



Федеральное государственное бюджетное образовательное учреждение  
высшего образования

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ  
АРХИТЕКТУРНО-СТРОИТЕЛЬНЫЙ УНИВЕРСИТЕТ

Кафедра Информатики

УТВЕРЖДАЮ  
Начальник учебно-методического управления

«29» июня 2023 г.

## **РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)**

Основы программирования на Python

направление подготовки/специальность 35.03.10 Ландшафтная архитектура

направленность (профиль)/специализация образовательной программы Ландшафтная архитектура

Форма обучения очная

Санкт-Петербург, 2023

## 1. Цели и задачи освоения дисциплины (модуля)

Целями освоения дисциплины являются получение знаний, умений и навыков в области разработки компьютерных программ на языке Python для решения прикладных задач в различных сферах деятельности на базе:

- системного изложения основ алгоритмического языка Python, его возможностей по разработке прикладного программного обеспечения;
- ознакомления студентов с современными технологиями и инструментальными средствами разработки программного обеспечения, основными структурами данных и методами работы с ними, в том числе с использованием библиотеки стандартных модулей.

Задачами освоения дисциплины являются:

- обучение студентов навыкам постановки задачи, разработки алгоритмов, выбора структуры и визуализации данных;
- составление как процедурно-ориентированных, так и объектно-ориентированных программ на языке Python для решения широкого круга прикладных задач в научных, инженерных и экономических расчетах, обработки текстовой, графической и другой информации.

## 2. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с индикаторами достижения компетенций

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине, обеспечивающие достижение планируемых результатов освоения ОПОП
ОПК-7 Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности	ОПК-7.3 Составляет алгоритм решения сформулированной задачи	<b>знает</b> <ul style="list-style-type: none"><li>- основные принципы процедурного программирования;</li><li>- основные парадигмы объектно-ориентированного программирования;</li><li>- основные функции и классы языка Python;</li><li>- основные типы данных языка Python;</li><li>- основные структуры данных.</li></ul> <b>умеет</b> <ul style="list-style-type: none"><li>- разрабатывать программное обеспечение как в процедурном, так и в объектно-ориентированном стиле;</li><li>- использовать при разработке программного обеспечения библиотечные функции, классы и структуры данных языка Python;</li><li>- создавать собственные типы данных;</li><li>- создавать собственные структуры данных.</li></ul> <b>владеет</b> <ul style="list-style-type: none"><li>- навыками разработки процедурного программного обеспечения с использованием стандартных средств языка Python;</li><li>- разработки объектно-ориентированного программного обеспечения с использованием прикладных библиотек языка Python;</li><li>- разработки графических пользовательских интерфейсов;</li><li>- отладки программного обеспечения на языке Python;</li><li>- сборки и запуска Python-проектов в различных средах разработки.</li></ul>

### 3. Указание места дисциплины (модуля) в структуре образовательной программы

Данная дисциплина (модуль) включена в Блок «Дисциплины, модули» ФТД.03 основной профессиональной образовательной программы 35.03.10 Ландшафтная архитектура и относится к факультативным дисциплинам ОПОП.

№ п/п	Предшествующие дисциплины	Код и наименование индикатора достижения компетенции
1	Информационные технологии	ОПК-7.1, ОПК-7.2, ОПК-7.3, УК-1.1, УК-1.2, УК-1.6

Для успешного освоения дисциплины обучающийся должен знать:

- основные понятия информатики;
- основы разработки алгоритмов;
- основные типы данных;
- основные структуры данных;
- основные механизмы обработки информации в памяти компьютера;
- основные принципы процедурно-ориентированного программирования;
- основные парадигмы объектно-ориентированного программирования;

уметь:

- работать на персональном компьютере;
- пользоваться операционной системой;
- использовать возможности вычислительной техники и программного обеспечения;

владеть:

- первичными навыками и основными методами решения математических задач;
- навыками работы с учебной литературой;
- основными приёмами работы на компьютере с прикладным программным обеспечением;
- навыками составления простейших алгоритмов;
- навыками составления простейших компьютерных программ.

№ п/п	Последующие дисциплины	Код и наименование индикатора достижения компетенции
1	Информационное моделирование в строительстве (ТИМ)	ПК(Ц)-1.1, ПК(Ц)-1.2, ПК(Ц)-1.3, ПК(Ц)-1.4, ПК(Ц)-1.5

