

**Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ М.В.ЛОМОНОСОВА»**

**ФИЛИАЛ МГУ В Г. ГРОЗНОМ**

**УТВЕРЖДАЮ**  
Заместитель директора  
Филиала – руководитель  
образовательных программ  
**А.С. Воронцов**



**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

Наименование дисциплины:

**Основы программирования**

---

Уровень высшего образования:

**Бакалавриат**

---

Направление подготовки

**42.03.05 Медиакоммуникации**

---

Направленность (профиль)/специализация образовательной программы

**Общий**

---

Форма обучения

**Очная**

---

Москва 2025

Рабочая программа дисциплины разработана и утверждена в соответствии с самостоятельно установленным МГУ образовательным стандартом, утвержденным приказом МГУ от 29 декабря 2018 года № 1775 (в редакции приказов МГУ от 11 сентября 2019 года № 1109, от 10 июня 2021 года № 609, от 29 мая 2023 года № 700, 702, 703) для реализуемых основных профессиональных образовательных программ высшего образования по направлению подготовки 42.03.05 «Медиакоммуникации».

## Аннотация

Целями освоения дисциплины «Основы программирования» являются понимание сущности алгоритмизации и программирования, получение практической подготовки в области использования вычислительной техники и программного обеспечения для решения задач в области профессиональной деятельности.

### 1.1 Цели освоения дисциплины

Целями освоения дисциплины «Основы программирования» являются: изучение студентами основных понятий, методов, базовых принципов, механизмов и средств языка программирования Python; приобретение знаний в области работы с массивами данных; рассмотрение алгоритмов решения базовых задач анализа данных; приобретение навыков решения практических задач.

### 1.2. Задачи освоения дисциплины

1. Владение навыками использования возможностей языка программирования Python; владение основными методами, способами и средствами решения задач анализа данных.

2. Знание особенностей работы журналиста в условиях цифровизации медиакоммуникаций и развития журналистики данных.

## 2. Место дисциплины в структуре ООП

Дисциплина «Основы программирования» относится к базовой части ООП модуль «Цифровые технологии». Студенты должны иметь знания в объеме курсов "Информатика", «Основы высшей математики», «Статистика и анализ данных».

## 3. Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование следующих компетенций по ОС МГУ:

Планируемые результаты обучения по дисциплине (модулю)		
Содержание и код компетенции.	Индикатор (показатель) достижения компетенции	Планируемые результаты обучения по дисциплине, сопряженные с индикаторами достижения компетенций
<b>ОПК-8.</b> Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности	<b>ОПК-8.1 (общий по УГНС)</b> Отбирает для осуществления профессиональной деятельности необходимое техническое оборудование и программное обеспечение <b>ОПК-8.2 (по направлению подготовки Медиакоммуникации)</b> Эксплуатирует современные стационарные и мобильные цифровые устройства на всех этапах	<b>ОПК-8.1. 3-1 Знать</b> основные приемы работы в среде PyCharm; синтаксис и семантику операторов, принципы построения программ для языка программирования Python; способы описания и использования основных структур данных: структура, массив, файл в языке программирования Python;

	создания медиапроекта и (или) медиапродукта	<p><b>ОПК-8.2. 3-1 Знать</b> основные алгоритмические конструкции, методы программирования, методы обработки числовой, текстовой и структурированной информации; алгоритмы решения типовых задач по программированию.</p> <p><b>ОПК-8.1. У-1 Уметь</b> выбирать метод программирования в зависимости от типов данных; разрабатывать алгоритм в соответствии с этапами решения задачи;</p> <p><b>ОПК-8.2. У-1 Уметь</b> реализовывать алгоритмы средствами языка программирования процедурной парадигмы; представлять отчеты по результатам решения задач.</p> <p><b>ОПК-8.1. В-1 Владеть</b> технологией разработки алгоритмов типовых задач по программированию,</p> <p><b>ОПК-8.2. В-1 Владеть</b> основными методами обработки числовых, текстовых и структурированных данных</p>
ПК-6. Способен участвовать в производственном процессе выпуска мультимедийного контента с применением современной технологической инфраструктуры	<p><b>ПК-6.1.</b> Знает архитектуру информационных и программных продуктов, алгоритмы и принципы их создания</p> <p><b>ПК-6.2.</b> Умеет разрабатывать концепции информационных и программных продуктов</p> <p><b>ПК-6.3.</b> Владеет навыками проектирования информационных и программных продуктов</p>	<p><b>ПК-6.1. 3-1 Знать</b> основные понятия программной инженерии и жизненный цикл программных продуктов;</p> <p><b>ПК-6.2. 3-1 Знать</b> технологии разработки программных продуктов и методы производства программных продуктов;</p> <p><b>ПК-6.3. 3-1 Знать</b> основные программные средства, применяемые при автоматизированной разработке ПО.</p>

