексное обеспечение защиты информации объекта информатизации

4. Структура и содержание дисциплины

4.1 Очная форма обучения, 4 года

Код зан.	Тема и краткое содержание занятия	Вид зан.	Кол. час.	Компетенции	УМИО
1	2	3	4	5	6
Курс 2, Семестр 4 – 54 часа аудиторных занятий + 54 часа СРС = 108 часов					
Модуль 1. Теоретические основы алгоритмизации и программирования (28 + 28 = 56 часов)					
1.1	Лекция 1. Алгоритмы и их свойства. Алгоритм. Свойства алгоритма. Способы описания алгоритма. Назначение функциональных блоков. Основные этапы решения задач. Алфавит языка Python. Идентификаторы и общие правила их написания. Оператор присваивания. Типы данных. Функции приведения типов. Запись математических функций. Операции отношения.	Лек.	2	ОПК-7	Л1.1, Л1.2, Л2.1 Л2.2 Л2.3
1.2	Лабораторная работа 1. Введение в язык программирования Python.	ЛР	2	ОПК-7	Л1.1, Л3.1 Л3.2
1.3	Документирование кода в Python. PEP 257. Однострочные строки документации. Многострочные строки документации.	СРС	8	ОПК-7	Л1.1, Л2.2 Л2.3
1.4	Лекция 2. Введение в Python. Процесс создания проекта в Python. Методы ввода и вывода данных и обработка исключений. Контрольные вопросы.	Лек.	2	ОПК-7	Л1.1, Л1.2, Л2.1 Л2.2 Л2.3
1.5	Практическая работа 1. Перевод чисел между различными системами счисления	ПЗ	2	ОПК-7	Л1.1, Л3.2
1.6	Лабораторная работа 2. Математические операции в Python.	ЛР	4	ОПК-7	Л1.1, Л3.1
1.7	Числа: целые, вещественные, комплексные. Функции и методы строк	СРС	10	ОПК-7	Л1.1, Л2.3
1.8	Лекция 3. Линейный и разветвляющийся алгоритм. Разветвляющийся алгоритм. Простой условный оператор. Сокращенный условный оператор. Составной условный оператор. Многочисленные ветвления. Алгоритмы поиска максимального и минимального элементов. Примеры решения задач.	Лек.	2	ОПК-7	Л1.1, Л1.2, Л2.1 Л2.2 Л2.3
1.9	Практическая работа 2. Построение линейных алгоритмов в Python.	ПЗ	2	ОПК-7	Л1.1, Л3.2
1.11	Лабораторная работа 3. Структура ветвление в Python.	ЛР	2	ОПК-7	Л1.1, Л3.1
1.12	Лекция 4. Циклический алгоритм. Оператор цикла for. Контрольные вопросы. Задачи для самостоятельного решения. Оператор цикла while. Контрольные вопросы. Задачи для самостоятельного	Лек.	2	ОПК-7	Л1.1, Л1.2, Л2.1 Л2.2 Л2.3

	решения.				
1.13	Практическая работа 3. Работа с циклами в Python	ПЗ	2	ОПК-7	Л1.1, Л3.2
1.14	PEP 8 - руководство по написанию кода на Python. Внешний вид кода. Пробелы в выражениях и инструкциях. Комментарии. Соглашения по именованию. Общие рекомендации	СРС	10	ОПК-7	Л1.1, Л2.3
1.15	Лекция 5. Работа с кортежами, списками и строками. Объявление кортежей. Классические способы обработки кортежей. Работа со списками. Работа со словарями. Примеры решения задач. Контрольные вопросы. Задачи для самостоятельного решения. Основные понятия. Функции для работы с символами. Методы работы со строками. Базовые алгоритмы обработки строк. Примеры решения задач. Контрольные вопросы. Задачи для самостоятельного решения	Лек.	2	ОПК-7	Л1.1, Л1.2, Л2.1 Л2.2 Л2.3
1.17	Лабораторная работа 4. Работа со строками в Python	ЛР	4	ОПК-7	Л3.1 Л3.2
Модуль 2. Программирование на Python– (26 + 26 = 52 часа)					
2.1	Лекция 6. Работа с функциями. Создание модулей. Создание пользовательских функций. Создание модулей. Примеры решения задач. Контрольные вопросы. Задачи для самостоятельного решения.	Лек.	2	ОПК-7	Л1.1, Л1.2, Л2.1 Л2.2 Л2.3
2.2	Практическая работа 4. Функции и процедуры в Python	ПЗ	2	ОПК-7	Л1.1, Л3.2
2.3	Встроенные типы. Анонимные функции. Генераторы списков.	СРС	14	ОПК-7	Л1.1, Л2.3
2.4	Лекция 7. Работа с файлами. Запись информации в текстовый файл. Чтение информации из текстового файла. Запись информации в двоичный файл. Примеры решения задач. Контрольные вопросы. Задачи для самостоятельного решения.	Лек.	2	ОПК-7	Л1.1, Л1.2, Л2.1 Л2.2 Л2.3
2.5	Практическая работа 5. Создание оконных требований.	ПЗ	2	ОПК-7	Л1.1, Л3.2
2.6	Лекция 8. Объектно-ориентированное программирование на Python. Создание классов. Создание конструкторов. Инкапсуляция. Создание свойств. Наследование. Примеры решения задач. Контрольные вопросы. Задачи для самостоятельного решения.	Лек.	2	ОПК-7	Л1.1, Л1.2, Л2.1 Л2.2 Л2.3
2.7	Инструкция class. Методы. Наследование. Перегрузка операторов. Пространства имен.	СРС	12	ОПК-7	Л1.1, Л2.3
2.8	Лабораторная работа 5. Создание типа данных «класс»	ЛР	2	ОПК-7	Л1.1, Л3.1
2.9	Практическая работа 6. Наследование и полиморфизм	ПЗ	4	ОПК-7	Л1.1, Л3.2
2.10	Лекция 9. Событийно-ориентированное программирование на	Лек.	2	ОПК-7	Л1.1, Л1.2,

	Python. Создание формы и виджетов: Кнопка, Текстовое поле, Надпись. Создание виджета Флажок. Создание виджета Переключатель. Примеры решения задач. Контрольные вопросы. Задачи для самостоятельного решения				Л2.1 Л2.2 Л2.3
2.11	Лабораторная работа 6. Работа с графикой	ЛР	4	ОПК-7	Л1.1, Л3.1
2.12	Практическая работа 7. Сортировки	ПЗ	4	ОПК-7	Л1.1, Л3.1
	Зачет			ОПК-7	Л1.1, Л1.2, Л2.1 Л2.2 Л2.3
	Итого		108		

5. Учебно-методическое и информационное обеспечение дисциплины

5.1 Рекомендуемая литература				
5.1.1. Основная литература				
Код	Авторы, составители	Заглавие	Издательство, год	Кол.
Л1.1	Файфель Б.Л.	Основы языка Python: учебное пособие / Файфель Б.Л., Колобова А.Е.	Саратов: СГТУ, ЭБС АСВ, 2024	Э1
Л1.2	Агалаков А.А.	Программирование на языке Python. Базовый уровень: учебное пособие / Агалаков А.А., Дементьева К.И.	Новосибирск: СГУТиИ, 2024	Э2
5.1.2 Дополнительная литература				
Код	Авторы, составители	Заглавие	Издательство, год	Кол.
Л2.1	Дроботун Н.В., Рудков Е.О., Баев Н.А.	Алгоритмизация и программирование. Язык Python : учебное пособие	Санкт-Петербург : Санкт-Петербургский государственный университет промышленных технологий и дизайна, 2020. — 119 с.	Э3
Л2.2	Сузи Р.А.	Язык программирования Python : учебное пособие	Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2024. — 350 с.	Э4
Л2.3	Лысаков, К. Ф.	Практическое программирование на Python: учебное пособие / К. Ф. Лысаков	Новосибирск: НГУ, 2023	Э5
5.1.3 Учебно-методическое обеспечение				

