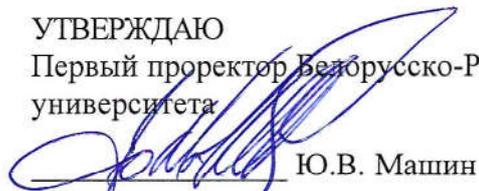


Межгосударственное образовательное учреждение высшего образования  
«Белорусско-Российский университет»

УТВЕРЖДАЮ  
Первый проректор Белорусско-Российского  
университета



Ю.В. Машин

23 06 2023

Регистрационный № УД-010304/Б.Р.О.12/р

## ПРОГРАММИРОВАНИЕ

### РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

**Направление подготовки** 01.03.04 Прикладная математика  
**Направленность (профиль)** Разработка программного обеспечения

**Квалификация** Бакалавр

	Форма обучения
	Очная
Курс	1
Семестр	1, 2
Лекции, часы	68
Лабораторные занятия, часы	84
Экзамен, семестр	1, 2
Контактная работа по учебным занятиям, часы	152
Самостоятельная работа, часы	208
Всего часов / зачётных единиц	360 / 10

Кафедра-разработчик программы: «Высшая математика»

Составитель: Д.В. Роголев, канд. физ.-мат. наук, доц.

Могилёв, 2023

Рабочая программа составлена в соответствии с федеральным государственным образовательным стандартом высшего образования - бакалавриат по направлению подготовки 01.03.04 Прикладная математика №11 от 10.01.2018, учебным планом рег. номер 010304-2.1 от 28.04.2023

Рассмотрена и рекомендована к утверждению кафедрой «Высшая математика» «25» мая 2023 г., протокол № 9.

Зав. кафедрой  В.Г. Замураев

Одобрена и рекомендована к утверждению Научно-методическим советом Белорусско-Российского университета

«21» июня 2023 г., протокол № 6.

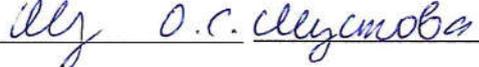
Зам. председателя  
Научно-методического совета

 С.А. Сухоцкий

Рецензент: Наталья Владимировна Кожуренко, доцент кафедры программного обеспечения информационных технологий учреждения образования «Могилевский государственный университет имени А.А. Кулешова», кандидат физико-математических наук

Рабочая программа согласована:

Ведущий библиотекарь



Начальник учебно-методического  
отдела

 О.Е. Печковская

# 1 ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

## 1.1 Цель учебной дисциплины

Целью учебной дисциплины является формирование у студентов базовых знаний в программировании, развитие навыков постановки, формализации и решения задачи на языке программирования высокого уровня, необходимых для изучения последующих учебных дисциплин и успешной дальнейшей работы.

## 1.2 Планируемые результаты изучения дисциплины

В результате освоения учебной дисциплины студент должен

### знать:

- основы алгоритмизации;
- основные конструкции языков программирования высокого уровня;
- основные структуры данных;
- принципы создания программного обеспечения;
- технологии разработки программного обеспечения;

### уметь:

- составлять алгоритмы;
- использовать основные конструкции языков высокого уровня;
- реализовывать алгоритмы в виде программ на языке высокого уровня;

### владеть:

- методами и инструментальными средствами создания программного обеспечения;
- навыками разработки, отладки и тестирования программ.

## 1.3 Место учебной дисциплины в системе подготовки студента

Дисциплина относится к Блоку 1 «Дисциплины (модули)» (обязательная часть).

Перечень учебных дисциплин, которые будут опираться на данную дисциплину:

- «Математическая логика и теория алгоритмов»;
- «Современные математические системы»;
- «Численный анализ»;
- «Численные методы математической физики»;
- «Методы анализа больших данных»;
- «Искусственный интеллект, машинное обучение, нейронные сети»;
- «Объектно-ориентированное программирование»;
- «Практики написания программного кода»;
- «Основы Web-программирования»;
- «Базы данных».

Кроме того, результаты изучения дисциплины используются в ходе ознакомительной и преддипломной практик и при подготовке выпускной квалификационной работы, а также в профессиональной деятельности.

