

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ

Ректор Белорусского
государственного университета

А.Д.Король

27 июня 2025 г.

Регистрационный № 3274/б.



ЯЗЫК ПРОГРАММИРОВАНИЯ PYTHON

Учебная программа для учреждения образования по учебной дисциплине для
специальности

6-05-0533-07 Математика и компьютерные науки

Профилизация:

Веб-программирование и интернет-технологии

2025 г.

Учебная программа составлена на основе ОСВО 6-05-0533-07-2023 и учебных планов БГУ № 6-5.4-55/03 от 15.05.2023, № 6-5.4-55/13и. от 31.05.2023, № 6-5.4-55/23з. от 31.05.2023.

СОСТАВИТЕЛЬ:

Игорь Михайлович Галкин, доцент кафедры веб-технологий и компьютерного моделирования механико-математического факультета Белорусского государственного университета, кандидат физико-математических наук, доцент

РЕЦЕНЗЕНТ:

О.Г.Смолякова, доцент кафедры программного обеспечения информационных технологий факультета компьютерных систем и сетей Белорусского государственного университета информатики и радиоэлектроники, кандидат технических наук

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

Кафедрой веб-технологий и компьютерного моделирования БГУ
(протокол № 17 от 02.06.2025)

Научно-методическим Советом БГУ
(протокол № 11 от 26.06.2025)

Заведующий кафедрой



М.В.Игнатенко

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Цели и задачи учебной дисциплины

Учебная дисциплина "Язык программирования Python" продолжает формировать и развивать у студентов основы алгоритмического мышления и способствует их обучению современным методам решения задач с помощью компьютера.

Данная дисциплина ориентирована на студентов высших учебных заведений, обучающихся по специальности 6-05-0533-07 "Математика и компьютерные науки" очной и заочной форм получения высшего образования профилизации "Веб-программирование и интернет-технологии". Дисциплина изучается в 5-м семестре очной формы и в 7-м семестре заочной формы получения высшего образования, что позволяет расширить ранее полученные знания в программировании.

В качестве изучаемого языка программирования выбран распространенный язык Python, позволяющий осваивать классические приемы и современные технологии программирования.

Целью дисциплины является формирование у студентов навыков программирования на одном из самых популярных и востребованных языков программирования, а также получение знаний и практических умений, необходимых для решения широкого спектра задач в области математики и информационных технологий.

Задачами дисциплины являются:

- изучение синтаксиса и основных конструкций языка Python;
- приобретение навыков работы с библиотеками для выполнения математических и статистических расчетов;
- практическое применение Python для решения математических задач, анализа данных и автоматизации рутинных операций.

Место учебной дисциплины в системе подготовки специалиста с высшим образованием.

Учебная дисциплина относится к модулю **«Математическая информатика»** компонента учреждения образования, который является модулем по выбору (1 из 2).

Связи с другими учебными дисциплинами, включая учебные дисциплины компонента учреждения высшего образования, дисциплины специализации и др. Учебная дисциплина "Язык программирования Python" опирается на знания, полученные при изучении дисциплин "Методы программирования" и "Технологии программирования". Изучение дисциплины "Язык программирования Python" и характер решаемых задач должны способствовать расширению у студентов представления о современных подходах к программному решению научных и прикладных задач, что необходимо для дальнейшего освоения специализированных дисциплин, в частности "Обработка больших данных".

Требования к компетенциям

Освоение учебной дисциплины "Язык программирования Python" должно обеспечить формирование следующих компетенций:

специализированные компетенции:

Разрабатывать информационные решения для поиска и извлечения данных, ставить и тестировать прикладные гипотезы на основе данных, разрабатывать специализированные модели машинного обучения и алгоритмы анализа данных, развертывать модели в том числе в облачной среде, осуществлять визуализацию и бизнес-анализ полученных решений.