Код	Авторы, составители	Заглавие	Издательство, год	Кол.
ЛЗ.1	Шамышев А.А., Шамышева О.Н.	Программирование на Python. Решение задач по основам алгоритмизации: практикум	Владимир: Издательство Владимирского государственного университета, 2025. — 176 с.	Э6
ЛЗ.2	Титов А.Н.	Введение в Tkinter. Разработка графических интерфейсов в Python: учебно-методическое пособие / Титов А.Н., Тазиева Р.Ф.	Казань: Издательство КНИТУ, 2023	Э7

5.2 Электронные образовательные ресурсы

Э1	https://www.iprbookshop.ru/147706.html
Э2	https://www.iprbookshop.ru/149539.html
Э3	https://www.iprbookshop.ru/102400.html
Э4	https://www.iprbookshop.ru/142310.html
Э5	https://www.iprbookshop.ru/134584.html
Э6	https://www.iprbookshop.ru/152332.html
Э7	https://www.iprbookshop.ru/136144.html

5.3 Информационно-справочные системы и профессиональные базы данных

5.3.1	Консультант Плюс – https://www.consultant.ru/
5.3.2	Гарант – https://www.garant.ru/
5.3.2	Консорциум Кодекс – https://docs.cntd.ru/
5.3.3	Научная электронная библиотека – https://www.elibrary.ru/
5.3.4	Профессиональная база данных: Федеральный портал «Российское образование»: [Электронный ресурс] – Режим доступа: http://www.edu.ru/ (открытый доступ)
5.3.5	Профессиональная база данных: Федеральный центр информационно-образовательных ресурсов: [Электронный ресурс] – Режим доступа: http://fcior.edu.ru/ (открытый доступ)

5.4 Программное обеспечение

СКФ МТУСИ обеспечен необходимым комплектом лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства:

П.1	CodeBlocks свободная кроссплатформенная среда разработки
П.2	IDLE Среда программирования на языке Python
П.3	LibreOffice Кроссплатформенный, свободно распространяемый офисный пакет
П.4	Яндекс браузер
П.5	Kaspersky Endpoint Security

6. Материально-техническое обеспечение дисциплины

6.1 МТО лекционных занятий

1	Лекционная аудитория, оснащенная проектором, ПК (ноутбуком), экраном
---	--

6.2 МТО лабораторных работ и практических занятий

1	Лаборатория «Вычислительной техники и информатики», укомплектованная специализированной мебелью, оборудованная компьютерной техникой, с возможностью подключения к сети «Интернет», и обеспечением доступа в электронную информационно-образовательную среду СКФ МТУСИ
2	Учебные аудитории, оборудованные компьютерной техникой, с возможностью

	подключения к сети «Интернет», и обеспечением доступа в электронную информационно-образовательную среду СКФ МТУСИ
6.3 МТО рубежных контролей, зачетов, экзаменов	
1	Учебные аудитории, оборудованные компьютерной техникой с возможностью выхода в локальную сеть Филиала и Интернет.
6.4 МТО самостоятельной работы обучающихся	
1	Помещение для самостоятельной работы обучающихся оснащено компьютерной техникой с возможностью подключения к сети Интернет и обеспечением доступа в электронную информационно-образовательную среду СКФ МТУСИ

7. Оценочные материалы
Оценочные материалы и перечень видов оценочных средств текущего контроля и промежуточной аттестации представлены в Приложении 1 к рабочей программе дисциплины.

8. Методические рекомендации для обучающихся по освоению дисциплины
Методические рекомендации по освоению дисциплины представлены в Приложении 2 к рабочей программе дисциплины.

9. Особенности реализации дисциплины (модуля) при обучении инвалидов и лиц с ограниченными возможностями здоровья

Рабочая программа дисциплины (модуля) при необходимости может быть адаптирована для обучения (в том числе с применением дистанционных образовательных технологий) лиц с ограниченными возможностями здоровья, инвалидов. Для этого требуется заявление обучающихся, являющихся лицами с ограниченными возможностями здоровья, инвалидами, или их законных представителей и рекомендации психолого-медико-педагогической комиссии. При обучении лиц с ограниченными возможностями здоровья учитываются их индивидуальные психофизические особенности. Обучение инвалидов осуществляется также в соответствии с индивидуальной программой реабилитации инвалида (при наличии).

Для лиц с нарушением слуха возможно предоставление учебной информации в визуальной форме (краткий конспект лекций; тексты заданий, напечатанные увеличенным шрифтом), на аудиторных занятиях допускается присутствие ассистента, а также сурдопереводчиков и тифлосурдопереводчиков.

Текущий контроль успеваемости осуществляется в письменной форме: обучающийся письменно отвечает на вопросы, письменно выполняет практические задания.

Доклад (реферат) также может быть представлен в письменной форме, при этом требования к содержанию остаются теми же, а требования к качеству изложения материала (понятность, качество речи, взаимодействие с аудиторией и т. д.) заменяются на соответствующие требования, предъявляемые к письменным работам (качество оформления текста и списка литературы, грамотность, наличие иллюстрационных материалов и т.д.).

Промежуточная аттестация для лиц с нарушениями слуха проводится в письменной форме, при этом используются общие критерии оценивания. При необходимости время подготовки к ответу может быть увеличено.

Для лиц с нарушением зрения допускается аудиальное предоставление информации, а также использование на аудиторных занятиях звукозаписывающих устройств (диктофонов и т.д.). Допускается присутствие на занятиях ассистента (помощника), оказывающего обучающимся необходимую техническую помощь.

Текущий контроль успеваемости осуществляется в устной форме.

При проведении промежуточной аттестации для лиц с нарушением зрения тестирование может быть заменено на устное собеседование по вопросам.

Для лиц с ограниченными возможностями здоровья, имеющих нарушения опорно-двигательного аппарата, на аудиторных занятиях, а также при проведении процедур текущего контроля успеваемости и промежуточной аттестации могут быть предоставлены необходимые технические средства (персональный компьютер, ноутбук или другой гаджет); допускается присутствие ассистента (ассистентов), оказывающего обучающимся необходимую техническую помощь (занять рабочее место, передвигаться по аудитории, прочитать задание, оформить ответ, общаться с преподавателем).

10. Дополнения и изменения в Рабочей программе

Лист актуализации рабочей программы дисциплины «Б1.О.37 Основы программирования на Python» для использования в 20__/20__ учебном году

Утверждаю

Зам. директора по УР _____
«__» _____ 20__ г.

Направление: 10.03.01 Информационная безопасность

Профиль: Безопасность компьютерных систем (по отрасли или в сфере профессиональной деятельности)

Форма обучения: очная

(Возможны следующие варианты):

а) Рабочая программа действует без изменений.

б) В рабочую программу вносятся следующие изменения:

- 1)
- 2)
- 3)

Разработчик (и): _____
(ФИО, ученая степень, ученое звание)

«__» _____ 20__ г.

Рабочая программа пересмотрена и одобрена на заседании кафедры ИВТ
Протокол № _____ от «__» _____ 20__ г.

Заведующий кафедрой _____

Оценочные материалы для проведения текущего контроля и промежуточной аттестации по дисциплине

1. Оценочные материалы для проведения промежуточной аттестации по дисциплине

1.1 Шкала оценивания компетенций

Шкала оценивания компетенций		
Оценка	Уровень освоения компетенции	Критерии оценивания
«Отлично»	Высокий уровень	Обучающийся показывает всестороннее, систематическое и глубокое знание основного и дополнительного учебного материала, умеет свободно выполнять задания, предусмотренные программой; усвоил основную и знаком с дополнительной рекомендованной литературой; может объяснить взаимосвязь основных понятий дисциплины в их значении для последующей профессиональной деятельности; проявляет творческие способности в понимании, изложении и использовании учебного материала.
«Хорошо»	Повышенный уровень	Обучающийся показывает достаточный уровень знаний в пределах основного учебного материала, без существенных ошибок выполняет предусмотренные в программе задания; усвоил основную литературу, рекомендованную в программе; способен объяснить взаимосвязь основных понятий дисциплины при дополнительных вопросах преподавателя. Допускает не существенные погрешности в ответах, устраняет их без помощи преподавателя.
«Удовлетворительно»	Пороговый уровень	Обучающийся показывает знания основного учебного материала в минимальном объеме, необходимом для дальнейшей учебы; справляется с выполнением заданий, предусмотренных программой, допуская при этом большое количество не принципиальных ошибок; знаком с основной литературой, рекомендованной программой. Допускает существенные погрешности в ответах, но обладает необходимыми знаниями для их устранения под руководством преподавателя.
«Неудовлетворительно»	Минимальный уровень не достигнут	Обучающийся обнаруживает пробелы в знаниях основного учебного материала, допускает принципиальные ошибки в выполнении предусмотренных программой