### 4. Объем дисциплины (модуля) в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся

Вид учебной работы	Всего часов	Из них часы на практическую подготовку	Семестр
			5
<b>Контактная работа</b>	32		32
Лабораторные занятия (Лаб)	32	0	32
<b>Иная контактная работа, в том числе:</b>			
консультации по курсовой работе (проекту), контрольным работам (РГР)			
контактная работа на аттестацию (сдача зачета, зачета с оценкой; защита курсовой работы (проекта); сдача контрольных работ (РГР))			
контактная работа на аттестацию в сессию (консультация перед экзаменом и сдача			
<b>Часы на контроль</b>	4		4
<b>Самостоятельная работа (СР)</b>	36		36
<b>Общая трудоемкость дисциплины (модуля)</b>			
<b>часы:</b>	72		72
<b>зачетные единицы:</b>	2		2

**5. Содержание дисциплины (модуля), структурированное по разделам (темам) с указанием отведенного на них количества академических часов и видов учебных занятий**

**5.1. Тематический план дисциплины (модуля)**

№	Разделы дисциплины	Семестр	Контактная работа (по учебным занятиям), час.						СР	Всего, час.	Код индикатора достижения компетенции
			лекции		ПЗ		ЛР				
			всего	из них на практическую подготовку	всего	из них на практическую подготовку	всего	из них на практическую			
1.	1 раздел. Основы программирования на языке Python										
1.1.	Основные типы данных, основные операторы. Преобразование типов данных. Основные встроенные функции. Организация консольного ввода/вывода данных разного типа.	5					2		3	5	ОПК-7.3
1.2.	Основные операции языка Python. Символьные вычисления. Построение графиков. Библиотека SymPy.	5					4		4	8	ОПК-7.3
1.3.	Работа со строками. Регулярные выражения.	5					4		4	8	ОПК-7.3
1.4.	Основные типы объектов языка Python.	5					4		4	8	ОПК-7.3
1.5.	Функции и модули. Обработка исключений	5					4		3,2	7,2	ОПК-7.3
1.6.	Объектно-ориентированное программирование на языке Python.	5					4		4,8	8,8	ОПК-7.3
1.7.	Работа с файлами и каталогами. Взаимодействие с операционной системой. Шифрование строк. Кодировка строк.	5					4		4	8	ОПК-7.3
1.8.	Разработка приложений с графическим пользовательским интерфейсом с помощью графической библиотеки Tk. Основные подходы.	5					2		3	5	ОПК-7.3
1.9.	Разработка приложений с графическим пользовательским интерфейсом с помощью графической библиотеки Tk. Основные виджеты.	5					2		3	5	ОПК-7.3

1.10	Создание графических изображений. Анимация.	5					2		3	5	ОПК-7. 3
2.	2 раздел. Контроль										
2.1.	Зачет	5								4	ОПК-7. 2

### 5.1. Лабораторные работы

№ разд	Наименование раздела и темы лабораторных работ	Наименование и содержание лабораторных работ									
1	Основные типы данных, основные операторы. Преобразование типов данных. Основные встроенные функции. Организация консольного ввода/вывода данных разного типа.	Основные типы данных, основные операторы. Преобразование типов данных. Основные встроенные функции. Организация консольного ввода/вывода данных разного типа. Основные типы данных, ключевые слова языка Python. Проверка типов. Преобразование типов данных. Основные операторы. Основные встроенные функции. Консольный ввод/вывод данных разного типа в Python.									
2	Основные операции языка Python. Символьные вычисления. Построение графиков. Библиотека SymPy.	Основные операции языка Python. Символьные вычисления. Построение графиков. Библиотека SymPy. Основные операции языка Python. Организация вычислений с помощью циклов и условных операций. Модуль math: работа с математическими функциями. Модуль random: генерация случайных чисел. Символьные вычисления в Python. Символьные переменные и символьные выражения. Вычисление символьных выражений. Графика в Python. Библиотека SymPy.									
3	Работа со строками. Регулярные выражения.	Работа со строками. Регулярные выражения. Создание строк. Управляющие последовательности. Форматирование строк. Основные функции для работы со строками. Регулярные выражения. Основные метасимволы. Основные комбинации символов. Основные функции и методы для поиска текстовых фрагментов.									
4	Основные типы объектов языка Python.	Основные типы объектов языка Python. Основные типы объектов языка Python. Списки. Кортежи. Множества. Словари. Основные функции для работы с объектами языка Python.									
5	Функции и модули. Обработка исключений	Функции и модули. Обработка исключений Пользовательские функции. Область видимости переменных. Подстановка аргументов. Возвращение значений. Вызов функции. Необязательные аргументы. Переменное число параметров функции. Возвращение значений. Использование обратного вызова. Анонимные функции. Функции-генераторы. Вложенные функции. Модули и пакеты. Инструкция import. Инструкция from. Пути поиска модулей. Повторная загрузка модулей. Обработка исключений. Инструкция try...except...else... finally. Инструкция with...as. Основные классы встроенных исключений. Пользовательские исключения.									

