

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**федеральное государственное автономное  
образовательное учреждение высшего образования\_  
«Национальный исследовательский Нижегородский государственный университет  
им. Н.И. Лобачевского»**

Институт информационных технологий, математики и механики

---

УТВЕРЖДЕНО

решением президиума Ученого совета ННГУ

протокол № 1 от 16.01.2024 г.

**Рабочая программа дисциплины**

Язык Python

---

Уровень высшего образования

Магистратура

---

Направление подготовки / специальность

01.04.02 - Прикладная математика и информатика

---

Направленность образовательной программы

Анализ данных в прикладных областях

---

Форма обучения

очная

---

г. Нижний Новгород

2024 год начала подготовки

## 1. Место дисциплины в структуре ОПОП

Дисциплина Б1.В.ДВ.01.02 Язык Python относится к части, формируемой участниками образовательных отношений образовательной программы.

## 2. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства	
	Индикатор достижения компетенции (код, содержание индикатора)	Результаты обучения по дисциплине	Для текущего контроля успеваемости	Для промежуточной аттестации
ПК-12: Способен разрабатывать и применять математические методы, системное и прикладное программное обеспечение для решения задач производственно-технологической деятельности	<p>ПК-12.1: Знает основные методы разработки математических методов, системного и прикладного программного обеспечения для решения задач производственно-технологической деятельности</p> <p>ПК-12.2: Умеет оценивать трудоемкость разработки программных средств для решения задач производственно-технологической деятельности</p> <p>ПК-12.3: Имеет навыки разработки системного программного обеспечения для решения задач производственно-технологической деятельности</p>	<p>ПК-12.1: ЗНАТЬ Методику разработки программ на языке Python, анализа их эффективности и подходы к ее повышению для решения задач производственно-технологической деятельности</p> <p>ПК-12.2: УМЕТЬ Применять на практике технологии Scipy, Numpy, Pandas, Matplotlib для разработки программ для решения задач производственно-технологической деятельности</p> <p>ПК-12.3: ВЛАДЕТЬ Навыками настройки PyChart для разработки программ на языке Python, использования технологий Scipy, Numpy, Pandas, Matplotlib для решения задач производственно-технологической деятельности</p>	Отчет по лабораторным работам	Зачёт: Контрольные вопросы
ПК-5: Способен	ПК-5.1: Знает типовые	ПК-5.1:	Отчет по	

разрабатывать и применять математические методы, системное и прикладное программное обеспечение для решения задач научной деятельности	математические методы и методологии разработки системного и прикладного программного обеспечения для решения задач научной деятельности ПК-5.2: Умеет применять типовые математические методы и методологии разработки системного и прикладного программного обеспечения для решения задач научной деятельности ПК-5.3: Имеет навыки разработки системного и прикладного программного обеспечения для решения задач научной деятельности	<p><b>ЗНАТЬ</b> Методику разработки программ на языке Python, анализа их эффективности и подходы к ее повышению для решения задач научной деятельности.</p> <p><b>ПК-5.2:</b> <b>УМЕТЬ</b> Применять на практике технологии Scipy, Numpy, Pandas, Matplotlib для разработки программ для решения задач научной деятельности</p> <p><b>ПК-5.3:</b> <b>ВЛАДЕТЬ</b> Навыками настройки PyChart для разработки программ на языке Python, использования технологий Scipy, Numpy, Pandas, Matplotlib для решения задач производственно-технологической деятельности.</p>	лабораторным работам	Зачёт: Контрольные вопросы
--	--	--	----------------------	-------------------------------

### 3. Структура и содержание дисциплины

#### 3.1 Трудоемкость дисциплины

	<b>очная</b>
<b>Общая трудоемкость, з.е.</b>	<b>6</b>
<b>Часов по учебному плану</b>	<b>216</b>
в том числе	
<b>аудиторные занятия (контактная работа):</b>	
- занятия лекционного типа	32
- занятия семинарского типа (практические занятия / лабораторные работы)	32
- КСР	2
<b>самостоятельная работа</b>	<b>150</b>
<b>Промежуточная аттестация</b>	<b>0</b> <b>Зачёт</b>