– основными понятиями, связанными с общим синтаксисом и созданием алгоритмов на языке Python	Лабораторные работы 3-4	Модуль 1 0÷5 за каждую ЛР «Неудовлетворительно» - 0-2 «Удовлетворительно» - 3 «Хорошо» - 4 «Отлично» - 5
	Практические занятия 3	Модуль 1 0÷5 за каждое ПЗ «Неудовлетворительно» - 0-2 «Удовлетворительно» - 3 «Хорошо» - 4 «Отлично» - 5
– навыками создания пользовательских функций, объектно-ориентированного программирования, событийно-ориентированного программирования на Python.	Лабораторные работы 5-6	Модуль 2 0÷5 за каждую ЛР «Неудовлетворительно» - 0-2 «Удовлетворительно» - 3 «Хорошо» - 4 «Отлично» - 5
	Практические занятия 6-7	Модуль 2 0÷5 за каждое ПЗ «Неудовлетворительно» - 0-2 «Удовлетворительно» - 3 «Хорошо» - 4 «Отлично» - 5
	Зачет	«Не зачтено» - 0-40 «Зачтено» - 41-100

1.3 Оценочные материалы: типовые контрольные задания, иные материалы

1.3.1 Оценочные материалы текущего контроля для очной формы обучения

Модуль 1 (50 баллов):

Модуль содержит 5 лекционных занятий, 3 практических и 4 лабораторных работы. Знание лекционного материала оценивается по контрольной работе, состоящей из двух вопросов, максимальное количество баллов за контрольную работу составляет 15. За выполненные и защищенные практические и лабораторные занятия обучающийся получает максимум 35 баллов. Общее максимальное количество баллов за модуль 1 составляет 50 баллов.

Вопросы для контрольной работы (ОПК-7):

1. Литералы чисел и строк в Python.
2. Условный оператор и каскадная условная конструкция в Python.
3. Строки в Python. Методы find, count, replace, strip, upper, lower и другие.
4. Строки в Python.
5. Срезы с двумя и тремя параметрами.
6. Кортежи в Python.
7. Создание и использование. Кортежи переменных.
8. Списки в Python.
9. Методы списков и операции со списками

10. Срезы списков.
11. Присваивание в срез.
12. Проблема копирования списка.
13. Списки в Python.
14. List comprehensions: генерация списков.
15. Цикл for и его особенности в Python. Функция range()
16. Двумерные массивы (списки списков). Вложенная генерация.
17. Классы в Python. Перегрузка операторов.
18. Классы в Python. Конструктор класса.
19. Классы в Python. Наследование.
20. Оператор yield и функции-генераторы.
21. Алгоритмические и теоретические вопросы:
22. Позиционные системы счисления
23. Связь двоичной и шестнадцатеричной систем счисления
24. Основы алгебры логики.
25. Операции И, ИЛИ, НЕ, исключающее или, импликация, эквиваленция.
26. Таблицы истинности функций
27. Основы алгебры логики.
28. Свойства операций И и ИЛИ
29. Основы алгебры логики.
30. Операция НЕ. Законы де Моргана
31. Основы алгебры логики.
32. Табличное задание логической функции.
33. Дизъюнктивная нормальная форма
34. Структурное программирование.
35. Декомпозиция задачи и проектирование «сверху-вниз».
36. Кодировки текста ASCII и Unicode.
37. Рекурсия.
38. Прямой и обратный ход рекурсии.
39. Стек вызовов.
40. Однопроходные алгоритмы: подсчёт, сумма, произведение.
41. Среднеквадратическое отклонение: однопроходный алгоритм.
42. Однопроходные алгоритмы: поиск числа в потоке, максимум.
43. Однопроходные алгоритмы: местоположение максимума, три максимальных элемента.

Практическая работа 1. Перевод чисел между различными системами счисления.

Контрольные вопросы (ОПК-7):

1. В каком из чисел содержится больше бит: 6 килобайт или 6 кибибайт?
2. Сколько килобайт в 14 MiB? В ответе представить только целую часть результата.
3. Сколько бит содержится в 2 KiB? В ответе написать только целую часть результата.
4. Дано десятичное число 135,0079. Требуется выполнить его перевод в двоичную систему счисления.
5. Можно ли отбросить два нуля после запятой при умножении дробной части на два при осуществлении перевода?
6. Система счисления Бергмана основана на использовании...
7. Выполнить перевод числа 110(Б) в десятичную. Ответ округлить до одного знака после запятой отбрасыванием более младших разрядов.
8. В результате перевода числа 44(10) в факториальную систему счисления было получено число 538(Ф). Корректным ли был перевод?

9. Вычислить значение выражения и результат представить в десятичной системе счисления: $20(14) + 18(13)$.
10. Пусть имеется $n=5$ чисел (1,2,3,4,5). Найти 33-ю перестановку. Ответ записать в виде 5 чисел подряд без знаков препинания, пробелов и т.п.
11. Имеется 22-ричная система счисления. Какой символ будет означать максимально возможное число каждого разряда?
12. Перевести число 1630, представленное в системе счисления с основанием -7, в десятичную.
13. В чём заключается разница между числами $BB(329)$ и $B\ B(329)$?
14. В чём преимущества и недостатки банковского способа округления?
15. Даны несколько чисел, заданных в системах счисления 2, 8 и 32.
16. Каким способом можно сравнить эти числа? Является ли данный способ самым быстрым?
17. Имеется число $F1E2D(16)$. Требуется перевести в систему счисления с основанием 11.
18. Пояснить, какими свойствами будет обладать новое число, не выполняя при этом перевод.

Практическая работа 2. Построение линейных алгоритмов в Python.

Контрольные вопросы (ОПК-7):

1. Даны действительные числа A, B, C . Найти максимальное и минимальное из этих чисел.
2. Известны длины трёх сторон треугольника. Вычислить периметр треугольника и площадь по формуле Герона (указание: использовать библиотеку `math` и функцию `sqrt()`).
3. Задан вес в граммах. Определить вес в тоннах и килограммах.
4. Известен объем информации в байтах. Перевести в Кбайты, Мбайты.
5. Определить значение функции $Z=1/(XY)$ при X и Y не равных 0.

Практическая работа 3. Работа с циклами в Python.

Контрольные вопросы (ОПК-7):

1. Даны действительные числа: X, Y, Z . Определить существует ли треугольник с такими длинами сторон и, если существует, будет ли он прямоугольным.
2. Дано число X . Определить принадлежит ли это число заданному промежутку $[a, b]$.
3. Определить значение функции $Z=1/(XY)$ при произвольных X и Y .
4. Даны действительные числа: A, B, C . Определить выполняются ли неравенства $A < B < C$ или $A > B > C$ и какое именно неравенство выполняется.
5. Даны действительные числа X и Y . Вычислить Z . $Z = X * Y$ при $X > Y$, $Z = \ln(X + Y)$, в противном случае.
6. Даны действительные положительные числа a, b, c, d . Выясните, может ли прямоугольник со сторонами a, b уместиться внутри прямоугольника со сторонами c, d так, чтобы каждая сторона внутреннего прямоугольника была параллельна или перпендикулярна стороне внешнего прямоугольника.
7. Дано действительное число A . Вычислить $f(A)$, если $f(x) = x^2 + 4x + 5$, при $x \leq 2$; в противном случае $f(x) = 1/(x^2 + 4x + 5)$.
8. Дано действительное число A . Вычислить $f(A)$, если $f(x) = 0$, при $x \leq 0$; $f(x) = x$, при $0 < x \leq 1$, в противном случае $f(x) = x^4$.
9. Дано действительное число A . Вычислить $f(A)$, если $f(x) = 0$ при $x \leq 0$; $f(x) = x^2 - x$ при $0 < x \leq 1$, в противном случае $f(x) = x^2 \sin(\pi * x^2)$.
10. Составит алгоритм и программу для реализации логических операций "И" и "ИЛИ" для двух переменных.