## 1.4 Требования к освоению учебной дисциплины

Освоение данной учебной дисциплины должно обеспечивать формирование следующих компетенций:

Коды формируемых компетенций	Наименования формируемых компетенций
УК-1	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач
УК-2	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений
ОПК-4	Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения

ПК-3	Способен осуществлять непосредственное руководство процессами разработки программного обеспечения
------	---

## 2 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Вклад дисциплины в формирование результатов обучения выпускника (компетенций) и достижение обобщённых результатов обучения происходит путём освоения содержания обучения и достижения частных результатов обучения, описанных в данном разделе.

### 2.1 Содержание учебной дисциплины

Номер тем	Наименование тем	Содержание	Коды формируемых компетенций
1	Алгоритмы.	Алгоритм и его свойства. Принципы разработки алгоритмов.	УК-1, УК-2, ОПК-4, ПК-3
2	Языки программирования.	Основные парадигмы программирования. Структура и описание языков программирования высокого уровня.	УК-1, УК-2, ОПК-4, ПК-3
3	Структура программы.	Среда разработки языка программирования Python. Алфавит, синтаксис, семантика языка программирования Python. Структура программы. Простейший ввод-вывод данных.	УК-1, УК-2, ОПК-4, ПК-3
4	Типы данных.	Литералы, выражения, переменные. Предопределённые типы данных. Преобразование типов данных.	УК-1, УК-2, ОПК-4, ПК-3
5	Операторы.	Операторы присваивания. Действия над целыми числами. Действия над вещественными числами. Математические операторы модуля math. Операторы для работы с последовательностями.	УК-1, УК-2, ОПК-4, ПК-3
6	Условные операторы.	Логические константы. Логические операторы и операторы сравнения. Оператор ветвления if.	УК-1, УК-2, ОПК-4, ПК-3
7	Оператор match...case	Синтаксис оператора match. Соответствие шаблону базовой структуры. Соответствие структуре объекта. Распаковка последовательности и словаря. Перечисления.	УК-1, УК-2, ОПК-4, ПК-3
8	Обработка исключений.	Типы ошибок. Инструкция try...except. Вызов исключений. Инструкция with...as.	УК-2, ОПК-4, ПК-3
9	Циклы.	Цикл for. Функции range, enumerate, zip. Цикл while. Бесконечные циклы. Операторы continue и break.	УК-2, ОПК-4, ПК-3
10	Функции.	Определение функции и её вызов. Расположение определений функций. Рекурсия. Области видимости. Число параметров функции. Анонимные функции. Вложенные функции. Функции-генераторы. Декораторы функций. Функции высшего порядка. Модуль random.	УК-1, УК-2, ОПК-4, ПК-3
11	Строки.	Строки. Операции над строками. Функции и методы для работы со строками. Функции для работы с символами. Типы данных bytes и bytearray.	УК-1, УК-2, ОПК-4, ПК-3
12	Регулярные выражения.	Понятие и синтаксис регулярных выражений. Компилирование регулярных выражений. Поиск совпадений с шаблоном. Объект сопоставления Match. Функции модуля re. Объект регулярного выражения Pattern.	УК-1, УК-2, ОПК-4, ПК-3
13	Кортежи и списки.	Кортежи. Списки. Многомерные списки.	УК-1, УК-2, ОПК-4, ПК-3
14	Словари, множества и диапазоны.	Множества. Словари. Диапазоны. Модуль itertools.	

15	Дата и время.	Получение даты и времени. Форматирование и манипуляции с датой и временем. Время выполнения фрагментов кода.	УК-1, УК-2, ОПК-4, ПК-3
16	Модули и пакеты.	Импорт модулей. Перезагрузка модуля. Пакеты модулей. Вложенные пакеты.	УК-1, УК-2, ОПК-4, ПК-3
17	Работа с файлами.	Работа с текстовыми и бинарными файлами. Модуль csv. Модуль json. Функции для работы с файлами и каталогами.	УК-1, УК-2, ОПК-4, ПК-3
18	Основы объектно-ориентированного программирования (ООП).	Введение в ООП. Поддержка ООП в Python. Класс, методы класса. Конструкторы, деструкторы. Инкапсуляция, наследование, полиморфизм.	УК-1, УК-2, ОПК-4, ПК-3
19	Графический интерфейс пользователя.	Модуль tkinter. Виджеты. Упаковка виджетов. Привязка событий. Canvas.	УК-1, УК-2, ОПК-4, ПК-3
20	Работа с графикой.	Загрузка готового изображения. Создание нового изображения. Манипулирование изображением.	УК-1, УК-2, ОПК-4, ПК-3