		<p><b>ПК-6.1. У.1. Уметь</b> работать с инструментальными средствами, поддерживающими создание программных продуктов;</p> <p><b>ПК-6.2. У.1. Уметь</b> применять теоретические знания при решении профессионально значимых задач</p> <p><b>ПК-6.3. У.1. Уметь</b> проектировать и разрабатывать программные продукты;</p> <p><b>ПК-6.1. В-1 Владеть</b> основными технологиями, методами, средствами и навыками выбора программных продуктов для решения задач в различных предметных областях.</p> <p><b>ПК-6.2. В-1 Владеть</b> основными технологиями, методами, средствами и навыками реализации программных продуктов для решения задач в различных предметных областях.</p> <p><b>ПК-6.3. В-1 Владеть</b> основными технологиями, методами, средствами и навыками проектирования программных продуктов для решения задач в различных предметных областях.</p>
--	--	---

#### 4. Структура и содержание дисциплины «Основы программирования»

Общая трудоемкость дисциплины составляет 2 зачетных единицы (72 часа)

##### 4.1. Структура дисциплины

№ п/п	Раздел Дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)	Формы текущего контроля успеваемости (по неделям семестра) Форма промежуточ-
-------	-------------------	---------	-----------------	--	---

				лекции	семинары	СРС	Коллоквиумы	ной аттестации (по семестрам)
1	Формализация, моделирование, алгоритмы.	7	1-2		4	4		
2	Основы языка программирования Python	7	3-6		8	8		
3	Обработка массивов на языке Python	7	7-8		4	4		
4	Алгоритмы обработки данных	7	9-13		10	10		Тестирование
5	Основы использования NumPy и Pandas для анализа данных	7	14-18		10	10		
	<b>ИТОГО</b>		<b>72</b>		36	36		<b>Зачет</b>

#### 4.2. Содержание разделов дисциплины

№ п/п	Наименование раздела дисциплины	Содержание раздела
1	Формализация, моделирование, алгоритмы.	<p>Элементы теории множеств. Основы алгебры логики. Конечные автоматы как формальная модель вычислительной системы.</p> <p>Алгоритмы и программы. Основные алгоритмические конструкции. Блок-схемы. Структуры данных. Алгоритмы поиска и сортировки. Программирование: процедурное, декларативное, функциональное. Объектно-ориентированное программирование. Интерпретаторы и компиляторы.</p>
2	Основы языка программирования Python	<p>Основные этапы компьютерного решения задач. Программа. Общие сведения о языке Python. Основные объекты языка: команды, операторы, данные. Константы, идентификаторы, переменные. Правила записи программ на языке Python. Структура программы на языке программирования Python. Отладка и тестирование программы. Использование возможностей среды PyCharm для трассировки выполнения, отслеживания текущих значений переменных</p> <p>Операторы ввода и вывода данных. Ввод данных различных типов. Форматированный вывод данных различных типов. Знаки операций. Оператор присваивания. Простой оператор. Составной оператор. Структурные операторы: условный, переключатель (оператор передачи управления). Примеры разветвляющихся программ (Нахождение минимума и максимума двух,</p>