11. Известен ГОД. Определить будет ли этот год високосным, и к какому веку этот год относится.

Лабораторная работа 1. Введение в язык программирования Python.

Контрольные вопросы (ОПК-7):

1. Команда для ввода информации.
2. Команда для вывода информации.
3. Типы данных в Python.
4. Доступ к файлу Python.
5. Расширение файла Python.
6. Комментирование кода.
7. Как использовать среду IDLE.

Лабораторная работа 2. Математические операции в Python.

Контрольные вопросы (ОПК-7):

1. Перечислит унарные арифметические операции Python
2. Что понимается под унарной арифметической операцией в Python
3. Привести пример записи сложения двух чисел
4. Привести пример записи вычитания двух чисел
5. Привести пример записи умножения двух чисел
6. Привести пример записи деления двух чисел
7. Привести пример записи присвоения двух чисел
8. Привести пример записи получение целой части от деления двух чисел
9. Привести пример записи получение остатка от деления двух чисел
10. Привести пример записи возведения в степень.

Лабораторная работа 3. Структура ветвление в Python.

Контрольные вопросы (ОПК-7):

1. Дано натуральное число. Определить будет ли это число: чётным, кратным 4.
2. Дано натуральное число. Определить будет ли это число: нечётным, кратным 5.
3. Дано натуральное число. Определить будет ли это число: нечётным, кратным 7.
4. Дано натуральное число. Определить будет ли это число: чётным, кратным 10.
5. Имеется коробка со сторонами: $A \times B \times C$. Определить пройдёт ли она в дверь с размерами $M \times K$.
6. Дано вещественное число. Определить какое это число: положительное, отрицательное, ноль.
7. Можно ли из бревна, имеющего диаметр поперечного сечения D , выпилить квадратный брус шириной A ?
8. Можно ли в квадратном зале площадью S поместить круглую сцену радиусом R так, чтобы от стены до сцены был проход не менее K ?
9. Дан номер места в плацкартном вагоне. Определить, какое это место: верхнее или нижнее, в купе или боковое.
10. Известна денежная сумма. Разменять её купюрами 500, 100, 10 и монетой 2 руб., если это возможно.
11. Имеются две ёмкости: кубическая с ребром A , цилиндрическая с высотой H и радиусом основания R . Определить поместится ли жидкость объёма M в первую ёмкость, во вторую, в обе.
12. Имеются две ёмкости: кубическая с ребром A , цилиндрическая с высотой H и радиусом основания R . Определить можно ли заполнить жидкостью объёма M первую ёмкость, вторую, обе.

Лабораторная работа 4. Работа со строками в Python.

Контрольные вопросы (ОПК-7):

1. Дана строка, содержащая русскоязычный текст. Найти количество слов, начинающихся с буквы "е".
2. В строке заменить все двоеточия (:) знаком процента (%). Подсчитать количество замен.
3. В строке удалить символ точку (.) и подсчитать количество удаленных символов.
4. В строке заменить букву(а) буквой (о). Подсчитать количество замен. Подсчитать, сколько символов в строке.
5. В строке заменить все заглавные буквы строчными.
6. В строке удалить все буквы "а" и подсчитать количество удаленных символов.
7. Дана строка. Преобразовать ее, заменив звездочками все буквы "п", встречающиеся среди первых $n/2$ символов. Здесь n - длина строки.
8. Дана строка, заканчивающаяся точкой. Подсчитать, сколько слов в строке.
9. Определить, сколько раз в тексте встречается заданное слово.
10. Дана строка-предложение на английском языке. Преобразовать строку так, чтобы каждое слово начиналось с заглавной буквы.
11. Дана строка. Подсчитать самую длинную последовательность подряд идущих букв «н». Преобразовать ее, заменив точками все восклицательные знаки.
12. Дана строка. Вывести все слова, оканчивающиеся на букву "я".
13. Дана строка символов, среди которых есть одна открывающаяся и одна закрывающаяся скобки. Вывести на экран все символы, расположенные внутри этих скобок.
14. Дана строка. Вывести все слова, начинающиеся на букву "а" и слова оканчивающиеся на букву "я".
15. Дана строка текста. Подсчитать количество букв «т» в строке.

Модуль 2: (50 баллов):

Модуль содержит 4 лекционных, 4 практических занятий и 2 лабораторных работы. Знание лекционного материала оценивается по контрольной работе, состоящей из двух вопросов, максимальное количество баллов за контрольную работу составляет 20. За практические занятия обучающийся получает максимум 30 баллов. Общее максимальное количество баллов за модуль 1 составляет 50.

Вопросы для контрольной работы (ОПК-7):

1. Тест простоты числа.
2. Разложение числа на множители.
3. Наивный поиск подстроки в строке.
4. Алгоритм обращения массива.
5. Алгоритм циклического сдвига в массиве.
6. Сортировка обезьяны.
7. Сортировка вставками.
8. Сортировка выбором.
9. Сортировка методом пузырька.
10. Сортировка дурака
11. Сортировка подсчётом.
12. Поразрядная сортировка.
13. Рекурсивные сортировки. Быстрая сортировка.
14. Рекурсивные сортировки. Сортировка слиянием.
15. Пирамида (куча). Пирамидальная сортировка.
16. Виды сортировок. Устойчивость сортировок.
17. Алгоритм Евклида

18. Вычисление чисел Фибоначчи.
19. Ханойские башни.
20. Генерация всех перестановок (рекурсивная)
21. Использование списка как стека. Метод pop()
22. Динамическое программирование. Задачи о Кузнечике на прямой.
23. Динамическое программирование. Задачи о Короле на плоскости.
24. Списки: односвязный, двусвязный, кольцо.
25. Стек.
26. Очередь.
27. Хеш-функция. Открытая хеш-таблица.
28. Хеш-функция. Закрытая хеш-таблица.

Практическая работа 4. Функции и процедуры в Python.

Контрольные вопросы (ОПК-7):

1. Составить программу для вычисления площади разных геометрических фигур.
2. Даны 3 различных массива целых чисел (размер каждого не превышает 15). В каждом массиве найти сумму элементов и среднеарифметическое значение.
3. Вычислить площадь правильного шестиугольника со стороной a , используя подпрограмму вычисления площади треугольника.
4. Пользователь вводит две стороны трех прямоугольников. Вывести их площади.
5. Даны катеты двух прямоугольных треугольников. Написать функцию вычисления длины гипотенузы этих треугольников. Сравнить и вывести какая из гипотенуз больше, а какая меньше.
6. Преобразовать строку так, чтобы буквы каждого слова в ней были отсортированы по алфавиту.
7. Даны две дроби A/B и C/D (A, B, C, D — натуральные числа). Составить программу деления дроби на дробь. Ответ должен быть несократимой дробью. Использовать подпрограмму алгоритма Евклида для определения НОД.
8. Задана окружность $(x-a)^2 + (y-b)^2 = R^2$ и точки $P(p_1, p_2)$, $F(f_1, f_1)$, $L(l_1, l_2)$. Выяснить и вывести на экран, сколько точек лежит внутри окружности. Проверку, лежит ли точка внутри окружности, оформить в виде процедуры.
9. Даны две дроби A/B и C/D (A, B, C, D — натуральные числа). Составить программу вычитания из первой дроби второй. Ответ должен быть несократимой дробью. Использовать подпрограмму алгоритма Евклида для определения НОД.
10. Напишите программу, которая выводит в одну строчку все делители переданного ей числа, разделяя их пробелами.
11. Составить программу нахождения наибольшего общего делителя (НОД) и наименьшего общего кратного (НОК) двух натуральных чисел $НОК(A, B) = (A*B)/НОД(A, B)$. Использовать подпрограмму алгоритма Евклида для определения НОД.
12. Составить программу вычисления площади выпуклого четырехугольника, заданного длинами четырех сторон и диагонали.
13. Из заданного числа вычи сумму его цифр. Из результата вновь вычи сумму его цифр и т. д. Через, сколько таких действий получится нуль?
14. Даны 3 различных массива целых чисел. В каждом массиве найти произведение элементов и среднеарифметическое значение.
15. На отрезке $[100, N]$ ($210 < N < 231$) найти количество чисел, составленных из цифр a, b, c .
16. Составить программу, которая изменяет последовательность слов в строке на обратную.
17. Два простых числа называются «близнецами», если они отличаются друг от друга на 2 (например, 41 и 43). Напечатать все пары «близнецов» из отрезка $[n, 2n]$, где n — заданное натуральное число, большее 2.