#### 3.2. Содержание дисциплины

(структурированное по темам (разделам) с указанием отведенного на них количества академических часов и виды учебных занятий)

Наименование разделов и тем дисциплины	Всего (часы)	в том числе			
		Контактная работа (работа во взаимодействии с преподавателем), часы из них			Самостоятельная работа обучающегося, часы
		Занятия лекционного типа	Занятия семинарского типа (практические занятия/лабора- торные работы), часы	Всего	
	Ф Ф Ф	Ф Ф Ф	Ф Ф Ф	Ф Ф Ф	Ф Ф Ф
Введение в язык программирования Python.	46	8	8	16	30
Различные способы представления и визуализации научных данных с использованием Matplotlib и NumPy.	56	8	8	16	40
Применение машинного обучения к научным задачам	56	8	8	16	40
Разработка параллельных приложений	56	8	8	16	40
Аттестация	0				
КСР	2				2
Итого	216	32	32	66	150

### Содержание разделов и тем дисциплины

Введение в язык программирования Python. Основы программирования на Python в Python Notebook. Введение в программирование с использованием формул, циклов, списков, функций, массивов (библиотека NumPy), строк, словарей. Введение в классы и ООП. Разработка модульных приложений в среде PyCharm. Различные способы представления и визуализации научных данных с использованием Matplotlib и NumPy. Генерация случайных чисел. Методы Монте-Карло. Символьные вычисления с использованием SymPy. Обзор возможностей NumPy, SciPy, Pandas. Применение машинного обучения к научным задачам. Кластеризация, классификация, регрессионный анализ. Использование статистических возможностей библиотек для анализа данных. Обработка изображений (SciPy). Анализ графов (библиотека igraph). Использование матричных вычислений. Преобразование Фурье. Разработка параллельных приложений. Задачи оптимизации и дифференциального исчисления. Аппроксимация функций. Кроссвалидация. Анализ эффективности разработанного кода. Алгоритмы и структуры данных. Регулярные выражения.

#### 4. Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа обучающихся включает в себя подготовку к контрольным вопросам и заданиям для текущего контроля и промежуточной аттестации по итогам освоения дисциплины приведенным в п. 5.

Самостоятельная работа студентов подразделяется на следующие категории:

- Изучение учебной литературы (см. перечень образовательных материалов).
- Выполнение лабораторных работ на следующие темы: настройка среды разработки и проекта в PyCharm и Anaconda для запуска пакетов scikit-learn, numpy, pandas, matplotlib, решение систем линейных уравнений, визуализация научных данных, аппроксимация функций, численное решение дифференциальных уравнений, решение задач оптимизации, применение машинного обучения для классификации и регрессии фенотипов пациентов с синдромом Дауна,

статистический анализ характеристик кальциевых событий, построение и анализ паренхлитических графов, частотный анализ кальциевых событий, автоматическая разметка медицинских данных, симуляция хемотаксиса бактерий.

## **5. Фонд оценочных средств для текущего контроля успеваемости и промежуточной аттестации по дисциплине (модулю)**

### **5.1 Типовые задания, необходимые для оценки результатов обучения при проведении текущего контроля успеваемости с указанием критериев их оценивания:**

#### **5.1.1 Типовые задания (оценочное средство - Отчет по лабораторным работам) для оценки сформированности компетенции ПК-12:**

1. Вычислить сумму первых  $n$  членов ряда Тейлора для заданной функции.
2. Используя правило прямоугольников, трапеций и Симпсона с помощью вычисления значения интеграла найти приближение числа  $\pi$ .
3. Построение полиномиальной интерполяции искусственно сгенерированных данных.
4. Провести серию экспериментов, тестирующую функции `linalg.solve` и `linalg.cond` на матрицах специального вида.
5. Сравнить алгоритм Штрассена и обычный алгоритм умножения матриц.
6. Экспериментально найти все качественно различные фазовые траектории в модели Вольтерра-Лотке простой экосистемы.
7. Смоделируйте полёт камня Робин Гуда, бросающего его через бойницу крепостной стены в открытое окно, стоящего за ней замка.