		<p>трех, четырех данных чисел без использования массивов и циклов, проверка числа на четность и др.) Оператор цикла с предусловием, оператор цикла с постусловием. Оператор цикла с параметром. Операторы break, continue, return. Примеры циклических программ (последовательности чисел: поиск суммы, произведения элементов последовательности; работа с разрядами чисел; числа-палиндромы; запись натурального числа в позиционной системе с основанием, меньшим или равным 10; обработка и преобразование такой записи числа; поиск наименьшего простого делителя данного натурального числа, проверка числа на простоту и др.)</p> <p>Функции пользователя. Локальные и глобальные переменные. Параметры функций. Способы передачи параметров в функции. Основные встроенные модули. Random, Math, Turtle.</p>
3	Обработка массивов на языке Python	<p>Списки: объявление, размер, размерность, ограничения на типы индексов и элементов. Этапы работы со списками. Способы формирования списков.</p>
4	Алгоритмы обработки данных	<p>Понятие рекурсии. Рекурсия в информатике. Рекурсивная триада. Рекурсивный спуск. Рекурсивный подъем. Алгоритмы теории чисел (факториал, степень, моделирование арифметических операций, распознавание простого числа, вычисление количества делителей натурального числа). Поиск элемента с заданным свойством. (нахождение минимального (максимального) значения в данном массиве и количества элементов, равных ему, за однократный просмотр массива; нахождение второго по величине (второго максимального или второго минимального) значения в данном массиве за однократный просмотр массива и др.) Дихотомический поиск. Простые алгоритмы сортировки: простыми обменами, простым выбором, простыми включениями. Слияние двух упорядоченных массивов в один без использования сортировки. Вставка и удаление элементов в массиве; перестановка элементов данного массива в обратном порядке.</p>
5	Основы использования NumPy и Pandas для анализа данных	<p>Введение в NumPy. Векторизованные вычисления. Агрегирующие функции. Broadcasting. Логические операции. Введение в pandas. Индексы в pandas. Иерархические (множественные) индексы. Иерархические колонки. Агрегирование в pandas. Объединение нескольких наборов данных. Решение базовых задач анализа данных.</p>

## 5. Рекомендуемые образовательные технологии

При проведении занятий по дисциплине используются следующие виды образовательных технологий: традиционные образовательные технологии (семинарское занятие); технологии проблемного обучения (семинарское занятие); интерак-

тивные технологии; информационно-коммуникационные образовательные технологии.

Дисциплина включает семинарские занятия.

Семинарские занятия проводятся в форме проблемной ситуации, когда некоторый аспект рассмотренной темы излагается преподавателем более подробно. Часть информации конспектируется. Большая часть времени выделена на работу с использованием компьютерной техники и программного обеспечения.

В рамках курса используются активные и интерактивные методы обучения в процессе проведения занятий. Основными особенностями интерактивных занятий являются интерактивные практические упражнения и задания, которые выполняются обучающимися не только и не столько на закрепление изученного материала, но и на самостоятельное изучение нового.

Это предполагает отличную от привычной логику образовательного процесса: не от теории к практике, а от формирования нового опыта к его теоретическому осмыслению через применение.

Делясь своими знаниями и опытом деятельности, студенты берут на себя часть обучающих функций преподавателя, что повышает их мотивацию и способствует большей продуктивности обучения.

В процессе проведения занятий важное место занимает обучение с использованием ресурсов Интернета, электронных учебников и справочников, работой в режиме онлайн, что позволяет студентам вступать в «живой» (интерактивный) диалог (письменный или устный) с реальным партнером (преподавателем). Такой диалог также делает возможным активный обмен сообщениями между пользователем и информационной системой в режиме реального времени.

Удельный вес занятий, проводимых в интерактивных формах, составляет около 30% аудиторных занятий.

Основная часть самостоятельной работы должна включать самоподготовку студентов с использованием учебной литературы согласно списку литературы, приведенному в Рабочей программе по указанной дисциплине.

Студент должен самостоятельно освоить разделы, указанные в Рабочей программе для самостоятельной работы. Как правило, эти разделы включают в себя темы дисциплины, на которые в курсе читаемых лекций уделялось недостаточное внимание, либо эти разделы не включены в курс лекций, а должны осваиваться студентом самостоятельно. В разделы самостоятельной работы студентов также включаются наиболее сложные для понимания части дисциплины, требующие более детального и углубленного изучения и осмысления.

Студент должен найти в учебной литературе соответствующую тему, прочитать ее и попытаться изложить устно или письменно основные положения или идеи прочитанного раздела.

Далее студент должен составить сам письменно вопросы, отражающие основные положения разбираемой темы и устно (или письменно) ответить на них.

Во многих рекомендуемых учебных пособиях в конце каждого раздела имеются тесты или уже сформулированные вопросы, на которые студент должен самостоятельно ответить.