Практическая работа 5. Создание оконных требований.

Контрольные вопросы (ОПК-7):

1. Размещение на окне программы элемента кнопка.
2. Размещение на окне программы элемента метка.
3. Размещение на окне программы элемента поле ввода.
4. Размещение на окне программы элемента text.
5. Размещение на окне программы элемента переключатели и флажки.
6. Размещение на окне программы элемента checkbutton.
7. Размещение на окне программы элемента список.
8. Размещение на окне программы элемента рамка.
9. Размещение на окне программы элемента дочерние окна.
10. Размещение на окне программы элемента шкала.
11. Размещение на окне программы элемента полоска прокрутки.
12. Размещение на окне программы элемента меню.

Практическая работа 6. Наследование и полиморфизм.

Контрольные вопросы (ОПК-7):

1. Что такое наследование и как оно реализуется в Python.
2. Что такое полиморфизм и как он реализуется в Python.
3. Написать программу, в которой описана иерархия классов: средство передвижения (велосипед, автомобиль, грузовик). Базовый класс должен иметь поля для хранения средней скорости, названия модели, числа пассажиров, а также методы получения потребления топлива для данного расстояния и вычисления времени движения на заданное расстояние. Продемонстрировать работу всех методов классов, предоставив пользователю выбор типа объекта для демонстрации.
4. Написать программу, в которой описана иерархия классов: человек («дошкольник», «школьник», «обучающийся», «работающий»). Базовый класс должен иметь поля для хранения ФИО, возраста, пола, а также методы получения среднего дохода и среднего расхода в денежном эквиваленте.
5. Продемонстрировать работу всех методов классов, предоставив пользователю выбор типа объекта для демонстрации.
6. Написать программу, в которой описана иерархия классов: геометрические фигуры (круг, прямоугольник, треугольник). Реализовать методы вычисления площади и периметра фигуры. Продемонстрировать работу всех методов классов, предоставив пользователю выбор типа фигуры для демонстрации.
7. Написать программу, в которой описана иерархия классов: геометрические фигуры (эллипс, квадрат, трапеция). Реализовать методы вычисления площади и периметра фигуры. Продемонстрировать работу всех методов классов, предоставив пользователю выбор типа фигуры для демонстрации.
8. Написать программу, в которой описана иерархия классов: геометрические фигуры (ромб, параллелепипед, эллипс). Реализовать методы вычисления площади и периметра фигуры. Продемонстрировать работу всех методов классов, предоставив пользователю выбор типа фигуры для демонстрации.

Практическая работа 7. Сортировки.

Контрольные вопросы (ОПК-7):

1. Зачем необходимо производить сортировку.
2. Произвести сортировку вставками для последовательности (5,6,2,9,11,0,-3,17,4)
3. Произвести сортировку выбором (5,6,2,9,11,0,-3,17,4)
4. Произвести сортировку пузырьком (5,6,2,9,11,0,-3,17,4)
5. Произвести сортировку Шелла (5,6,2,9,11,0,-3,17,4)

6. Произвести сортировку слиянием (5,6,2,9,11,0,-3,17,4)
7. Произвести сортировку сортировка (5,6,2,9,11,0,-3,17,4)
8. Произвести сортировку вставками (5,6,2,9,11,0,-3,17,4)

Лабораторная работа 5. Создание типа данных «Класс».

Контрольные вопросы (ОПК-7):

1. Что такое класс в ООП python
2. Что такое метод класса в ООП python
3. Что такое атрибут класса в ООП python
4. Работа конструктора класса — метод `__init__`
5. Что значит аргумент `self` в Python 3 в методе класса.
6. Построить класс человек.
7. Построить класс легковой автомобиль.
8. Построить класс самолёт.
9. Построить класс число.
10. Построить класс строка.

Лабораторная работа 6. Работа с графикой.

Контрольные вопросы (ОПК-7):

1. Построить график функции $y = a \sin(bx + c) - d \cdot \cos(gx^2 + fx + h)$
2. Построить график функции $y = a \sin(bx)$
3. Построить график функции $y = a \cos(bx^2 + cx + d)$
4. Построить график функции $y = ax^3 + bx^2 + cx + d$
5. Построить график функции $y = ax^5 + bx^4 + cx^3 + dx^2 + fx + g$
6. Построить график функции $y = ax^4 + bx^3 + cx^2 + dx + f$
7. Построить график функции $y = (ax^2 + bx + c) \cdot \sin(dx)$
8. Построить график функции $y = (ax^3 + bx^2 + cx + d) \cdot \sin(x^2)$
9. Построить график функции $y = (ax^4 + bx^3 + cx^2 + dx + f) \cdot \cos(ax^2 + bx + c)$
10. Построить график функции $y = e \sin(ax^2 + bx + c) + d \cdot \cos(fx + g)$
11. Построить график функции $y = d \sin(ax^2 + bx + c) \cdot \exp(-fx)$
12. Построить график функции $y = a \cos(bx) \cdot \exp(-cx)$
13. Построить график функции $y = a \exp(bx) + c$
14. Построить график функции $y = a \ln(bx + c) + d \cdot \lg(fx)$
15. Построить график функции $y = abx + c + f$

1.3.2 Вопросы, выносимые на зачет по дисциплине «Основы программирования на Python» (ОПК-7) (промежуточная аттестация):

1. Литералы чисел и строк в Python.
2. Условный оператор и каскадная условная конструкция в Python.
3. Строки в Python. Методы `find`, `count`, `replace`, `strip`, `upper`, `lower` и другие.
4. Строки в Python.
5. Срезы с двумя и тремя параметрами.
6. Кортежи в Python.
7. Создание и использование. Кортежи переменных.
8. Списки в Python.
9. Методы списков и операции со списками
10. Срезы списков.
11. Присваивание в срез.
12. Проблема копирования списка.
13. Списки в Python.
14. List comprehensions: генерация списков.

15. Цикл for и его особенности в Python. Функция range()
16. Двумерные массивы (списки списков). Вложенная генерация.
17. Классы в Python. Перегрузка операторов.
18. Классы в Python. Конструктор класса.
19. Классы в Python. Наследование.
20. Оператор yield и функции-генераторы.
21. Алгоритмические и теоретические вопросы:
22. Позиционные системы счисления
23. Связь двоичной и шестнадцатеричной систем счисления
24. Основы алгебры логики.
25. Операции И, ИЛИ, НЕ, исключающее или, импликация, эквиваленция.
26. Таблицы истинности функций
27. Основы алгебры логики.
28. Свойства операций И и ИЛИ
29. Основы алгебры логики.
30. Операция НЕ. Законы де Моргана
31. Основы алгебры логики.
32. Табличное задание логической функции.
33. Дизъюнктивная нормальная форма
34. Структурное программирование.
35. Декомпозиция задачи и проектирование «сверху-вниз».
36. Кодировки текста ASCII и Unicode.
37. Рекурсия.
38. Прямой и обратный ход рекурсии.
39. Стек вызовов.
40. Однопроходные алгоритмы: подсчёт, сумма, произведение.
41. Среднеквадратическое отклонение: однопроходный алгоритм.
42. Однопроходные алгоритмы: поиск числа в потоке, максимум.
43. Однопроходные алгоритмы: местоположение максимума, три максимальных элемента.
44. Тест простоты числа.
45. Разложение числа на множители.
46. Наивный поиск подстроки в строке.
47. Алгоритм обращения массива.
48. Алгоритм циклического сдвига в массиве.
49. Сортировка обезьяны.
50. Сортировка вставками.
51. Сортировка выбором.
52. Сортировка методом пузырька.
53. Сортировка дурака
54. Сортировка подсчётом.
55. Поразрядная сортировка.
56. Рекурсивные сортировки. Быстрая сортировка.
57. Рекурсивные сортировки. Сортировка слиянием.
58. Пирамида (куча). Пирамидальная сортировка.
59. Виды сортировок. Устойчивость сортировок.
60. Алгоритм Евклида
61. Вычисление чисел Фибоначчи.
62. Ханойские башни.
63. Генерация всех перестановок (рекурсивная)
64. Использование списка как стека. Метод pop()
65. Динамическое программирование. Задачи о Кузнечике на прямой.