В результате изучения данной дисциплины студент должен

знать:

- основные концепции и принципы языка программирования Python;
- назначение и функции базовых модулей Python;
- возможности и способы использования библиотек Python;

уметь:

– использовать базовые типы данных и функции Python;

– использовать стандартные библиотеки Python для решения различных задач программирования.

– применять библиотеки для научных вычислений и анализа данных, такие как NumPy и Pandas;

– самостоятельно изучать новые библиотеки и инструменты, расширяя свои навыки программирования в соответствии с требованиями современного рынка;

– применять полученные знания для решения конкретных математических задач, анализа данных и автоматизации вычислительных процессов;

иметь навык:

– использования языка Python для решения задач в различных областях.

Структура учебной дисциплины

Дисциплина изучается в 5-м семестре. В соответствии с учебным планом всего на изучение учебной дисциплины «Язык программирования Python» отведено для **очной формы** получения высшего образования – 90 часов, в том числе 36 аудиторных часов: лекции – 18 часов, лабораторные занятия – 16 часов, управляемая самостоятельная работа – 2 часа.

Трудоемкость учебной дисциплины составляет 3 зачетные единицы.

Форма промежуточной аттестации – зачет.

Дисциплина изучается в 7-м семестре. В соответствии с учебным планом всего на изучение учебной дисциплины «Язык программирования Python» отведено для **заочной формы** получения высшего образования – 90 часов, в том числе 8 аудиторных часов: лекции – 4 часа, лабораторные занятия – 4 часа.

Трудоемкость учебной дисциплины составляет 3 зачетные единицы.

Форма промежуточной аттестации – зачет.

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

Тема 1.	<i>Введение в Python.</i> История и философия Python. Особенности языка и его применение в науке и индустрии. Установка Python и настройка среды разработки. Основы синтаксиса: переменные, выражения, ввод и вывод данных.
Тема 2.	<i>Основные структуры данных.</i> Типы данных: числа, строки. Структуры (коллекции) данных: списки, кортежи. Операции над структурами данных: индексация, распаковка и упаковка, срезы, методы. Понятие изменяемости и неизменяемости данных.
Тема 3.	<i>Управляющие конструкции.</i> Условный оператор if. Циклы: for, while. Вложенные циклы и условия. Понятие итерируемых объектов.
Тема 4.	<i>Функции и модули.</i> Определение функций и возвращение значений. Аргументы функций и их типы. Области видимости переменных. Импорт модулей и использование стандартной библиотеки.
Тема 5.	<i>Объектно-ориентированное программирование (ООП).</i> Классы и объекты. Конструкторы и деструкторы. Наследование и композиция. Инкапсуляция, полиморфизм и абстракция.
Тема 6.	<i>Исключения и обработка ошибок.</i> Понятие исключений и их иерархия. Блоки try, except, finally. Создание собственных исключений. Рекомендации по обработке исключений.
Тема 7.	<i>Работа с файлами и внешними ресурсами.</i> Чтение и запись файлов. Работа с файловой системой. Использование контекстных менеджеров. Взаимодействие с внешними API.
Тема 8.	<i>Библиотеки для научных вычислений и анализа данных.</i> Обзор библиотеки NumPy. Основы работы с массивами и матрицами. Введение в библиотеку Pandas. Основы анализа и обработки данных с помощью Pandas.
Тема 9.	<i>Визуализация данных и введение в машинное обучение.</i> Обзор библиотеки Matplotlib. Основы построения графиков и диаграмм. Введение в библиотеку Seaborn для статистической визуализации. Краткий обзор машинного обучения с использованием Python.