## **6. Учебно-методическое обеспечение самостоятельной работы студентов. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.**

Описание показателей и критериев оценивания компетенций на различных этапах их формирования



№ п/п	Наименование оценочного средства	Краткая характеристика оценоч- ного средства	Представление оценочного средства в фон- де
1.	Тестирование	Работа, состоящая из тестовых заданий, реализуется в электронном виде.	Тест
2.	Проектное задание	Индивидуальное проектное задание заключается в разработке программы, удовлетворяющего системе требований.	Варианты проектных заданий, требования к проектному заданию, требования к отчету

2Б) Описание шкал оценивания

ТЕСТИРОВАНИЕ	A (90-100%)	По результатам тестирования обучающийся дал 90-100% правильных ответов на тестовые задания
	B (82-89%)	По результатам тестирования обучающийся дал 82-89% правильных ответов на тестовые задания
	C (75-81%)	По результатам тестирования обучающийся дал 75-81% правильных ответов на тестовые задания
	D (67-74%)	По результатам тестирования обучающийся дал 67-74% правильных ответов на тестовые задания
	E (60-66%)	По результатам тестирования обучающийся дал 60-67% правильных ответов на тестовые задания
ПРОЕКТНОЕ ЗАДАНИЕ	A (90-100%)	Проектное задание выполнено в полном соответствии с представленными требованиями, содержит все необходимые разделы, качественно оформлен отчет по проектному заданию
	B (82-89%)	Проектное задание выполнено в недостаточно полном соответствии с представленными требованиями, содержит все необходимые разделы, качественно оформлен отчет по проектному заданию
	C (75-81%)	Проектное задание выполнено в недостаточно полном соответствии с представленными требованиями, содержит не все необходимые разделы, качественно оформлен отчет по проектному заданию
	D (67-74%)	Проектное задание выполнено в недостаточно полном соответствии с представленными требованиями, содержит не все необходимые разделы, не качественно оформлен отчет

		по проектному заданию
	Е (60-66%)	Проектное задание выполнено в не полном соответствии с только отдельные разделы, не качественно оформлен отчет по проектному заданию

Виды работы	Знания и компетенции, проверяемые в процессе выполнения данного вида работы	Доля вида работы в итоговой оценке
Тест	Темы 1,2	от 0 до 30 %
Решение практических задач	Темы 1,2,3	от 0 до 30 %
Проектное задание	Темы 3	от 0 до 40 %
Итог (зачет)	Устный ответ студента на 2 вопроса в билете, направленный на определение знаний фактического и теоретического материала дисциплины.	от 0% до 100%

- «зачтено» выставляется студенту, если сумма баллов за два ответа составляет 61% и более;
- «не зачтено» выставляется студенту, если сумма баллов менее 60%.

### Примерные вопросы к зачету (1 семестр)

1. Понятие алгоритма. Исполнитель. Система команд исполнителя. Свойства алгоритмов. Компьютер как универсальный исполнитель.
2. Программирование как раздел информатики. Языки программирования. Обзор. Классификация.
3. Язык программирования Python. Общая характеристика языка: место в классификации языков, синтаксис и семантика языка, основные объекты языка, операторы, структура программы.
4. Синтаксические и логические ошибки. Тестирование и отладка программы.
5. Данные в языке Python: константы и переменные. Скалярные типы данных.
6. Данные числовых типов в языке Python: объявление, характеристика, допустимые операции, приведение типов. Пример использования.
7. Операции языка Python. Приоритет операций. Оператор и операция присваивания в языке Python. Множественное присваивание. Выражения.
8. Алгоритмическая конструкция ветвления: полная и неполная форма, блок-схемы. Условная операция. Условный оператор в языке Python: структура оператора, полная и неполная формы, использование сложных условий. Пример на языке Python.
9. Циклические алгоритмы: понятие, виды (перечислить). Алгоритмическая конструкция цикла с предусловием (понятие, использование, блок-схема). Оператор цикла for: структура оператора, пример использования.
10. Циклические алгоритмы: понятие, виды (перечислить). Алгоритмическая конструкция цикла с предусловием (понятие, использование, блок-схема). Оператор цикла с предусловием while в языке Python: структура оператора, допустимые и недопустимые условия, пример использования.