6	Объектно-ориентированное программирование на языке Python.	<p>Объектно-ориентированное программирование на языке Python. Классы и объекты в Python. Данные класса. Методы класса. Создание экземпляра класса. Доступ к атрибутам класса. Объекты в качестве аргументов функций. Объекты, возвращаемые функцией. Свойства класса. Декораторы классов.</p> <p>Наследование. Множественное наследование. Перегрузка операций. Статические методы. Абстрактные методы. Реализация полиморфизма в Python.</p> <p>Специальные методы. Перегрузка операторов. Статические методы. Абстрактные методы. Ограничение доступа к идентификаторам</p>
7	<p>Работа с файлами и каталогами.</p> <p>Взаимодействие с операционной системой.</p> <p>Шифрование строк.</p> <p>Кодировка строк.</p>	<p>Работа с файлами и каталогами. Взаимодействие с операционной системой. Шифрование строк. Кодировка строк.</p> <p>Работа с файлами. Открытие/закрытие файла. Методы для работы с файлами.</p> <p>Открытие файла. Методы для работы с файлами. Доступ к файлам с помощью модуля os. Перенаправление ввода/вывода. Сохранение объектов в файл. Закрытие файла.</p> <p>Взаимодействие с операционной системой. Работа с каталогами. Функции для манипулирования файлами. Получение информации от операционной системы.</p> <p>Шифрование строк. Основные методы шифрования строк.</p> <p>Кодировка строк. Основные методы кодировки строк.</p>
8	<p>Разработка приложений с графическим пользовательским интерфейсом с помощью графической библиотеки Tk.</p> <p>Основные подходы.</p>	<p>Разработка приложений с графическим пользовательским интерфейсом с помощью графической библиотеки Tk. Основные подходы.</p> <p>Основные принципы работы графических библиотек. Основные этапы создания оконного интерфейса. Размещение компонентов в окнах. Менеджеры размещения. Вывод текстовых сообщений с помощью виджетов. Использование стандартных диалоговых окон.</p>
9	<p>Разработка приложений с графическим пользовательским интерфейсом с помощью графической библиотеки Tk.</p> <p>Основные виджеты.</p>	<p>Разработка приложений с графическим пользовательским интерфейсом с помощью графической библиотеки Tk. Основные виджеты.</p> <p>Разработка графического интерфейса с помощью графической библиотеки Tk. Создание основных элементов управления: кнопок, флажков, переключателей, списков. Обработка событий.</p>
10	<p>Создание графических изображений.</p> <p>Анимация.</p>	<p>Создание графических изображений. Анимация.</p> <p>Создание графических изображений. Класс Canvas. Создание графических примитивов. Идентификаторы. Теги. Анимация. Добавление изображений из файла. Разработка приложения с графическим пользовательским интерфейсом.</p>

## 5.2. Самостоятельная работа обучающихся

№ разд	Наименование раздела дисциплины и темы	Содержание самостоятельной работы
1	<p>Основные типы данных, основные операторы.</p> <p>Преобразование типов данных. Основные встроенные функции.</p> <p>Организация консольного ввода/вывода данных разного типа.</p>	<p>Основные типы данных, основные операторы. Преобразование типов данных. Основные встроенные функции. Организация консольного ввода/вывода данных разного типа.</p> <p>Изучение материала, подготовка к лабораторным работам и тестированию.</p>
2	<p>Основные операции языка Python.</p> <p>Символьные вычисления.</p> <p>Построение графиков.</p> <p>Библиотека SymPy.</p>	<p>Основные операции языка Python. Символьные вычисления.</p> <p>Построение графиков. Библиотека SymPy.</p> <p>Изучение материала, подготовка к лабораторным работам и тестированию.</p>
3	<p>Работа со строками.</p> <p>Регулярные выражения.</p>	<p>Работа со строками. Регулярные выражения.</p> <p>Изучение материала, подготовка к лабораторным работам и тестированию.</p>
4	<p>Основные типы объектов языка Python.</p>	<p>Основные типы объектов языка Python.</p> <p>Изучение материала, подготовка к лабораторным работам и тестированию.</p>
5	<p>Функции и модули.</p> <p>Обработка исключений</p>	<p>Функции и модули. Обработка исключений</p> <p>Изучение материала, подготовка к лабораторным работам и тестированию.</p>
6	<p>Объектно-ориентированное программирование на языке Python.</p>	<p>Объектно-ориентированное программирование на языке Python.</p> <p>Изучение материала, подготовка к лабораторным работам и тестированию.</p>
7	<p>Работа с файлами и каталогами.</p> <p>Взаимодействие с операционной системой.</p> <p>Шифрование строк.</p> <p>Кодировка строк.</p>	<p>Работа с файлами и каталогами. Взаимодействие с операционной системой. Шифрование строк. Кодировка строк.</p> <p>Изучение материала, подготовка к лабораторным работам и тестированию.</p>
8	<p>Разработка приложений с графическим пользовательским интерфейсом с помощью графической библиотеки Tk.</p> <p>Основные подходы.</p>	<p>Разработка приложений с графическим пользовательским интерфейсом с помощью графической библиотеки Tk. Основные подходы.</p> <p>Изучение материала, подготовка к лабораторным работам и тестированию.</p>