## 2.2 Учебно-методическая карта учебной дисциплины

### 1 семестр

№ недели	Лекции (наименование тем)	Часы	Лабораторные занятия	Часы	Самостоятельная работа, часы	Форма контроля знаний	Баллы (max)
<b>Модуль 1</b>							
1	1. Алгоритмы.	2	Лаб. р. 1. Алгоритмы.	2	4		
2	2. Языки программирования.	2	Лаб. р. 1. Алгоритмы.	2	4	ЗЛР	6
3	3. Структура программы.	2	Лаб. р. 2. Введение в Python.	2	4	ЗЛР	6
4	4. Типы данных.	2	Лаб. р. 3. Операции над целочисленными данными.	2	4	ЗЛР	6
5	4. Типы данных.	2	Лаб. р. 4. Операции над вещественными данными.	2	4	ЗЛР	6
6	5. Операторы.	2	Лаб. р. 5. Линейные алгоритмы.	2	4		
7	6. Условные операторы.	2	Лаб. р. 5. Линейные алгоритмы.	2	4	ЗЛР	6
8	7. Оператор match...case.	2	Лаб. р. 6. Оператор ветвления if.	2	4	ПКУ	30
<b>Модуль 2</b>							
9	7. Оператор match...case.	2	Лаб. р. 6. Оператор ветвления if.	2	4	ЗЛР	6
10	8. Обработка исключений.	2	Лаб. р. 7. Оператор цикла for.	2	5		
11	9. Циклы.	2	Лаб. р. 7. Оператор цикла for.	2	5	ЗЛР	6
12	10. Функции.	2	Лаб. р. 8. Оператор цикла while.	2	5		
13	10. Функции.	2	Лаб. р. 8. Оператор цикла while.	2	5	ЗЛР	6
14	11. Строки.	2	Лаб. р. 9. Работа со встроенными функциями.	2	5		
15	11. Строки.	2	Лаб. р. 10. Создание пользовательских функций.	2	5	ЗЛР	6
16	12. Регулярные выражения.	2	Лаб. р. 11 Работа со строками.	2	5		
17	12. Регулярные выражения.	2	Лаб. р. 11 Работа со строками.	2	5	ЗЛР ПКУ	6 30
18-21					36	ПА (экзамен)	40
	Итого	34		34	112		100

## 2 семестр

№ недели	Лекции (наименование тем)	Часы	Лабораторные занятия	Часы	Самостоятельная работа, часы	Форма контроля знаний	Баллы (max)
Модуль 1							
1	13. Кортежи и списки.	2	Лаб. р. 12. Работа с кортежами.	2	2		
			Лаб. р. 12. Работа с кортежами.	2	2	ЗЛР	5
2	13. Кортежи и списки.	2	Лаб. р. 13. Работа со списками.	2	2		
3	14. Словари, множества и диапазоны.	2	Лаб. р. 13. Работа со списками.	2	2	ЗЛР	5
			Лаб. р. 14. Работа с двумерными массивами.	2	2		
4	14. Словари, множества и диапазоны.	2	Лаб. р. 14. Работа с двумерными массивами.	2	2	ЗЛР	5
5	15. Дата и время.	2	Лаб. р. 15. Работа со словарями.	2	2		
			Лаб. р. 15. Работа со словарями.	2	2	ЗЛР	5
6	15. Дата и время.	2	Лаб. р. 16. Работа с датой и временем.	2	2	ЗЛР	5
7	15. Дата и время.	2	Лаб. р. 16. Работа с датой и временем.	2	2		
			Лаб. р. 17. Работа с модулями.	2	2		
8	16. Модули и пакеты.	2	Лаб. р. 17. Работа с модулями.	2	2	ЗЛР ПКУ	5 30
Модуль 2							
9	16. Модули и пакеты.	2	Лаб. р. 18. Работа с пакетами.	2	2		
			Лаб. р. 18. Работа с пакетами.	2	2	ЗЛР	5
10	17. Работа с файлами.	2	Лаб. р. 19. Работа с текстовыми файлами.	2	2		
11	17. Работа с файлами.	2	Лаб. р. 19. Работа с текстовыми файлами.	2	3	ЗЛР	5
			Лаб. р. 20. Работа с бинарными файлами.	2	3		
12	18. Основы объектно-ориентированного программирования (ООП).	2	Лаб. р. 20. Работа с бинарными файлами.	2	3	ЗЛР	5
13	18. Основы объектно-ориентированного программирования (ООП).	2	Лаб. р. 21. Основы объектно-ориентированного программирования.	2	3		
			Лаб. р. 21. Основы объектно-ориентированного программирования.	2	3	ЗЛР	5
14	19. Графический интерфейс пользователя.	2	Лаб. р. 22. Графический интерфейс.	2	3		
15	19. Графический интерфейс пользователя.	2	Лаб. р. 22. Графический интерфейс.	2	3		
			Лаб. р. 22. Графический интерфейс.	2	3	ЗЛР	5
16	20. Работа с графикой.	2	Лаб. р. 23. Работа с графикой.	2	3		
17	20. Работа с графикой.	2	Лаб. р. 23. Работа с графикой.	2	3	ЗЛР ПКУ	5 30
18-20					36	ПА (экзамен)	40
	Итого	34		50	96		100