11. Типовые циклические алгоритмы: максимум/минимум, сумма/произведение, количество.
12. Линейный массив: понятие массива, объявление, инициализация массива, индексация элементов. Формирование и вывод массива.
13. Функции в языке Python: понятие, объявление, прототипы функций. Вызов функции. Типы возвращаемых значений.
14. Локальные и глобальные переменные. Области действия и области видимости. Экранирование переменных.
15. Основные задачи анализа данных.
16. Библиотечные функции. Заголовочные файлы. Подключение библиотек. Функции математической библиотеки.
17. Векторизованные вычисления. Агрегирующие функции. Broadcasting.
18. Логические операции. Введение в pandas. Индексы в pandas.
19. Иерархические (множественные) индексы. Иерархические колонки.
20. Агрегирование в pandas. Объединение нескольких наборов данных

### Примерные тестовые задания

1. Задан фрагмент алгоритма:

если  $a < b$ , то  $c = b - a$ ,

иначе  $c = 2 * (a - b)$

$d = 0$

пока  $c > a$  выполнить действия

$d = d + 1$ ,

$c = c - 1$

В результате выполнения данного алгоритма с начальными значениями  $a = 8$ ,  $b = 3$ , переменные  $c$  и  $d$  примут значения:

A)  $c = 10$ ,  $d = 1$

B)  $c = 5$ ,  $d = 0$

C)  $c = 8$ ,  $d = 2$

D)  $c = -5$ ,  $d = 1$

2. Константы - это\_\_\_\_\_.

A) данные, значение которых изменяется в процессе выполнения программы

B) операторы

C) данные, значение которых не изменяется в процессе выполнения программы

D) массивы

3. Алгоритм называется линейным, если\_\_\_\_\_.

A) Выполнение операций зависит от условия

B) Операции выполняются друг за другом

C) Одни и те же операции выполняются многократно

D) Присутствие всех возможных операций в одном алгоритме

4. Цикл с постусловием завершает работу, когда условие становится:

A) истинным

B) ложным

C) равным нулю

D) неопределенным

5. Циклическая часть (тело цикла)- это \_\_\_\_\_.

- A) Группа команд, не входящих в циклическую структуру.
- B) Произвольный текст.
- C) Произвольная группа команд.
- D) Группа команд, повторяющихся некоторое число раз.

6. Следующий фрагмент программы

```
ЕСЛИ X < Y ТО
    ЕСЛИ X < Z ТО
        M := X
    ИНАЧЕ
        M := Z
ИНАЧЕ
    ЕСЛИ Y < Z ТО
        M := Y
    ИНАЧЕ M := Z
```

вычисляет:

- A) минимум из трех чисел
- B) наименьшее из чисел Y и Z
- C) максимум из трех чисел
- D) наибольшее из чисел X и Y

7. Инструментальные средства разработки программ-это

- A) Средства создания новых программ
- B) Сервисные средства разработки ПО
- C) Аналитические средства разработки ПО

8. Интерпретатор

- A) Программа для создания и редактирования электронных таблиц
- B) Переводит в коды сразу всю программу и создает независимый исполняемый файл
- C) Переводит в машинные коды одну строку программы и сразу ее выполняет
- D) Программа для создания и редактирования текстовых документов

9. Компилятор

- A) Прикладное ПО
- B) Специальная утилита системного ПО
- C) Операционная оболочка
- D) Переводит в коды сразу всю программу и создает независимый исполняемый файл

10. Моделирование основывается на принципах

- A) Выбор модели оказывает определяющее влияние на подход к решению проблемы и на то, как будет выглядеть это решение
- B) Декомпозиция системы на отдельные подзадачи
- C) Инкапсуляция и полиморфизма
- D) Децентрализации управления системой

11. Транслятор - это

- A) Программа, выполняющая перевод программы с одного языка программирования на другой
- B) Комплекс программ мультимедийных технологий

- С) Программа-переводчик с одного иностранного языка на другой
- D) Техническое устройства передачи и преобразования аудио и видеосигналов
- E) Техническое устройство для кодирования и декодирования информации

12. Что будет выведено следующей программой

```
A = B = 1
a = "A" * 5
b = "B" * 5
a = a + b * 5
print (a, b)
```

- A) 30 5
- B) AAAAABBBBBAAAAABBBBBAAAAABBBBBAAAAABBBBB, BBBBB
- C) AAAAABBBBBAAAAABBBBBAAAAABBBBBAAAAABBBBBAAAAABBBBB BBBBB
- D) AAAAABBBBBBBBBBBBBBBBBBBBBBBBBBBBB BBBBB

13. Что будет выведено следующей программой

```
a = "A"
b = "B"
b = b + a
print(a + b)
```

- A. BA
- B. ABA
- C. сообщение об ошибке в третьей строке
- d. AB

14. Что выведет следующая программа

```
S = 0
for i in range(1, 10):
    if i % 2 == 0:
        S = S + i
print(S)
```