66. Динамическое программирование. Задачи о Короле на плоскости.
67. Списки: односвязный, двусвязный, кольцо.
68. Стек.
69. Очередь.
70. Хеш-функция. Открытая хеш-таблица.
71. Хеш-функция. Закрытая хеш-таблица.

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

Порядок и методика проведения текущего контроля и промежуточной аттестации

Предусмотрены следующие виды контроля:

- текущий контроль по каждому модулю – в форме написания контрольной работы для оценки теоретических знаний;
- текущий контроль по каждому модулю – в форме отчетов по лабораторным работам и практическим занятиям для оценки практических навыков;
- промежуточная аттестация по дисциплине – в форме зачета.

Текущий контроль успеваемости в форме защиты лабораторных и практических проводится в два этапа:

- 1-й этап: допуск к выполнению лабораторной работы – проводится в форме письменной «летучки» (5-10 мин) с целью контроля знаний обучающихся теоретической части лабораторной работы и готовности к выполнению практических исследований;
- 2-й этап выполняется по окончанию каждой лабораторной работы (практического занятия) в форме индивидуального собеседования по выполненным исследованиям или расчетам. Проводится с целью контроля закрепления теоретической части материала и степени отработки обучающимся практических навыков исследования на ЭВМ.

Контрольные работы выполняются в виде короткого письменного ответа на один вопрос, изученный на предыдущей лекции в начале каждой последующей лекции. Ответ на вопрос дается в течение 5-10 минут. Таким образом, после лекционного курса каждого модуля формируется общая оценка за теоретические знания.

С целью повышения качества обучения за счет побуждения обучающихся к активной текущей учебной работе, четкого и оперативного контроля всего хода учебного процесса, снижения роли случайных и субъективных факторов при оценивании учебной деятельности обучающихся в образовательном процессе реализована модульно-рейтинговая система.

Правила ее использования прописаны в «Положении об МРС».

Набранные обучающимся в рамках текущего контроля баллы переводятся в оценку в соответствии со следующей шкалой:

- «не зачтено» - от 0 до 40 от максимального количества баллов;
- «зачтено» - от 41 до 100 максимального количества баллов.

Соотношения максимального количества баллов, полученных обучающимся по блокам модулей, показаны в Таблице 1.

Таблица 1 - Распределение баллов по модулям дисциплины «Основы программирования на Python»

Модуль	Всего баллов (Максимальное значение)	Теоретический блок (Контрольная работа)	Практический блок
Модуль 1	50	15	35
Модуль 2	50	20	30
Зачет	100		

Теоретический блок оценивается по результатам контрольной работы. Практический блок оценивается по результатам выполнения лабораторных работ и заданий на практических занятиях.

На зачете производится оценка тех компетенций, которые должны быть в той или иной форме освоены в процессе изучения. Рекомендуется формировать вопросы в билетах к зачету таким образом, чтобы преподаватель смог оценить все компетенции данной дисциплины.

3. Промежуточная аттестация

Промежуточная аттестация проводится в форме зачета. Зачет проводится по расписанию в устном виде по билетам к зачету. Результаты аттестации заносятся в ведомость и зачетную книжку обучающегося.

Образец билета к зачету.

	МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ Северо-Кавказский филиал ордена Трудового Красного Знамени федерального государственного бюджетного образовательного учреждения высшего образования «Московский технический университет связи и информатики»	Утверждаю Зав. кафедрой «ИВТ» _____ Соколов С.В. «___» _____ 20___ г.
Направление подготовки: 10.03.01 Информационная безопасность Курсы: 2 Дисциплина: Основы программирования на Python		
Билет №1 1. Условный оператор и каскадная условная конструкция в Python. 2. Вычисление чисел Фибоначчи. ст. преподаватель кафедры ИВТ _____ Осмоловская Н.С. «___» _____ 20___ г.		

Один комплект отпечатанных билетов к зачету, подписанных преподавателем кафедры и утвержденных заведующим кафедрой хранится у заведующего кафедрой, другой комплект – у преподавателя, ведущего дисциплину.

4. ТЕСТОВЫЕ ЗАДАНИЯ ПО ДИСЦИПЛИНЕ (ОПК-7)

Тестовые задания позволяют оценить уровень сформированности компетенции ОПК-7.

Задания распределены по блокам в соответствии с уровнем сложности. Каждый блок содержит номер задания, и текст задания.

Базовый уровень содержит примерно 50 % заданий.

Он формируется из заданий с выбором одного или нескольких верных ответов из предложенных или воспроизведения фактического материала (терминология, факты, классификация, параметры и др.).

Повышенный уровень в среднем составляет 35 % заданий.

В нем используются задания на сопоставление, сравнение, установление последовательности.

Высокий уровень в среднем составляют – 15 % заданий.

В задании используются: решения нетиповых задач, алгоритмы, доказательства, задания с развернутым ответом – определения, перечисление, изображение схемы, структуры и др.

Количество заданий по компетенции равно 20.

Один комплект тестовых вопросов с указанием правильных ответов хранится у заведующего кафедрой, другой комплект – у преподавателя, ведущего дисциплину.

ОПК-7. Способен использовать языки программирования и технологии разработки программных средств для решения задач профессиональной деятельности.

БЛОК А (базовый уровень) – задание с выбором одного или нескольких верных ответов из предложенных вариантов

Инструкция по тестам Блока А: прочитайте текст и выберите один или несколько правильных ответов

№	Дисциплина	Тексты заданий
1	Б1.О.37 Основы программирования на Python	С помощью оператора *** можно выполнить операцию: 1. возведения в степень 2. умножения 3. это ошибочно поставленный символ
2	Б1.О.37 Основы программирования на Python	Функция int () 1. используется для преобразования строки, массива объектов и объектов в целое 2. используется для преобразования строки, массива объектов и объектов в список (массив объектов) 3. используется для преобразования строки, массива объектов и объектов в тип дробное
3	Б1.О.37 Основы программирования на Python	A='pri', s='vet'. A+s – это: 1. копирование 2. конкатенация 3. ввод строк
4	Б1.О.37 Основы программирования на Python	Последовательность элементов, которые разделены между собой запятой и заключены в скобки – неизменяемый упорядоченный тип данных: 1. кортеж 2. словарь 3. список
5	Б1.О.37 Основы программирования на Python	Цикл for называется циклом: 1. с постусловием 2. с предусловием 3. с параметром
6	Б1.О.37 Основы программирования на Python	Выберите все формы условного оператора: 1. if 2. match 3. elif 4. switch 5. if_true/ if_false

№	Дисциплина	Тексты заданий
7	Б1.О.37 Основы программирования на Python	Выберите правильные имена переменных: 1. a,f,k 2. print 3. 23ty 4. T_23y 5. Л_23у
8	Б1.О.37 Основы программирования на Python	Для кортежа из одного элемента обязательно добавлять ... после имени или значения этого элемента: 1. точку с запятой 2. двоеточие 3. тире 4. запятую
9	Б1.О.37 Основы программирования на Python	Для каких типов элементов списка метод sort() работает без ошибок? 1. только для целочисленных 2. для однотипных элементов одного списка 3. для любых типов элементов 4. только для строковых
10	Б1.О.37 Основы программирования на Python	Какой метод отвечает за добавление элемента в конец списка? 1. intend() 2. extend() 3. addend() 4. append()

БЛОК Б (повышенный уровень) – задание закрытого типа на установление соответствия
Инструкция: Прочитайте текст и установите соответствие