Принятые обозначения:  
Текущий контроль –

ЗЛР – защита лабораторной работы;  
 ПКУ – промежуточный контроль успеваемости.  
 ПА – Промежуточная аттестация.

Итоговая оценка определяется как сумма текущего контроля и промежуточной аттестации и соответствует баллам:

#### Экзамен

Оценка	Отлично	Хорошо	Удовлетворительно	Неудовлетворительно
Баллы	87-100	65-86	51-64	0-50

### 3 ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

При изучении дисциплины используется модульно-рейтинговая система оценки знаний студентов. Применение форм и методов проведения занятий при изучении различных тем курса представлено в таблице.

№ п/п	Форма проведения занятия	Вид аудиторных занятий		Всего часов
		Лекции	Лабораторные занятия	
1	Мультимедиа	1-20	–	68
2	С использованием ЭВМ	–	1-23	84
<b>ИТОГО</b>		68	84	

### 4 ОЦЕНОЧНЫЕ СРЕДСТВА

Используемые оценочные средства по учебной дисциплине представлены в таблице и хранятся на кафедре.

№ п/п	Вид оценочных средств	Количество комплектов
1	Задания в тестовой форме	1
2	Вопросы к экзамену	2
3	Билеты к экзамену	2

### 5 МЕТОДИКА И КРИТЕРИИ ОЦЕНКИ КОМПЕТЕНЦИЙ СТУДЕНТОВ

#### 5.1 Уровни сформированности компетенций

№ п/п	Уровни сформированности компетенции	Содержательное описание уровня	Результаты обучения
УК-1. Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач			
ИУК-1.4 Способен применять системный подход при формализации и алгоритмизации поставленных задач и при написании программного кода			
1	Пороговый уровень	Базовые знания в объеме рабочей программы (знание основных понятий и методов), умение решать типовые задачи под руководством преподавателя.	Имеет представление о системном подходе при формализации и алгоритмизации задач и при написании программного кода.
2	Продвинутый уровень	Полные знания в объеме рабочей программы, правильное использование терминологии, способность самостоятельно решать типовые задачи учебной дисциплины.	Способен под руководством преподавателя применять системный подход при формализации и алгоритмизации поставленных задач и при написании программного кода.