9	<p>Разработка приложений с графическим пользовательским интерфейсом с помощью графической библиотеки Tk. Основные виджеты.</p>	<p>Разработка приложений с графическим пользовательским интерфейсом с помощью графической библиотеки Tk. Основные виджеты. Изучение материала, подготовка к лабораторным работам и тестированию.</p>
10	<p>Создание графических изображений. Анимация.</p>	<p>Создание графических изображений. Анимация. Изучение материала, подготовка к лабораторным работам и тестированию.</p>



## 6. Методические материалы для самостоятельной работы обучающихся по дисциплине (модулю)

Программой дисциплины предусмотрено проведение лабораторных работ, предполагающих закрепление изученного материала и формирование у обучающихся необходимых знаний, умений и навыков. Кроме того, важнейшим этапом изучения дисциплины является самостоятельная работа обучающихся с использованием всех средств и возможностей современных образовательных технологий.

В объем самостоятельной работы по дисциплине включается следующее:

- изучение теоретических вопросов по всем темам дисциплины;
- подготовка к лабораторным работам;
- подготовка к текущему контролю успеваемости студентов;
- подготовка к зачету.

Залогом успешного освоения дисциплины является обязательное посещение занятий, так как пропуск одного (тем более, нескольких) занятий может осложнить освоение разделов курса.

Приступая к изучению дисциплины, необходимо в первую очередь ознакомиться с содержанием РПД, а также методическими указаниями по организации самостоятельной работы.

В рамках самостоятельной работы по изучению дисциплины обучающимся необходимо:

- повторить законспектированный на лекционном занятии материал и дополнить его с учетом рекомендованной по данной теме литературы;
- при самостоятельном изучении теоретической темы сделать конспект, используя рекомендованные в РПД источники;
- выполнить практические задания в рамках изучаемой темы;
- ответить на контрольные вопросы по теме, используя материалы ФОС;
- подготовиться к тестированию, предусмотренному в контрольных точках;
- подготовиться к промежуточной аттестации.

При написании программ рекомендуется вначале написать программный код самостоятельно, а затем сравнить его с каким-либо примером. Такой подход позволяет студентам быстрее освоить методы, способы и стиль написания программ на языке Python и сократить время на его изучение.

Работы, выполняемые на практических занятиях, сдаются только лично на занятиях преподавателю, который ведет группу. Задания, выполняемые на компьютере, студенты сначала показывают только в электронном виде в среде разработки PyCharm или в интерактивной оболочке Shell. При необходимости, в присутствии преподавателя доделывают или исправляют ошибки. Если требуется распечатать выполненные работы и сдать их в бумажном виде, преподаватель говорит об этом на занятиях.

Итогом изучения дисциплины является зачет. Зачет проводится по расписанию. Форма проведения занятия может быть устная, письменная и в электронном виде. Студенты, не прошедшие аттестацию, должны ликвидировать задолженность в установленном порядке.

## 7. Оценочные материалы для проведения текущего контроля успеваемости и промежуточной аттестации обучающихся по дисциплине (модулю)

### 7.1. Перечень компетенций с указанием этапов их формирования в процессе освоения дисциплины

№ п/п	Контролируемые разделы дисциплины (модуля)	Код и наименование индикатора контролируемой компетенции	Вид оценочного средства
1	Основные типы данных, основные операторы. Преобразование типов данных. Основные встроенные функции. Организация консольного ввода/вывода данных разного типа.	ОПК-7.3	Практические задания, тесты.
2	Основные операции языка Python. Символьные вычисления. Построение графиков. Библиотека SymPy.	ОПК-7.3	Практические задания, тесты.

3	Работа со строками. Регулярные выражения.	ОПК-7.3	Практические задания, тесты.
4	Основные типы объектов языка Python.	ОПК-7.3	Практические задания, тесты.
5	Функции и модули. Обработка исключений	ОПК-7.3	Практические задания, тесты.
6	Объектно-ориентированное программирование на языке Python.	ОПК-7.3	Практические задания, тесты.
7	Работа с файлами и каталогами. Взаимодействие с операционной системой. Шифрование строк. Кодировка строк.	ОПК-7.3	Практические задания, тесты.
8	Разработка приложений с графическим пользовательским интерфейсом с помощью графической библиотеки Tk. Основные подходы.	ОПК-7.3	Практические задания, тесты.
9	Разработка приложений с графическим пользовательским интерфейсом с помощью графической библиотеки Tk. Основные виджеты.	ОПК-7.3	Практические задания, тесты.
10	Создание графических изображений. Анимация.	ОПК-7.3	Практические задания, тесты.
11	Зачет	ОПК-7.3	

