

**Федеральное государственное автономное образовательное учреждение
высшего образования
«Санкт-Петербургский политехнический университет Петра Великого»**

УТВЕРЖДАЮ
Директор ИБСиБ
_____ А.В. Васин
«30» мая 2025 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

«Программирование Python»

Разработчик	Высшая школа биомедицинских систем и технологий
Направление (специальность) подготовки	06.05.01 Биоинженерия и биоинформатика
Наименование ООП	06.05.01_01 Биоинженерия и биоинформатика
Квалификация (степень) выпускника	биоинженер и биоинформатик
Образовательный стандарт	СУОС
Форма обучения	Очная

СОГЛАСОВАНО

Руководитель ОП

_____ Д.И. Богомаз

«15» апреля 2025 г.

Соответствует СУОС

Утверждена протоколом заседания
высшей школы "ВШБСиТ"

от «15» апреля 2025 г. № №6

РПД разработали:

Доцент, к.б.н. Д.И. Богомаз

Доцент, к.ф.-м.н. Е.А. Скребенков

1. Цели и планируемые результаты изучения дисциплины

Цели освоения дисциплины

1. Знать основы и уметь применять на практике: объектно-ориентированное программирование.
2. Знать основы и уметь применять на практике: обработка исключений.
3. Знать основы и уметь применять на практике: элементы функционального программирования в python.
4. Знать основы и уметь применять на практике: регулярные выражения, итераторы, декораторы.

Результаты обучения выпускника

Код	Результат обучения (компетенция) выпускника ООП
ОПК-6	Способен создавать компьютерные программы, используемые в биоинформатике и биоинженерии
ИД-3 ОПК-6	Создает компьютерные программы на языке Питон, используемые в биоинформатике и биоинженерии

Планируемые результаты изучения дисциплины

знания:

- Знание способов создания компьютерных программ на языке Питон, используемые в биоинформатике и биоинженерии

умения:

- Умение создавать компьютерные программы на языке Питон, используемые в биоинформатике и биоинженерии

навыки:

- Владение навыками создания компьютерных программы на языке Питон, используемые в биоинформатике и биоинженерии

2. Место дисциплины в структуре ООП

В учебном плане дисциплина «Программирование Python» относится к модулю «Модуль цифровых компетенций (Digital)».

Изучение дисциплины базируется на результатах освоения следующих дисциплин:

- Цифровая грамотность
- Цифровой практикум

3. Распределение трудоёмкости освоения дисциплины по видам учебной работы и формы текущего контроля и промежуточной аттестации

3.1. Виды учебной работы

Виды учебной работы	Трудоёмкость по семестрам
	Очная форма
Лабораторные занятия	30
Самостоятельная работа	25
Часы на контроль	5
Промежуточная аттестация (зачет)	4
Промежуточная аттестация (курсовая работа, проект)	4
Курсовое проектирование	4
Общая трудоёмкость освоения дисциплины	72, ач
	2, зет

3.2. Формы текущего контроля и промежуточной аттестации

Формы текущего контроля и промежуточной аттестации	Количество по семестрам
	Очная форма
Текущий контроль	
Курсовые работы, шт.	1
Промежуточная аттестация	
Зачеты, шт.	1

4. Содержание и результаты обучения

4.1 Разделы дисциплины и виды учебной работы

№ раздела	Разделы дисциплины, мероприятия текущего контроля	Очная форма	
		Лаб, ач	СР, ач
1.	Функции. Декораторы.	6	4
2.	Объектно-ориентированное программирование.	8	6

3.	Обработка исключений.	3	2
4.	Элементы функционального программирования в python.	3	3
5.	Регулярные выражения.	2	2
6.	Итераторы, генераторы.	4	3
7.	Модули, пакеты, PyPI.	4	5
Итого по видам учебной работы:		30	25
Зачеты, ач			5
Часы на контроль, ач			5
Курсовое проектирование		4	
Промежуточная аттестация (зачет)		4	
Промежуточная аттестация (курсовая работа, проект)		4	
Общая трудоёмкость освоения: ач / зет		72 / 2	