3	Высокий уровень	Систематизированные, глубокие и полные знания в объёме рабочей программы, точное использование научной терминологии и свободное владение инструментарием учебной дисциплины, умение анализировать и применять теоретические знания при самостоятельном решении типовых учебных задач и задач повышенной сложности.	Способен самостоятельно применять системный подход при формализации и алгоритмизации поставленных задач и при написании программного кода.
<i>УК-2.</i> Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений			
<i>ИУК-2.5</i> Способен выбирать оптимальные способы алгоритмизации поставленных задач			
1	Пороговый уровень	Базовые знания в объёме рабочей программы (знание основных понятий и методов), умение решать типовые задачи под руководством преподавателя.	Имеет представление о выборе способа алгоритмизации поставленной задачи.
2	Продвинутый уровень	Полные знания в объёме рабочей программы, правильное использование терминологии, способность самостоятельно решать типовые задачи учебной дисциплины.	Способен под руководством преподавателя выбрать оптимальный способ алгоритмизации поставленной задачи.
3	Высокий уровень	Систематизированные, глубокие и полные знания в объёме рабочей программы, точное использование научной терминологии и свободное владение инструментарием учебной дисциплины, умение анализировать и применять теоретические знания при самостоятельном решении типовых учебных задач и задач повышенной сложности.	Способен самостоятельно выбрать оптимальный способ алгоритмизации поставленной задачи.
<i>ОПК-4.</i> Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения			
<i>ИОПК-4.1</i> Способен формализовать и алгоритмизировать поставленные задачи, применять знание процедурного, структурного, функционального программирования при разработке компьютерных программ			
1	Пороговый уровень	Базовые знания в объёме рабочей программы (знание основных понятий и методов), умение решать типовые задачи под руководством преподавателя.	Имеет представление о формализации и алгоритмизации задачи, о выборе способа программирования.
2	Продвинутый уровень	Полные знания в объёме рабочей программы, правильное использование терминологии, способность самостоятельно решать типовые задачи учебной дисциплины.	Способен под руководством преподавателя формализовать и алгоритмизировать задачу, выбрать способ программирования.
3	Высокий уровень	Систематизированные, глубокие и полные знания в объёме рабочей программы, точное использование научной терминологии и свободное владение инструментарием учебной дисциплины, умение анализировать и применять теоретические знания при самостоятельном решении типовых учебных задач и задач повышенной сложности.	Способен самостоятельно формализовать и алгоритмизировать задачу, выбрать способ программирования.
<i>ПК-3.</i> Способен осуществлять непосредственное руководство процессами разработки программного обеспечения			

ИПК-3.2 Способен оценивать качество и эффективность программного кода, принимать решения по изменению программного кода, редактировать программный код			
1	Пороговый уровень	Базовые знания в объёме рабочей программы (знание основных понятий и методов), умение решать типовые задачи под руководством преподавателя.	Имеет представление о написании, проверке и отладке программного кода.
2	Продвинутый уровень	Полные знания в объёме рабочей программы, правильное использование терминологии, способность самостоятельно решать типовые задачи учебной дисциплины.	Способен под руководством преподавателя написать программный код и осуществить его проверку и отладку.
3	Высокий уровень	Систематизированные, глубокие и полные знания в объёме рабочей программы, точное использование научной терминологии и свободное владение инструментарием учебной дисциплины, умение анализировать и применять теоретические знания при самостоятельном решении типовых учебных задач и задач повышенной сложности.	Способен самостоятельно написать программный код и осуществить его проверку и отладку.

## 5.2 Методика оценки знаний, умений и навыков студентов

Результаты обучения	Оценочные средства
<i>УК-1.</i> Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	
Пороговый уровень	Задания в тестовой форме. Вопросы к экзамену. Билеты к экзамену.
Продвинутый уровень	Задания в тестовой форме. Вопросы к экзамену. Билеты к экзамену.
Высокий уровень	Задания в тестовой форме. Вопросы к экзамену. Билеты к экзамену.
<i>УК-2.</i> Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	
Пороговый уровень	Задания в тестовой форме. Вопросы к экзамену. Билеты к экзамену.
Продвинутый уровень	Задания в тестовой форме. Вопросы к экзамену. Билеты к экзамену.
Высокий уровень	Задания в тестовой форме. Вопросы к экзамену. Билеты к экзамену.
<i>ОПК-4.</i> Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	
Пороговый уровень	Задания в тестовой форме. Вопросы к экзамену. Билеты к экзамену.
Продвинутый уровень	Задания в тестовой форме. Вопросы к экзамену. Билеты к экзамену.
Высокий уровень	Задания в тестовой форме. Вопросы к экзамену. Билеты к экзамену.
<i>ПК-2.</i> Способен обоснованно выбирать методы решений поставленных задач, разрабатывать алгоритмы решений, реализовывать алгоритмы в виде программ, анализировать результаты	

Пороговый уровень	Задания в тестовой форме. Вопросы к экзамену. Билеты к экзамену.
Продвинутый уровень	Задания в тестовой форме. Вопросы к экзамену. Билеты к экзамену.
Высокий уровень	Задания в тестовой форме. Вопросы к экзамену. Билеты к экзамену.
<i>ПК-3. Способен осуществлять непосредственное руководство процессами разработки программного обеспечения</i>	
Пороговый уровень	Задания в тестовой форме. Вопросы к экзамену. Билеты к экзамену.
Продвинутый уровень	Задания в тестовой форме. Вопросы к экзамену. Билеты к экзамену.
Высокий уровень	Задания в тестовой форме. Вопросы к экзамену. Билеты к экзамену.