7.2. Типовые контрольные задания или иные материалы текущего контроля успеваемости, необходимые для оценки знаний, умений и навыков и (или) опыта профессиональной деятельности, характеризующих этапы формирования компетенций в процессе освоения дисциплины

Комплект заданий для проверки сформированности индикаторов достижений компетенций  
ОПК-7.3

Тестовые и контрольные задания размещены в ЭИОС Moodle по адресу  
<https://moodle.spbgasu.ru/course/index.php?categoryid=222>

7.3. Система оценивания результатов обучения по дисциплине (модулю) при проведении текущего контроля успеваемости

<p>Оценка «отлично» (зачтено)</p>	<p>знания: - систематизированные, глубокие и полные знания по всем разделам дисциплины, а также по основным вопросам, выходящим за пределы учебной программы; - точное использование научной терминологии, систематически грамотное и логически правильное изложение ответа на вопросы; - полное и глубокое усвоение основной и дополнительной литературы, рекомендованной рабочей программой по дисциплине (модулю)</p> <p>умения: - умеет ориентироваться в теориях, концепциях и направлениях дисциплины и давать им критическую оценку, используя научные достижения других дисциплин</p> <p>навыки: - высокий уровень сформированности заявленных в рабочей программе компетенций; - владеет навыками самостоятельно и творчески решать сложные проблемы и нестандартные ситуации; - применяет теоретические знания для выбора методики выполнения заданий; - грамотно обосновывает ход решения задач; - безупречно владеет инструментарием учебной дисциплины, умение его эффективно использовать в постановке научных и практических задач; - творческая самостоятельная работа на практических/семинарских/лабораторных занятиях, активно участвует в групповых обсуждениях, высокий уровень культуры исполнения заданий</p>
<p>Оценка «хорошо» (зачтено)</p>	<p>знания: - достаточно полные и систематизированные знания по дисциплине; - усвоение основной и дополнительной литературы, рекомендованной рабочей программой по дисциплине (модулю)</p> <p>умения: - умеет ориентироваться в основных теориях, концепциях и направлениях дисциплины и давать им критическую оценку; - использует научную терминологию, лингвистически и логически правильно излагает ответы на вопросы, умеет делать обоснованные выводы; - владеет инструментарием по дисциплине, умение его использовать в постановке и решении научных и профессиональных задач</p> <p>навыки: - самостоятельная работа на практических занятиях, участие в групповых обсуждениях, высокий уровень культуры исполнения заданий; - средний уровень сформированности заявленных в рабочей программе компетенций; - без затруднений выбирает стандартную методику выполнения заданий; - обосновывает ход решения задач без затруднений</p>

<p>Оценка «удовлетворительно» (зачтено)</p>	<p>знания: - достаточный минимальный объем знаний по дисциплине; - усвоение основной литературы, рекомендованной рабочей программой; - использование научной терминологии, стилистическое и логическое изложение ответа на вопросы, умение делать выводы без существенных ошибок умения: - умеет ориентироваться в основных теориях, концепциях и направлениях по дисциплине и давать им оценку; - владеет инструментарием учебной дисциплины, умение его использовать в решении типовых задач; - умеет под руководством преподавателя решать стандартные задачи навыки: - работа под руководством преподавателя на практических занятиях, допустимый уровень культуры исполнения заданий; - достаточный минимальный уровень сформированности заявленных в рабочей программе компетенций; - испытывает затруднения при обосновании алгоритма выполнения заданий</p>
<p>Оценка «неудовлетворительно» (не зачтено)</p>	<p>знания: - фрагментарные знания по дисциплине; - отказ от ответа (выполнения письменной работы); - знание отдельных источников, рекомендованных рабочей программой по дисциплине; умения: - не умеет использовать научную терминологию; - наличие грубых ошибок навыки: - низкий уровень культуры исполнения заданий; - низкий уровень сформированности заявленных в рабочей программе компетенций; - отсутствие навыков самостоятельной работы; - не может обосновать алгоритм выполнения заданий</p>

7.4. Теоретические вопросы и практические задания для проведения промежуточной аттестации обучающихся, необходимые для оценки знаний, умений и навыков и (или) опыта профессиональной деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

7.4.1. Теоретические вопросы для проведения промежуточной аттестации обучающихся

Примерные вопросы

1. Основные типы данных в Python. Проверка типа данных. Преобразование типов данных.
2. Переменные. Именованное переменных. Присваивание значений переменным. Удаление переменных.
3. Операторы. Математические операторы. Двоичные операторы. Операторы присваивания.

Приоритет выполнения операторов.