#### **5.1.2 Типовые задания (оценочное средство - Отчет по лабораторным работам) для оценки сформированности компетенции ПК-5:**

1. Применение метода Ньютона для решения уравнений.
2. Аппроксимация данных суммой трёх экспонент.
3. Найти параметры модели светимости сверхновой звезды типа I нелинейным методом наименьших квадратов.
4. Подобрать модель, аппроксимирующую данные солнечной активности.
5. Укладка без пересечений  $n$  конгруэнтных кругов в один круг с максимальным радиусом.

#### **Критерии оценивания (оценочное средство - Отчет по лабораторным работам)**

Оценка	Критерии оценивания
зачтено	Работа выполнена в полном объеме и в срок, результаты работы алгоритма корректные на тестовых примерах, результаты работы представлены преподавателю.
не зачтено	Работа не выполнена или выполнена не в полном объеме (программа работает некорректно на тестовых примерах, результаты работы не представлены преподавателю).

### **5.2. Описание шкал оценивания результатов обучения по дисциплине при промежуточной аттестации**

#### **Шкала оценивания сформированности компетенций**

Уровень сформированности компетенций (индикатора достижения компетенций)	плохо	неудовлетворительно	удовлетворительно	хорошо	очень хорошо	отлично	превосходно
	не зачтено		зачтено				
<u>Знания</u>	Отсутствие знаний теоретического материала. Невозможность оценить полноту знаний вследствие отказа обучающегося от ответа	Уровень знаний ниже минимальных требований. Имели место грубые ошибки	Минимально допустимый уровень знаний. Допущено много негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько несущественных ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Ошибок нет.	Уровень знаний в объеме, превышающем программу подготовки.
<u>Умения</u>	Отсутствие минимальных умений. Невозможность оценить наличие умений вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы основные умения. Имели место грубые ошибки	Продemonстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания, но не в полном объеме	Продemonстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания в полном объеме, но некоторые с недочетами	Продemonстрированы все основные умения. Решены все основные задачи. Выполнены все задания в полном объеме, но некоторые с недочетами.	Продemonстрированы все основные умения. Решены все основные задачи с отдельными несущественными недочетами, выполнены все задания в полном объеме	Продemonстрированы все основные умения. Решены все основные задачи. Выполнены все задания, в полном объеме без недочетов
<u>Навыки</u>	Отсутствие базовых навыков. Невозможность оценить наличие навыков вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы базовые навыки. Имели место грубые ошибки	Имеется минимальный набор навыков для решения стандартных задач с некоторыми недочетами	Продemonстрированы базовые навыки при решении стандартных задач с некоторыми недочетами	Продemonстрированы базовые навыки при решении стандартных задач без ошибок и недочетов	Продemonстрированы навыки при решении нестандартных задач без ошибок и недочетов	Продemonстрирован творческий подход к решению нестандартных задач

### Шкала оценивания при промежуточной аттестации

Оценка		Уровень подготовки
зачтено	превосходно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «превосходно», продемонстрированы знания, умения, владения по соответствующим компетенциям на уровне выше предусмотренного программой

	<b>отлично</b>	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «отлично».
	<b>очень хорошо</b>	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «очень хорошо»
	<b>хорошо</b>	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «хорошо».
	<b>удовлетворительно</b>	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «удовлетворительно», при этом хотя бы одна компетенция сформирована на уровне «удовлетворительно»
<b>не зачтено</b>	<b>неудовлетворительно</b>	Хотя бы одна компетенция сформирована на уровне «неудовлетворительно».
	<b>плохо</b>	Хотя бы одна компетенция сформирована на уровне «плохо»

### 5.3 Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения на промежуточной аттестации с указанием критериев их оценивания:

#### 5.3.1 Типовые задания (оценочное средство - Контрольные вопросы) для оценки сформированности компетенции ПК-12

<ul style="list-style-type: none"> <li>• Методы повышения эффективности работы программ, разработанных на Python</li> </ul>
<ul style="list-style-type: none"> <li>• Способы организации параллельных вычислений в Python</li> </ul>
<ul style="list-style-type: none"> <li>• Алгоритмы машинного обучения библиотеки SciPy. Примеры задач</li> </ul>
<ul style="list-style-type: none"> <li>• Методы аппроксимации функций в библиотеке SciPy</li> </ul>
<ul style="list-style-type: none"> <li>• Разбор регулярными выражениями текста в Python</li> </ul>
<ul style="list-style-type: none"> <li>• Обработка изображений с помощью SciPy</li> </ul>
<ul style="list-style-type: none"> <li>• Алгоритмы кроссвалидации</li> </ul>

#### 5.3.2 Типовые задания (оценочное средство - Контрольные вопросы) для оценки сформированности компетенции ПК-5

<ul style="list-style-type: none"> <li>• Способы представления и визуализации научных данных</li> </ul>
<ul style="list-style-type: none"> <li>• Использование объектов array и matrix. Решение задач матричной алгебры.</li> </ul>
<ul style="list-style-type: none"> <li>• Примеры использования списков, словарей, строк</li> </ul>
<ul style="list-style-type: none"> <li>• Основные операторы Python, использование функций в Python, рекурсия, передача и возврат функций в</li> </ul>

качестве параметров, обработка последовательностей и итераторы, генераторы

- Возможности библиотеки `igraph` для анализа графов. Примеры задач

### Критерии оценивания (оценочное средство - Контрольные вопросы)

Оценка	Критерии оценивания
зачтено	Владение основным и дополнительным материалом достаточное или с незначительными ошибками и погрешностями
не зачтено	владение материалом, необходимым по данному предмету, недостаточно. Работу за время семестра можно оценить как неудовлетворительную

## 6. Учебно-методическое и информационное обеспечение дисциплины (модуля)

Основная литература:

1. Борзунов С. В. Языки программирования. Python: решение сложных задач : учебное пособие для вузов / Борзунов С. В., Кургалин С. Д. - Санкт-Петербург : Лань, 2023. - 192 с. - Книга из коллекции Лань - Информатика. - ISBN 978-5-507-45923-0., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=862855&idb=0>.

Дополнительная литература:

1. Лысаков К.Ф. Практическое программирование на Python : учебное пособие / Лысаков К.Ф. - Москва : Новосибирский ГУ, 2023. - 76 с., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=879092&idb=0>.

Программное обеспечение и Интернет-ресурсы (в соответствии с содержанием дисциплины):

1. Официальный сайт о языке Python.  
[<http://www.python.org/>].
2. Библиотека автоматизации GUI тестирования `pywinauto`. [<http://pywinauto.github.io/>]
3. Язык программирования Python. Открытый курс на Stepik. Институт биоинформатики.  
[<https://stepik.org/course/67/>]
4. Python: основы и применение. Открытый курс на Stepik. Институт биоинформатики.  
[<https://stepik.org/course/512/>]
5. Free Interactive Python Tutorial (англ.). [<https://www.learnpython.org/>]
6. Соревновательные задачи на языке Python (англ., HackerRank).  
[<https://www.hackerrank.com/domains/python/py-introduction>]
7. Алексей Задойный. Практикум: математика и Python для анализа данных. Открытый курс на Stepik. [<https://stepik.org/course/3356/>]

## 7. Материально-техническое обеспечение дисциплины (модуля)

Учебные аудитории для проведения учебных занятий, предусмотренных образовательной



программой, оснащены мультимедийным оборудованием (проектор, экран), техническими средствами обучения.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечены доступом в электронную информационно-образовательную среду.

Программа составлена в соответствии с требованиями ОС ННГУ по направлению подготовки/специальности 01.04.02 - Прикладная математика и информатика.

Автор(ы): Золотых Николай Юрьевич, доктор физико-математических наук, доцент.

Заведующий кафедрой: Золотых Николай Юрьевич, доктор физико-математических наук.

Программа одобрена на заседании методической комиссии от 13.12.2023, протокол № 3.