№	Дисциплина	Тексты заданий
11	Б1.О.37 Основы программирования на Python	Установите соответствие между понятиями и их определениями: 1. Очередь – 2. Список – 3. Кортеж – 4. Строки – А) это один из основных типов данных, который используют для хранения и обработки текстовой информации Б) это абстрактный тип данных, представляющий последовательность элементов, упорядоченных в соответствии с набором правил В) это последовательность элементов, которая является неизменяемым типом Г) это упорядоченная, изменяемая коллекция элементов, которые могут быть разных типов
12	Б1.О.37 Основы программирования на Python	Установите соответствие между понятиями и их определениями: 1. int 2. float 3. str 4. list 5. bool А) список элементов Б) целое число В) логический тип Г) число с плавающей запятой Д) строка текста
13	Б1.О.37 Основы программирования на Python	Установите соответствие между понятиями и их определениями: 1. IndexError 2. TypeError 3. KeyError 4. ValueError 5. NameError 6. SyntaxError 7. ZeroDivisionError

№	Дисциплина	Тексты заданий
		А) деление на ноль Б) исключение, возникающее при обращении к несуществующему ключу в словаре В) исключение, возникающее при синтаксической ошибке в коде Г) исключение, возникающее при получении аргумента с некорректным значением, например, при преобразовании строки в число Д) исключение, возникающее при обращении к элементу списка по несуществующему индексу Е) исключение, возникающее при обращении к несуществующему имени (переменной, функции и т.д.) Ж) исключение, возникающее при попытке выполнить операцию с несоответствующим типом данных
14	Б1.О.37 Основы программирования на Python	Установите соответствие между понятиями и их определениями: 1. def 2. return 3. global 4. nonlocal 5. yield 6. lambda А) используется для создания анонимных функций Б) используется для определения новой функции В) используется для возврата значения из функции Г) используется в функциях-генераторах Д) используется для указания, что нужно искать переменную во внешней области видимости используется для указания, что переменная внутри функции ссылается на глобальную переменную

БЛОК Б (повышенный уровень) – задание закрытого типа на установление последовательности

Инструкция: Прочитайте текст и установите правильную последовательность

№	Дисциплина	Тексты заданий
15	Б1.О.37 Основы программирования на Python	Расставьте строки кода программы в правильной последовательности: 1. imin = 0 2. T = [3,5,6,4,2,9,7,5,4,7] 3. if T[i] < T[imin]: imin = i 4. print ("Минимальная температура: ", T[imin]) 5. for i in range(1, 10): 6. print ("Среднесуточная температура за декаду",T)
16	Б1.О.37 Основы программирования на Python	Расставьте языки программирования в хронологическом порядке их появления: 1. PASCAL 2. MOJO 3. KOTLIN 4. SWIFT 5. VISUAL BASIC и PYTHON 6. ALGOL 7. SQL и C
17	Б1.О.37 Основы программирования на Python	Расположите операторы по их приоритету выполнения, начиная с самого высокого: 1. ^ 2. () 3. not 4. <= и >= 5. + и - 6. and

БЛОК В (высокий уровень) – задание открытого типа с развернутым ответом

Инструкция: Прочитайте текст и запишите обоснованный ответ

№	Дисциплина	Тексты заданий
18	Б1.О.37 Основы программирования на Python	Определите значение переменной «а» после выполнения фрагмента программы: <code>a = 10</code> <code>b = 5</code> <code>if a > 1 or a < b:</code> <code> a -= 5</code> <code>if a > 1 and a == b:</code> <code> a -= 5</code>
19	Б1.О.37 Основы программирования на Python	Какую логическую операцию нужно добавить в программу вместо многоточия, чтобы значение переменной «а» после выполнения фрагмента программы стало равно 15? <code>a = 10</code> <code>b = 5</code> <code>if a < 1 ... a > b</code> <code> a -= 5</code> <code>else:</code> <code> a += 5</code>
20	Б1.О.37 Основы программирования на Python	Что выведет программа <code>a = [1, 2, 3, 4]</code> в 4й строке? 1. <code>a[0]</code> 2. <code>a[2]</code> 3. <code>a[3]</code> 4. <code>a[4]</code>

Система оценивания выполнения тестовых заданий по компетенции ОПК-7

Номер задания	Указания по оцениванию	Результат оценивания (баллы, полученные за выполнение задания/характеристика правильности ответа)
Задания 1-10	Задания с выбором одного или нескольких верных ответов из предложенных считается верным, если правильно указана цифра или цифры, означающие верные ответы.	Совпадение с верным ответом оценивается 1 баллом; неверный ответ или его отсутствие - 0 баллов.
Задания 11-14	Задания закрытого типа на установление соответствия считается верным, если правильно установлены все соответствия (позиции из одного столбца верно сопоставлены с позициями другого)	Полное совпадение с верным ответом оценивается 1 баллом; неверный ответ или его отсутствие - 0 баллов.
Задания 15-17	Задания закрытого типа на установление последовательности считается верным, если правильно указана вся последовательность цифр	Полное совпадение с верным ответом оценивается 1 баллом; если допущены ошибки или ответ отсутствует - 0 баллов.
Задания 18-20	Задание открытого типа с развернутым ответом считается верным, если ответ совпадает с эталонным по содержанию и полноте.	Полный правильный ответ на задание оценивается 1 баллом. Если в ответе используются перечисления, то различный порядок следования правильных ответов в перечислении не считается ошибкой. Если допущена ошибка, ответ не полный, неправильный или отсутствует - 0 баллов.

Методические рекомендации по освоению дисциплины

1. Методические рекомендации преподавателю

Перед началом изучения дисциплины преподаватель должен ознакомить обучающихся с рабочей программой и оценочными материалами по дисциплине, с видами учебной и самостоятельной работы, перечнем литературы и интернет-ресурсов, с формами текущей и промежуточной аттестации, с критериями оценки качества знаний для итоговой оценки по дисциплине.

При проведении лекций, преподаватель:

- 1) формулирует тему и цель занятия, объявляет учебные вопросы;
- 2) излагает основные теоретические положения;
- 3) с помощью технических средств обучения и/или под запись дает определения основных понятий, расчетных формул;
- 4) проводит примеры из отечественного и зарубежного опыта, дает текущие статистические данные для наглядного и образного представления изучаемого материала;
- 5) в конце занятия выдает вопросы для самостоятельного изучения.

На занятиях лабораторного цикла следует обратить внимание на соответствие выбираемых обучающимся средств выполнения решаемым в работе задачам.

Каждая лабораторная работа должна быть оформлена и защищена в соответствии с требованиями. Защита производится после оформления отчета по работе.

Во время выполнения заданий в учебной аудитории обучающийся может консультироваться с преподавателем, определять наиболее эффективные методы решения поставленных задач. Если какая-то часть задания остается не выполненной, обучающийся может продолжить её выполнение во время внеаудиторной самостоятельной работы.

Для оценки полученных знаний и освоения учебного материала по каждому разделу и в целом по дисциплине преподаватель использует формы текущего контроля и промежуточной аттестации знаний обучающихся.

2. Методические рекомендации обучающимся

Приступая к изучению новой учебной дисциплины, обучающиеся должны ознакомиться с рабочей программой, оценочными материалами, учебной, научной и методической литературой, имеющейся в ЭИОС СКФ МТУСИ,

Глубина усвоения дисциплины зависит от активной и систематической работы обучающегося на лекциях и практических занятиях, а также в ходе самостоятельной работы, по изучению рекомендованной литературы.

2.1 Методические указания для обучающихся по лекционным занятиям

Важно сосредоточить внимание на содержании лекции. Это поможет лучше воспринимать учебный материал и уяснить взаимосвязь проблем по всей дисциплине.

Основное содержание лекции целесообразнее записывать в тетради в виде ключевых фраз, понятий, тезисов, обобщений, схем, опорных выводов. Необходимо обращать внимание на термины, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации. Желательно оставлять в конспектах поля, на которых делать пометки из рекомендованной литературы, дополняющей материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений.

С целью уяснения теоретических положений, разрешения спорных ситуаций необходимо задавать преподавателю уточняющие вопросы. Для закрепления содержания лекции в памяти, необходимо во время самостоятельной работы внимательно прочесть свой конспект и дополнить его записями из учебников и рекомендованной литературы.