4. Операторы сравнения. Оператор ветвления if...else.
5. Цикл for. Цикл while.
6. Оператор continue. Оператор break.
7. Функции range() и enumerate().
8. Встроенные функции и методы для работы с числами.
9. Математические функции. Модуль math.
10. Генерация случайных чисел. Модуль random.
11. Строки. Создание строк. Операции над строками. Форматирование строк.
12. Функции и методы для работы со строками. Метод format().
13. Функции для работы с символами.
14. Форматированный ввод/вывод данных.

15. Списки. Создание списка. Операции над списками.
16. Многомерные списки. Перебор элементов списка.
17. Генераторы списков и выражения-генераторы.
18. Функции `map()`, `zip()`, `filter` и `reduce()`. Примеры использования.
19. Добавление и удаление элементов списка. Поиск элемента в списке.
20. Переворачивание и перемешивание списка. Выбор элементов случайным образом.
21. Сортировка списка. Заполнение списка числами.
22. Использование списков как стеков.
23. Использование списков как очередей.
24. Кортежи.
25. Множества.
26. Диапазоны.
27. Словари. Создание словаря. Операции над словарями. Отличие словарей от кортежей.
28. Словари. Перебор элементов словаря. Методы работы со словарями. Генераторы словарей.
29. Пользовательские функции. Необязательные параметры и сопоставление по ключам.

Переменное число параметров функции.

30. Пользовательские функции. Определение и вызов функции. Расположение определения функции.

31. Анонимные функции. Функции-генераторы. Декораторы функций.

32. Глобальные и локальные переменные. Вложенные функции.

33. Модули. Создание модулей.

34. Модули. Поиск модулей.

35. Модули. Повторная загрузка модулей.

36. Модули. Компиляция модулей.

37. Модули. Стандартные модули Python.

38. Пакеты.

39. Определение класса и создание экземпляра класса.

40. Методы `__init__()` и `__del__()`.

41. Объявление наследования.

42. Множественное наследование: объявление, примеры реализации.

43. Специальные методы.

44. Перегрузка операторов.

45. Статические методы.

46. Абстрактные методы.

47. Обработка исключений. Инструкции `try...except...else...finally`.

48. Обработка исключений. Инструкция `with...as`.

49. Классы встроенных исключений

50. Пользовательские исключения.

51. Итераторы.

52. Контейнеры. Контейнеры-последовательности. Контейнеры-словари.

53. Перечисления.

54. Работа с файлами. Методы для работы с файлами.

55. Работа с файлами. Доступ к файлам с помощью модуля `os`.

56. Классы `StringIO` и `BytesIO`.

57. Основы создания приложений с графическим интерфейсом. Визуальное проектирование.

Событийное программирование.

58. Модуль `tkinter`. Основные элементы интерфейса.

59. Модуль `tkinter`. Простейшие элементы управления: кнопка, текстовое поле, поле для ввода.

60. Модуль `tkinter`. Функциональные элементы управления: поле со списком, флажок, переключатель.

61. Модуль `tkinter`. Класс `Canvas`. Основные графические примитивы. Создание анимации.

#### 7.4.2. Практические задания для проведения промежуточной аттестации обучающихся

Практические задания размещены в ЭИОС Moodle по адресу

<https://moodle.spbgasu.ru/course/index.php?categoryid=222>

#### 7.4.3. Примерные темы курсовой работы (проекта) (при наличии)

Курсовые проекты (работы) учебным планом не предусмотрены

#### 7.5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта профессиональной деятельности, характеризующие этапы формирования компетенций

Процедура проведения промежуточной аттестации и текущего контроля успеваемости регламентируется локальным нормативным актом, определяющим порядок организации и проведения текущего контроля успеваемости и промежуточной аттестации обучающихся.

Процедура оценивания формирования компетенций при проведении текущего контроля приведена в п. 7.3.

Типовые контрольные задания или иные материалы текущего контроля приведены в п. 7.2.

Промежуточная аттестация по дисциплине проводится в форме зачета.

#### 7.6. Критерии оценивания сформированности компетенций при проведении промежуточной аттестации

Критерии оценивания	Уровень освоения и оценка			
	Оценка «неудовлетворительно»	Оценка «удовлетворительно»	Оценка «хорошо»	Оценка «отлично»
	«не зачтено»	«зачтено»		
	Уровень освоения компетенции «недостаточный». Компетенции не сформированы. Знания отсутствуют, умения и навыки не сформированы	Уровень освоения компетенции «пороговый». Компетенции сформированы. Сформированы базовые структуры знаний. Умения фрагментарны и носят репродуктивный характер. Демонстрируется низкий уровень самостоятельности практического навыка.	Уровень освоения компетенции «продвинутый». Компетенции сформированы. Знания обширные, системные. Умения носят репродуктивный характер, применяются к решению типовых заданий. Демонстрируется достаточный уровень самостоятельности устойчивого практического навыка.	Уровень освоения компетенции «высокий». Компетенции сформированы. Знания аргументированные, всесторонние. Умения успешно применяются к решению как типовых, так и нестандартных творческих заданий. Демонстрируется высокий уровень самостоятельности, высокая адаптивность практического навыка