4.2. Содержание разделов и результаты изучения дисциплины

Раздел дисциплины	Содержание
1. Функции. Декораторы.	Повторение: определение и вызов функций, аргументы функций: позиционные, именованные, аргументы по умолчанию. Области видимости переменных. Возврат значений из функций. Рекурсивные функции. Лямбда-функции. Декораторы: назначение, принцип работы, создание и использование.
2. Объектно-ориентированное программирование.	Основные принципы ООП: абстракция, инкапсуляция, наследование, полиморфизм. Классы и объекты. Конструкторы и деструкторы. Методы класса и статические методы. Наследование и полиморфизм. Абстрактные классы и интерфейсы.
3. Обработка исключений.	Понятие исключения и ошибки в программировании. Типы исключений в Python. Обработка исключений с помощью блоков try-except. Обработка нескольких типов исключений. Использование блоков else и finally в конструкции try-except. Генерация собственных исключений с помощью оператора raise. Создание пользовательских типов исключений.
4. Элементы функционального программирования в python.	Основные концепции функционального программирования. Функции высшего порядка: map, filter, reduce. Лямбда-функции в контексте функционального программирования. Списковые выражения и генераторы списков.
5. Регулярные выражения.	Синтаксис регулярных выражений. Основные метасимволы и квантификаторы. Использование модуля re для работы с регулярными выражениями. Поиск и замена текста с помощью регулярных выражений.
6. Итераторы, генераторы.	Понятие итератора и итерируемого объекта. Создание собственных итераторов с помощью протокола итерации. Использование функций iter и next. Генераторы как способ создания итераторов.
7. Модули, пакеты, PyPI.	Создание и подключение модулей. Пространства имён и области видимости. Работа с пакетами. Обзор стандартной библиотеки Python. Установка и использование сторонних библиотек.

5. Образовательные технологии

1. Интерактивная лекция. Теоретический материал преподносится на примере блоков кода, который обучающийся может изменять и запускать.
2. Практическая работа. Упражнения для закрепления нового синтаксиса. Задачи для практического освоения новых концептов. Лабораторные для понимания контекста использования нового материала.

6. Лабораторный практикум

№ раздела	Наименование лабораторных работ	Трудоемкость, ач
		Очная форма
1.	Функции, декораторы.	6
2.	Объектно-ориентированное программирование.	8
3.	Генераторы и итераторы.	6
4.	Создание собственного пакета.	10
Итого часов		30

7. Практические занятия

Не предусмотрено

8. Организация и учебно-методическое обеспечение самостоятельной работы

Примерное распределение времени самостоятельной работы студентов

Вид самостоятельной работы	Примерная трудоемкость, ач
	Очная форма
Текущая СР	
работа с лекционным материалом, с учебной литературой	8
опережающая самостоятельная работа (изучение нового материала до его изложения на занятиях)	0
самостоятельное изучение разделов дисциплины	0
выполнение домашних заданий, домашних контрольных работ	17
подготовка к лабораторным работам, к практическим и семинарским занятиям	0
подготовка к контрольным работам, коллоквиумам	0
Итого текущей СР:	25
Творческая проблемно-ориентированная СР	
выполнение расчётно-графических работ	0
выполнение курсового проекта или курсовой работы	0
поиск, изучение и презентация информации по заданной проблеме, анализ научных публикаций по заданной теме	0
работа над междисциплинарным проектом	0
исследовательская работа, участие в конференциях, семинарах, олимпиадах	0
анализ данных по заданной теме, выполнение расчётов, составление схем и моделей на основе собранных данных	0
Итого творческой СР:	0
Общая трудоемкость СР:	25

9. Учебно-методическое обеспечение дисциплины

9.1. Адрес сайта курса

<https://dl-ibmst.spbstu.ru/>

9.2. Рекомендуемая литература

Основная литература

№	Автор, название, место издания, издательство, год (годы) издания	Год изд.	Источник
1	Бизли Д.М. Язык программирования Python: Киев: ДияСофт, 2000.	2000	ИБК СПбПУ