### 5.3 Критерии оценки лабораторных работ

Лабораторные работы оцениваются до 6 баллов:

0-1 баллов – оборудование и методы использованы неправильно, проявлена неудовлетворительная теоретическая подготовка, необходимые навыки и умения не освоены, результат работы не соответствует её целям;

2-3 баллов – оборудование и методы частично использованы правильно, проявлена удовлетворительная теоретическая подготовка, необходимые навыки и умения частично освоены, результат работы частично соответствует её целям;

4 баллов – оборудование и методы в основном использованы правильно, проявлена хорошая теоретическая подготовка, необходимые навыки и умения в основном освоены, результат работы в основном соответствует её целям;

5-6 баллов – оборудование и методы использованы правильно, проявлена отличная теоретическая подготовка, необходимые навыки и умения полностью освоены, результат работы полностью соответствует её целям.

### 5.4 Критерии оценки экзамена

На экзамене по дисциплине предусмотрены экзаменационные билеты, состоящие из 4 заданий. Каждое задание оценивается в 10 баллов. Экзамен считается сданным успешно, если правильно выполнено 2 задания и более. По итогам выполнения экзамена студент может набрать до 40 баллов включительно.

## 6 МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ОРГАНИЗАЦИИ И ВЫПОЛНЕНИЮ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

Самостоятельная работа студентов (СРС) направлена на закрепление и углубление освоения учебного материала, развитие практических умений. СРС включает следующие виды самостоятельной работы студентов:

- изучение конспекта лекций, методических рекомендаций, основной и дополнительной литературы, ресурсов сети Интернет;
- решение индивидуальных задач во время проведения лабораторных занятий под контролем преподавателя.

Перечень контрольных вопросов и заданий для самостоятельной работы студентов приведён в приложении и хранится на кафедре.

Для СРС рекомендуется использовать источники, приведённые в п. 7.

## 7 УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

### 7.1 Основная литература

№ п/п	Библиографическое описание	Гриф	Количество экземпляров/URL
1	Гуриков, С. Р. Основы алгоритмизации и программирования на Python : учебное пособие / С.Р. Гуриков. – Москва : ФОРУМ : ИНФРА-М, 2023. – 343 с.	Рек. в качестве уч. пособия для студ. высших учебных заведений, обучающихся по направлениям подготовки 09.03.01 «Информатика и вычислительная техника», 09.03.02 «Информационные системы и технологии» (квалификация (степень) «бакалавр»)	<a href="https://znanium.com/catalog/product/1913856">https://znanium.com/catalog/product/1913856</a>
2	Программирование, численные методы и математическое моделирование : учеб. пособие / И. Г. Семакин [и др.]. - М. : КНОРУС, 2021. - 298с.	Рек. ГОУ ВО "Моск. гос. технол. ун-т "Станкин"; МО и науки РФ ФГУ "Федер. ин-т развития образования" в качестве учеб. пособия для студ. вузов	5

### 7.2 Дополнительная литература

№ п/п	Библиографическое описание	Гриф	Количество экземпляров/URL
1	Жуков, Р. А. Язык программирования Python: практикум : учебное пособие / Р.А. Жуков. – Москва : ИНФРА-М, 2023. – 216 с.	Рек. Межрегиональным учебно-методическим советом профессионального образования в качестве учеб. пособия для студ. высших учебных заведений, обучающихся по направлению подготовки 38.03.05 «Бизнес-информатика» (квалификация (степень) «бакалавр»)	<a href="https://znanium.com/catalog/product/1915716">https://znanium.com/catalog/product/1915716</a>
	Чернышев, С. А. Основы программирования на Python : учебное пособие для вузов / С. А. Чернышев. – 2-е изд., перераб. и доп. – Москва : Издательство Юрайт, 2023. – 349 с.	Рек. УМО ВО в качестве учеб. пособия для студ. высших учебных заведений, обучающихся по IT-направлениям	<a href="https://urait.ru/bcode/532446">https://urait.ru/bcode/532446</a>
2	Карякин, М. И. Технологии программирования и компьютерный практикум на языке Python : учебное пособие / М. И. Карякин, К. А. Ватульян, Р. М. Мнухин ; Южный федеральный университет. - Ростов-на-Дону ; Таганрог : Изд-во Южного федерального ун-та, 2022. – 242 с.	-	<a href="https://znanium.com/catalog/product/2057604">https://znanium.com/catalog/product/2057604</a>