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ

Очная форма получения высшего образования с применением дистанционных образовательных технологий (ДОТ)

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов				Количество часов УСП	Формы контроля знаний
		Лекции	Практические занятия	Лабораторные занятия	Иное		
1	Введение в Python	2					Опрос
2	Основные структуры данных	2		2			Лабораторные работы
3	Управляющие конструкции	2		2			Лабораторные работы, контрольная работа
4	Функции и модули	2		2			Лабораторные работы, контрольная работа
5	Объектно-ориентированное программирование (ООП)	2		2		1	Лабораторные работы и задания для УСП, контрольная работа
6	Исключения и обработка ошибок	2		2			Лабораторные работы
7	Работа с файлами и внешними ресурсами	2		2			Лабораторные работы, контрольная работа
8	Библиотеки для научных вычислений и анализа данных	2		2		1	Лабораторные работы и задания для УСП
9	Визуализация данных и введение в машинное обучение	2		2			Лабораторные работы
	ВСЕГО ЧАСОВ	18		16		2	

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ

Заочная форма получения высшего образования

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов				Форма контроля
		Лекции	Практические занятия	Лабораторные занятия	Иное	
1	2	3	4	5	6	7
2	Основные структуры данных	2		2		Лабораторные работы
5	Объектно-ориентированное программирование (ООП)	2		2		Лабораторные работы
	ВСЕГО ЧАСОВ	4		4		

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

Основная литература

1. Бизли, Д. Python. Исчерпывающее руководство / Дэвид Бизли ; [пер. с англ. Е. Матвеев]. – Санкт-Петербург ; Москва ; Минск : Питер, 2023. – 366 с. – URL: <https://ibooks.ru/bookshelf/386795/reading>.
2. Бейдер, Д. Знакомство с Python / Дэн Бейдер и др. ; [пер. с англ. Е. Матвеев]. – Санкт-Петербург ; Москва ; Минск : Питер, 2023. – 507 с. – URL: <https://ibooks.ru/bookshelf/387727/reading>.
3. Такфилд, Б. Алгоритмы неформально. Инструкция для начинающих питонистов / Б. Такфилд ; [пер. с англ. Е. Матвеев]. – Санкт-Петербург ; Москва ; Минск : Питер, 2022. – 269 с.
4. Хиллард, Д. Секреты Python Pro / Д. Хиллард ; [пер. с англ. А. Логунов]. – Санкт-Петербург [и др.] : Питер, 2021. – 317 с. – URL: <https://ibooks.ru/bookshelf/374462>

Дополнительная литература

5. Python documentation [Электронный ресурс] – Электронные данные. – Режим доступа: <https://www.python.org/doc/>
6. Python developers guide [Электронный ресурс] – Электронные данные. – Режим доступа: <https://devguide.python.org/>
7. Python tutorial [Электронный ресурс] – Электронные данные. – Режим доступа: <https://docs.python.org/3/tutorial/>
8. PEP index [Электронный ресурс] – Электронные данные. – Режим доступа: <https://www.python.org/dev/peps/>
9. Awesome Python (free python resources index) [Электронный ресурс] – Электронные данные. – Режим доступа: <https://github.com/vinta/awesome-python/>

Перечень рекомендуемых средств диагностики и методика формирования итоговой отметки

Диагностика результатов учебной деятельности по учебной дисциплине проводится преподавателем, как правило, во время аудиторных занятий. Для диагностики используются:

- устный опрос;
- выполнение лабораторных работ и заданий для УСР;
- выполнение контрольных работ.

Формой промежуточной аттестации по дисциплине учебным планом предусмотрен зачет.

Примерный перечень заданий для управляемой самостоятельной работы студентов

Тема 5. Объектно-ориентированное программирование (ООП) (1 час)

Разработать на Python программу с использованием ООП и нескольких объектов разных классов.

Форма контроля – выполнение задания.

Тема 8. Библиотеки для научных вычислений и анализа данных (1 ч)

Выполнить разносторонний анализ существующего набора данных с использованием библиотек NumPy и Pandas.

Форма контроля – выполнение задания.

Примерная тематика лабораторных занятий

Задания для лабораторных и самостоятельных работ формируются на основании последних изученных во время лекций тем. Основное назначение лабораторных работ – получить навыки практического использования изученного материала.