Дополнительная литература

№	Автор, название, место издания, издательство, год (годы) издания	Год изд.	Источник
1	Bird A. и др. The Python workshop: Birmingham: Packt Publishing, 2019. URL: http://elib.spbstu.ru/dl/2/eb23-123.pdf	2019	ЭБ СПбПУ

Ресурсы Интернета

1. Real python: <https://realpython.com/>

9.3. Технические средства обеспечения дисциплины

1. Microsoft:

WorkStation + Office Pro Plus

Office 365 ProPlus Enrollment ID: 58313261

Parent Program: 75434048

Сублицензионный договор с ООО «СОФТЛАЙН ПРОЕКТЫ» от 23.10.2017 № 180/17-Д

2. Программа «Защита образования» компании «Лаборатория Касперского» Соглашение № 1CE0151102071341

Договор на оказание услуг по продлению техподдержки бессрочных академических лицензий с ООО «ПОЛИКОМ ПРО» от 23.10.2017 № 182/17-Д

3. Программное обеспечение «Антиплагиат.ВУЗ» Лицензионный договор с ЗАО «Анти-Плагат» от 26.03.2018 № 170

10. Материально-техническое обеспечение дисциплины

1. Microsoft:

WorkStation + Office Pro Plus

Office 365 ProPlus Enrollment ID: 58313261

Parent Program: 75434048

Сублицензионный договор с ООО «СОФТЛАЙН ПРОЕКТЫ» от 23.10.2017 № 180/17-Д

2. Программа «Защита образования» компании «Лаборатория Касперского» Соглашение № 1CE0151102071341

Договор на оказание услуг по продлению техподдержки бессрочных академических лицензий с ООО «ПОЛИКОМ ПРО» от 23.10.2017 № 182/17-Д

3. Программное обеспечение «Антиплагиат.ВУЗ» Лицензионный договор с ЗАО «Анти-Плагиат» от 26.03.2018 № 170

11. Критерии оценивания и оценочные средства

11.1. Критерии оценивания

Для дисциплины «Программирование Python» формой аттестации является зачёт. Оценивание качества освоения дисциплины производится с использованием рейтинговой системы.

Зачёт

Для получения зачёта необходимо набрать минимум 50 баллов из 100.

Зачёт выставляется в результате сдачи и защиты всех практических заданий, выполняемых в течение прохождения курса.

11.2. Оценочные средства

Оценочные средства по дисциплине представлены в фонде оценочных средств, который является резервной частью основной образовательной программы и размещается в электронной информационно-образовательной среде СПбПУ на портале etk.spbstu.ru

12. Методические рекомендации по организации изучения дисциплины

Программа курса нацелена на ознакомление студентов с основами программирования, нацеленного на решение научных задач. Кроме ознакомления с основами программирования ставится целью развитие общего кругозора, связанного с современными цифровыми технологиями с пониманием общих принципов их устройства.

Наибольшее значение при освоении данной дисциплины имеет самостоятельная работа студентов над решением учебных проблем, основанных на задачах, которые решаются в современных биоинформатических исследованиях. Также отводится время для решения практических задач, связанных с обработкой экспериментальных данных и их визуализацией, и на ознакомление с возможностями использовать программирование для моделирования естественнонаучных процессов.

Преподаватель в рамках курса даёт общие и стартовые знания для каждой темы, а затем, ставя перед студентами задачи, направляет и помогает студентам найти оптимальные пути решения. Студентам допускается использовать справочный материал, например, знакомится с документацией языка или используемой библиотеки.

13. Адаптация рабочей программы для лиц с ОВЗ

Адаптированная программа разрабатывается при наличии заявления со стороны обучающегося (родителей, законных представителей) и медицинских показаний (рекомендациями психолого-медико-педагогической комиссии). Для инвалидов адаптированная образовательная программа разрабатывается в соответствии с индивидуальной программой реабилитации.