### 7.3 Перечень ресурсов сети Интернет по изучаемой дисциплине

python.org  
 stackoverflow.com  
 github.com  
 habr.com

## **7.4 Перечень наглядных и других пособий, методических рекомендаций по проведению учебных занятий, а также методических материалов к используемым в образовательном процессе техническим средствам**

### **7.4.1 Методические рекомендации**

1. Программирование. Методические рекомендации к лабораторным работам для студентов направления подготовки 01.03.04 «Прикладная математика». Часть 1 / составители Маковецкий И.И., Роголев Д.В., Бондарев А.Н. – Могилев: Межгосударственное образовательное учреждение высшего образования «Белорусско-Российский университет», 2021 – 35 с. (50 экз.).

2. Программирование. Методические рекомендации к лабораторным работам для студентов направления подготовки 01.03.04 «Прикладная математика». Часть 2 / составители Маковецкий И.И., Роголев Д.В., Бондарев А.Н. – Могилев: Межгосударственное образовательное учреждение высшего образования «Белорусско-Российский университет», 2021 – 37 с. (50 экз.).

### **7.4.3 Информационные технологии**

Тема 1 – Алгоритмы.

Тема 2 – Языки программирования.

Тема 3 – Структура программы.

Тема 4 – Типы данных.

Тема 5 – Операторы.

Тема 6 – Условные операторы.

Тема 7 – Оператор match...case

Тема 8 – Обработка исключений.

Тема 9 – Циклы.

Тема 10 – Функции.

Тема 11 – Строки.

Тема 12 – Регулярные выражения.

Тема 13 – Кортежи и списки.

Тема 14 – Словари, множества и диапазоны.

Тема 15 – Дата и время.

Тема 16 – Модули и пакеты.

Тема 17 – Работа с файлами.

Тема 18 – Основы объектно-ориентированного программирования (ООП).

Тема 19 – Графический интерфейс пользователя.

Тема 20 – Работа с графикой.

### **7.4.3 Перечень программного обеспечения, используемого в образовательном процессе**

Свободно распространяемое ПО: язык программирования Python, редактор VS Code.

## **8 МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ**

Материально-техническое обеспечение дисциплины содержится в паспорте лаборатории «ауд. 405», рег. номер ПУЛ-4.535-405/1-22.

# ПРОГРАММИРОВАНИЕ

## АННОТАЦИЯ К РАБОЧЕЙ ПРОГРАММЕ ДИСЦИПЛИНЫ

Направление подготовки 01.03.04 Прикладная математика

Направленность (профиль) Разработка программного обеспечения

	Форма обучения
	Очная
Курс	1
Семестр	1, 2
Лекции, часы	68
Лабораторные занятия, часы	84
Экзамен, семестр	1, 2
Контактная работа по учебным занятиям, часы	152
Самостоятельная работа, часы	208
Всего часов / зачётных единиц	360 / 10

### 1. Цель учебной дисциплины

Целью учебной дисциплины является формирование у студентов базовых знаний в программировании, развитие навыков постановки, формализации и решения задачи на языке программирования высокого уровня, необходимых для изучения последующих учебных дисциплин и успешной дальнейшей работы.

### 2. Планируемые результаты изучения дисциплины

В результате освоения учебной дисциплины студент должен

**знать:**

- основы алгоритмизации;
- основные конструкции языков программирования высокого уровня;
- основные структуры данных;
- принципы создания программного обеспечения;
- технологии разработки программного обеспечения;

**уметь:**

- составлять алгоритмы;
- использовать основные конструкции языков высокого уровня;
- реализовывать алгоритмы в виде программ на языке высокого уровня;

**владеть:**

- методами и инструментальными средствами создания программного обеспечения;
- навыками разработки, отладки и тестирования программ.

### 3. Требования к освоению учебной дисциплины

Освоение данной учебной дисциплины должно обеспечивать формирование следующих компетенций:

Коды формируемых компетенций	Наименования формируемых компетенций
УК-1	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач
УК-2	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений
ОПК-4	Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения
ПК-3	Способен осуществлять непосредственное руководство процессами разработки программного обеспечения

### 4. Образовательные технологии

Мультимедиа, с использованием ЭВМ