Конспектирование читаемых лекций и их последующая доработка способствует более глубокому усвоению знаний, и поэтому являются важной формой учебной деятельности обучающихся.

2.2 Методические указания для обучающихся по подготовке к лабораторным работам

Лабораторные занятия по любой учебной дисциплине проводятся в соответствии с рабочей программой этой учебной дисциплины в установленные расписанием часы.

Инструктаж по технике безопасности, правилам пожарной безопасности и правилам внутреннего распорядка в лаборатории проводится преподавателем на первом лабораторном занятии. На этом же занятии обучающимся сообщаются:

- программа всего предстоящего лабораторного цикла;
- условия взаимодействия обучающихся с преподавателем в процессе выполнения лабораторных работ;
- условия контроля самостоятельной работы обучающихся, включая правила оформления отчетов по лабораторным работам и их последующей защиты;
- другая необходимая информация.

Организация лабораторных работ включает:

- самостоятельную внеаудиторную подготовку обучающегося к выполнению каждой отдельной лабораторной работы в соответствии с ее программой в рамках часов, выделенных на самостоятельную проработку материала в программе дисциплины;
- входной контроль преподавателем степени подготовленности каждого обучающегося к выполнению лабораторных работ;
- выполнение программы лабораторных работ в полном объеме;
- оформление отчета и его защиту каждым обучающимся в установленные сроки;
- формирование преподавателем рейтингов каждого из обучающихся по результатам выполнения и защиты им отдельных лабораторных работ и их циклов (используется Модульно-рейтинговая система).

Оценка качества выполнения лабораторных работ каждым обучающимся производится преподавателем отдельно за подготовку к работе, ее выполнение и защиту.

2.3 Методические указания для обучающихся по подготовке к практическим занятиям

Целью практических занятий является закрепление теоретических знаний, полученных при изучении дисциплины.

При подготовке к практическому занятию целесообразно выполнить следующие рекомендации:

- изучить основную литературу; ознакомиться с дополнительной литературой, новыми публикациями в периодических изданиях: журналах, газетах и т.д.;
- при необходимости доработать конспект лекций. При этом учесть рекомендации преподавателя и требования учебной программы.

При выполнении практических занятий основным методом обучения является самостоятельная работа обучающегося под управлением преподавателя. На них пополняются теоретические знания обучающихся, их умение творчески мыслить, анализировать, обобщать изученный материал, проверяется отношение обучающихся к будущей профессиональной деятельности.

Оценка выполненной работы осуществляется преподавателем комплексно: по результатам выполнения заданий, устному сообщению и оформлению работы.

После подведения итогов занятия обучающийся обязан устранить недостатки, отмеченные преподавателем при оценке его работы.

2.4 Учебно-методическое обеспечение проведения лабораторных и практических занятий

Методические указания по выполнению лабораторных и практических работ по дисциплине «Основы программирования на Python» для обучающихся очной формы обучения:

1. Для подготовки и выполнения практических занятий используются методическая разработка: Сосновский И.А. Методическое пособие по выполнению практических работ. Учебное пособие. – Ростов-на-Дону: СКФ МТУСИ, 2022. РнД: СКФ МТУСИ, 2022

2. Для подготовки и выполнения лабораторных занятий используются методическая разработка: Сосновский И.А. Методическое пособие по выполнению лабораторных работ. Учебное пособие. – Ростов-на-Дону: СКФ МТУСИ, 2022. РнД: СКФ МТУСИ, 2022

3. Для подготовки и выполнения контрольной работы используются методическая разработка: Сосновский И.А. Методическое пособие по выполнению контрольной работы. Учебное пособие. – Ростов-на-Дону: СКФ МТУСИ, 2022. РнД: СКФ МТУСИ, 2022

3. Методические рекомендации для обучающихся по самостоятельной работе

Самостоятельная работа обучающихся является составной частью учебной работы и имеет целью закрепление и углубление полученных знаний и навыков, поиск и приобретение новых знаний, в том числе с использованием автоматизированных обучающих курсов (систем), а также выполнение учебных заданий, подготовку к предстоящим занятиям, зачетам и экзаменам.

Самостоятельная работа во внеаудиторное время включает в себя:

- повторение лекционного материала;
- подготовки к лабораторным работам (практическим занятиям);
- изучения учебной и научной литературы;
- решения задач, выданных на практических занятиях;
- подготовки к тестированию;
- выделение наиболее сложных и проблемных вопросов по изучаемой теме, получение разъяснений и рекомендаций по данным вопросам с преподавателями кафедры на их еженедельных консультациях.
- проведение самоконтроля путем ответов на вопросы текущего контроля знаний, решения представленных в учебно-методических материалах кафедры задач, тестов, написания рефератов по отдельным вопросам изучаемой дисциплины.

Перед выполнением внеаудиторной самостоятельной работы преподаватель проводит инструктаж (консультацию) с определением цели задания, его содержания, сроков выполнения, основных требований к результатам работы, критериев оценки, форм контроля и перечня источников и литературы.

Обычно постановку задачи обучающимся на проведение самостоятельной работы преподаватель осуществляет на одном из занятий, предшествующему данному.

Методику самостоятельной работы все обучающиеся выбирают индивидуально, но методика достижения конечной цели может определяться преподавателем и включать: последовательность изучения и усвоения учебно-методического материала, пособий, руководств, наставлений, техники и т.д.; определение главного в изучаемом материале, материале, который необходимо законспектировать; просмотр учебных кинофильмов и их обсуждение; работу обучающихся по индивидуальным заданиям; опрос обучающихся в течении 7-10 минут с целью проверки усвоения главного из прочитанного материала.

При возникновении затруднений у обучающихся в разрешении вопросов задания преподавателю необходимо предусмотреть, чтобы каждый обучающийся мог получить

оперативную консультацию по любому вопросу, если же при самостоятельной работе возникают затруднения по одному и тому же материалу (вопросу) у многих обучающихся, то желательно провести групповую консультацию.

Для контроля усвоения учебного материала целесообразно проводить в групповое собеседование или обсуждение изучаемого материала, проведение контрольных работ и т.п.

Контрольные мероприятия при должной их организации позволяют не только оценивать знания материала, но и углубить и закрепить его у обучающихся.

Приветствуется использование компьютеров, которое:

- расширяет информационную базу учебных занятий;
- повышает активность обучающихся, из пассивного получателя информации они превращаются в её добытчиков:
- способствует развитию способностей к анализу и обобщению, улучшает связанность, широту и глубину мышления;
- облегчает усвоение абстрактного материала, позволяет многое из него представить в виде конкретных образов;
- приучает к точности, аккуратности, последовательности действий способствует развитию самостоятельности.

Компьютерные технологии и программные продукты для выполнения самостоятельной работы, предусмотренной разделом 4 рабочей программы, необходимо использовать в соответствии с указаниями методических разработок.

Таблица – Учебный материал, выносимый на самостоятельное изучение обучающимся очной формы обучения.

№ п/п	Темы, разделы, вынесенные на самостоятельную подготовку, вопросы для подготовки к практическим и лабораторным занятиям; курсовые работы, содержание контрольных работ; рекомендации по использованию литературы, ЭВМ и др.	Час
Модуль 1 – 28 часов		
1	Документирование кода в Python. PEP 257. документации.	4
2	Однострочные строки. Многострочные строки документации.	4
3	Числа: целые, вещественные, комплексные.	4
4	Функции и методы строк	4
5	PEP 8 - руководство по написанию кода на Python	4
6	Внешний вид кода.	4
7	Пробелы в выражениях и инструкциях. Комментарии.	2
8	Соглашения по именованию. Общие рекомендации.	2
Модуль 2 – 26 часов		
9	Встроенные типы.	4
10	Анонимные функции.	4
11	Генераторы списков.	4
12	Инструкция class.	4
13	Методы	4
14	Наследование	2
15	Перегрузка операторов.	2
16	Пространства имен.	2

Самостоятельная работа обучающихся над усвоением материала по дисциплине может выполняться в помещении для самостоятельной работы СКФ МТУСИ, посредством использования электронной библиотеки и ЭИОС.