Занятие 1. Основные структуры данных

Занятие 2. Управляющие конструкции

Занятие 3. Функции и модули

Занятие 4. Объектно-ориентированное программирование (ООП)

Занятие 5. Исключения и обработка ошибок

Занятие 6. Работа с файлами и внешними ресурсами

Занятие 7. Библиотеки для научных вычислений и анализа данных

Занятие 8. Визуализация данных и введение в машинное обучение

Описание инновационных подходов и методов к преподаванию учебной дисциплины

При организации образовательного процесса используются:

– *методы и приемы развития критического мышления*, которые представляют собой систему, формирующую навыки работы с информацией в

процессе чтения и письма; понимании информации как отправного, а не конечного пункта критического мышления;

– **практико-ориентированный подход**, который предполагает освоение содержание образования через решения практических задач.

Методические рекомендации по организации самостоятельной работы обучающихся

При изучении учебной дисциплины рекомендуется использовать следующие формы самостоятельной работы:

– изучение основной и дополнительной литературы по учебной дисциплине;

– изучение материалов электронных источников по вопросам учебной дисциплины, в том числе размещенных в электронной библиотеке БГУ elib.bsu.by;

– подготовка к лабораторным занятиям и контрольным работам;

– выполнение лабораторных работ.

Для организации самостоятельной работы студентов по учебной дисциплине рекомендуется использовать возможности образовательного портала <https://edummf.bsu.by> и системы управления обучением Moodle для размещения текущей организационной информации и учебно-программных материалов, в том числе вопросов к зачету, а также контроля выполнения заданий.

Примерный перечень вопросов к зачету

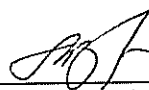
1. Основные концепции и характеристика языка.
2. Способы (режимы) разработки и выполнения программ.
3. Основы синтаксиса, ввод и вывод данных.
4. Переменные и их использование.
5. Общие принципы управления памятью.
6. Основные типы данных.
7. Изменяемые и неизменяемые данные.
8. Типизация в Python.
9. Основные операции и их приоритет.
10. Индексация в данных, представляющих коллекции.
11. Управляющие конструкции (операторы).
12. Логические операции, истинные и ложные значения.
13. Тернарная операция.
14. Операция присваивания.
15. Условный оператор if.
16. Операторы while и for.
17. Блок else в операторах цикла.
18. Определение и использование функций.
19. Параметры и возвращаемые значения функций.

20. Назначение и использование модулей.
21. Классы и объекты.
22. Реализация основных принципов ООП.
23. Обработка исключений.
24. Работа с файлами.
25. Упаковка и распаковка данных.
26. Срезы.
27. Назначение и возможности библиотек NumPy и Pandas
28. Назначение и возможности библиотек Matplotlib и Seaborn.

ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ УО

Название учебной дисциплины, с которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы учреждения образования по учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу (с указанием даты и номера протокола)
Построение и анализ алгоритмов	Веб-технологий и компьютерного моделирования	Предложения отсутствуют	Рекомендовать к утверждению учебную программу (протокол № 17 от 02.06.2025)
Системная архитектура и управление проектами	Веб-технологий и компьютерного моделирования	Предложения отсутствуют	Рекомендовать к утверждению учебную программу (протокол № 17 от 02.06.2025)

Заведующий кафедрой веб-технологий
и компьютерного моделирования,
кандидат физ.-мат. наук, доцент


(подпись)

М.В.Игнатенко

02.06.2025

ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ К УЧЕБНОЙ ПРОГРАММЕ УО
на ____/____ учебный год

№ п/п	Дополнения и изменения	Основание

Учебная программа пересмотрена и одобрена на заседании кафедры
_____ (протокол № ____ от _____ 202_ г.)

Заведующий кафедрой

УТВЕРЖДАЮ
Декан факультета