<p>знания</p>	<p>Обучающийся демонстрирует: -существенные пробелы в знаниях учебного материала; -допускаются принципиальные ошибки при ответе на основные вопросы билета, отсутствует знание и понимание основных понятий и категорий; -непонимание сущности дополнительных вопросов в рамках заданий билета.</p>	<p>Обучающийся демонстрирует: -знания теоретического материала; -неполные ответы на основные вопросы, ошибки в ответе, недостаточное понимание сущности излагаемых вопросов; -неуверенные и неточные ответы на дополнительные вопросы.</p>	<p>Обучающийся демонстрирует: -знание и понимание основных вопросов контролируемого объема программного материала; - знания теоретического материала -способность устанавливать и объяснять связь практики и теории, выявлять противоречия, проблемы и тенденции развития; -правильные и конкретные, без грубых ошибок, ответы на поставленные вопросы.</p>	<p>Обучающийся демонстрирует: -глубокие, всесторонние и аргументированные знания программного материала; -полное понимание сущности и взаимосвязи рассматриваемых процессов и явлений, точное знание основных понятий, в рамках обсуждаемых заданий; -способность устанавливать и объяснять связь практики и теории, -логически последовательные, содержательные, конкретные и исчерпывающие ответы на все задания билета, а также дополнительные вопросы экзаменатора.</p>
<p>умения</p>	<p>При выполнении практического задания билета обучающийся продемонстрировал недостаточный уровень умений. Практические задания не выполнены Обучающийся не отвечает на вопросы билета при дополнительных наводящих вопросах преподавателя.</p>	<p>Обучающийся выполнил практическое задание билета с существенными неточностями. Допускаются ошибки в содержании ответа и решении практических заданий. При ответах на дополнительные вопросы было допущено много неточностей.</p>	<p>Обучающийся выполнил практическое задание билета с небольшими неточностями. Показал хорошие умения в рамках освоенного учебного материала. Предложенные практические задания решены с небольшими неточностями. Ответил на большинство дополнительных вопросов.</p>	<p>Обучающийся правильно выполнил практическое задание билета. Показал отличные умения в рамках освоенного учебного материала. Решает предложенные практические задания без ошибок Ответил на все дополнительные вопросы.</p>

владение навыками	<p>Не может выбрать методику выполнения заданий.</p> <p>Допускает грубые ошибки при выполнении заданий, нарушающие логику решения задач.</p> <p>Делает некорректные выводы.</p> <p>Не может обосновать алгоритм выполнения заданий.</p>	<p>Испытывает затруднения по выбору методики выполнения заданий.</p> <p>Допускает ошибки при выполнении заданий, нарушения логики решения задач.</p> <p>Испытывает затруднения с формулированием корректных выводов.</p> <p>Испытывает затруднения при обосновании алгоритма выполнения заданий.</p>	<p>Без затруднений выбирает стандартную методику выполнения заданий.</p> <p>Допускает ошибки при выполнении заданий, не нарушающие логику решения задач</p> <p>Делает корректные выводы по результатам решения задачи.</p> <p>Обосновывает ход решения задач без затруднений.</p>	<p>Применяет теоретические знания для выбора методики выполнения заданий.</p> <p>Не допускает ошибок при выполнении заданий.</p> <p>Самостоятельно анализирует результаты выполнения заданий.</p> <p>Грамотно обосновывает ход решения задач.</p>
-------------------	---	--	---	---

Оценка по дисциплине зависит от уровня сформированности компетенций, закрепленных за дисциплиной, и представляет собой среднее арифметическое от выставленных оценок по отдельным результатам обучения (знания, умения, владение навыками).

Оценка «отлично»/«зачтено» выставляется, если среднее арифметическое находится в интервале от 4,5 до 5,0.

Оценка «хорошо»/«зачтено» выставляется, если среднее арифметическое находится в интервале от 3,5 до 4,4.

Оценка «удовлетворительно»/«зачтено» выставляется, если среднее арифметическое находится в интервале от 2,5 до 3,4.

Оценка «неудовлетворительно»/«не зачтено» выставляется, если среднее арифметическое находится в интервале от 0 до 2,4.

## 8. Учебно-методическое и материально-техническое обеспечение дисциплины (модуля)

### 8.1. Перечень основной и дополнительной литературы, необходимой для освоения дисциплины (модуля)

№ п/п	Автор, название, место издания, издательство, год издания учебной и учебно-методической литературы	Количество экземпляров/электронный адрес ЭБС
<b><u>Основная литература</u></b>		
1	Сузи Р. А., Язык программирования Python, Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016	<a href="http://www.iprbookshop.ru/52211.html">http://www.iprbookshop.ru/52211.html</a>
2	Уэс Маккинли, Слинкин А. А., Python и анализ данных, Саратов: Профобразование, 2017	<a href="http://www.iprbookshop.ru/64058.html">http://www.iprbookshop.ru/64058.html</a>
<b><u>Дополнительная литература</u></b>		
1	Сузи Р. А., Язык программирования Python, Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020	<a href="https://www.iprbookshop.ru/97589.html">https://www.iprbookshop.ru/97589.html</a>
2	Шелудько В. М., Язык программирования высокого уровня Python. Функции, структуры данных, дополнительные модули, Ростов-на-Дону, Таганрог: Издательство Южного федерального университета, 2017	<a href="https://www.iprbookshop.ru/87530.html">https://www.iprbookshop.ru/87530.html</a>



3	Букунов С. В., Букунова О. В., Основы программирования на языке Python, Санкт-Петербург: СПбГАСУ, 2019	<a href="http://ntb.spbgasu.ru/elib/01173/">http://ntb.spbgasu.ru/elib/01173/</a>
4	Буйначев С. К., Боклаг Н. Ю., Песин Ю. В., Основы программирования на языке Python, Екатеринбург: Уральский федеральный университет, ЭБС АСВ, 2014	<a href="http://www.iprbookshop.ru/66183.html">http://www.iprbookshop.ru/66183.html</a>

Обучающиеся из числа инвалидов и лиц с ОВЗ обеспечиваются печатными и (или) электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья.

#### 8.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля)

Наименование ресурса сети «Интернет»	Электронный адрес ресурса
Сайт, содержащий в свободном доступе необходимые дистрибутивы, полную информацию и уроки по программированию на языке Python	<a href="https://www.python.org/">https://www.python.org/</a>
Сайт, посвященный свободно распространяемому пакету sympy, представляющему собой библиотеку Python символьных вычислений.	<a href="https://www.sympy.org/en/index.html">https://www.sympy.org/en/index.html</a>
Сайт для скачивания графической библиотеки PyQt.	<a href="https://riverbankcomputing.com">https://riverbankcomputing.com</a>

#### 8.3. Перечень современных профессиональных баз данных и информационных справочных систем

Наименование	Электронный адрес ресурса
Электронно-библиотечная система издательства "Лань"	<a href="https://e.lanbook.com/">https://e.lanbook.com/</a>
Электронно-библиотечная система издательства "ЮРАЙТ"	<a href="https://www.biblio-online.ru/">https://www.biblio-online.ru/</a>
Электронно-библиотечная система издательства "IPRsmart"	<a href="http://www.iprbookshop.ru/">http://www.iprbookshop.ru/</a>
Научная электронная библиотека eLIBRARY.RU	Научная электронная библиотека eLIBRARY.RU
Система дистанционного обучения СПбГАСУ Moodle	<a href="https://moodle.spbgasu.ru/">https://moodle.spbgasu.ru/</a>

#### 8.4. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного и свободно распространяемого программного обеспечения

Наименование	Способ распространения (лицензионное или свободно распространяемое)
Python версия 3.7.6386.10	Свободно распространяемое
PyCharm Community	Свободно распространяемое

#### 8.5. Материально-техническое обеспечение дисциплины

Сведения об оснащённости учебных аудиторий и помещений для самостоятельной работы

Наименование учебных аудиторий и помещений для самостоятельной работы	Оснащённость оборудованием и техническими средствами обучения
72. Компьютерный класс	Рабочие места с ПК (стол компьютерный, системный блок, монитор, клавиатура, мышь), стол рабочий, подключение к компьютерной сети СПбГАСУ, выход в Internet.
72. Помещения для самостоятельной работы	Помещение для самостоятельной работы (читальный зал библиотеки, ауд. 217): ПК-23 шт., в т.ч. 1 шт.- ПК для лиц с ОВЗ (системный блок, монитор, клавиатура, мышь) с подключением к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду СПбГАСУ. ПО Microsoft Windows 10

<p>72. Учебные аудитории для проведения практических занятий, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации</p>	<p>Учебная аудитория для проведения практических занятий, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации – комплект мультимедийного оборудования (персональный компьютер, мультимедийный проектор, экран, аудиосистема), доска маркерная белая эмалевая, комплект учебной мебели, подключение к компьютерной сети СПбГАСУ, выход в Интернет.</p>
---	---

Для инвалидов и лиц с ОВЗ обеспечиваются специальные условия для получения образования в соответствии с требованиями нормативно-правовых документов.