Select one:

- A. 12
- B. 10
- C. 14
- B. 20

15. Набор методов по упорядочиванию данных, с целью наиболее простого и ясного восприятия этих данных называют

- a. методом оценивания
- b. методом проверки гипотез
- c. регрессионным анализом
- d. описательной статистикой

Образцы заданий к семинарским занятиям:

1. Дана сторона квадрата  $a$ . Найти его периметр  $P = 4 \cdot a$ .
2. Дана сторона квадрата  $a$ . Найти его площадь  $S = a^2$ .
3. Даны стороны прямоугольника  $a$  и  $b$ . Найти его площадь  $S = a \cdot b$  и периметр  $P = 2 \cdot (a + b)$ .

4. Дан диаметр окружности  $d$ . Найти ее длину  $L = \pi \cdot d$ . В качестве значения  $\pi$  использовать 3.14.
5. Дана длина ребра куба  $a$ . Найти объем куба  $V = a^3$  и площадь его поверхности  $S = 6 \cdot a^2$ .
6. Даны длины ребер  $a, b, c$  прямоугольного параллелепипеда. Найти его объем  $V = a \cdot b \cdot c$  и площадь поверхности  $S = 2 \cdot (a \cdot b + b \cdot c + a \cdot c)$ .
7. Найти длину окружности  $L$  и площадь круга  $S$  заданного радиуса  $R$ :  $L = 2 \cdot \pi \cdot R$ ,  $S = \pi \cdot R^2$ . В качестве значения  $\pi$  использовать 3.14.
8. Даны два числа  $a$  и  $b$ . Найти их среднее арифметическое:  $(a + b)/2$ .
9. Даны два неотрицательных числа  $a$  и  $b$ . Найти их среднее геометрическое, то есть квадратный корень из их произведения:  $\sqrt{a \cdot b}$ .
10. Даны два ненулевых числа. Найти сумму, разность, произведение и частное их квадратов.

### **Индивидуальные творческие задания (проекты):**

*Описание проектного задания*

Дан файл в формате .csv, содержащий данные из определенной предметной области.

Необходимо:

1. Определить наличие пропущенных и невалидных данных и провести их коррекцию;
2. Построить описательную статистику и корреляцию по числовым данным;
3. Выполнить категоризацию качественных данных;
4. Разделить набор данных на группы используя методы агрегирования данных;
5. Выполнить визуализацию полученных данных;

Отчет по проектному заданию:

1. Постановка задачи
2. Код и результаты выполнения заданий
3. Выводы

Варианты проектных заданий:

1. iris.csv данные о цветках ириса
2. bank.csv данные о банковской деятельности
3. games.csv данные о компьютерных играх
4. brilliant.csv данные о бриллиантах
5. edu.csv данные о статистике по образовательным учреждениям РФ.

## **7. Учебно-методическое и информационное обеспечение дисциплины**

а) основная литература:

1. Васильев, А. Н., Программирование на Python в примерах и задачах / Васильев А. Н. - М.: Эксмо, 2021. – 614 с.
2. Кольцов, Д. М. Python. Полное руководство / Кольцов Д. М. - СПб.: Наука и техника, 2022. - 475 с.

б) дополнительная литература:

1. Скиена, С. Наука о данных: учеб. курс. / Стивен С. Скиена; пер. с англ. и ред. В. А. Коваленко. - СПб.: Диалектика, 2020. – 543.с.

2. Дейтел, Пол Дж. Python. Искусственный интеллект, большие данные и облачные вычисления / Пол Дейтел, Харви Дейтел; пер. с англ. Е. Матвеева. - СПб. [и др.]: Питер, 2020. - 861, [1] с

3. Мартин, Роберт К. Идеальный программист. Как стать профессионалом разработки ПО: [пер. с англ.]. / Роберт Мартин. - СПб. [и др.]: Питер, 2019. - 214, [1] с.

4. Плас, Джейк Вандер Python для сложных задач. Наука о данных и машинное обучение / Дж. Вандер Плас; [пер. с англ. И. Пальти]. - СПб. [и др.]: Питер, 2020. - 572, [2] с.

в) программное обеспечение и Интернет-ресурсы

1. Система федеральных образовательных порталов. Информационно-коммуникационные технологии в образовании. <http://www.ict.edu.ru/lib/>

2. Интернет университет информационных технологий. <http://www.intuit.ru/>

3. Система федеральных образовательных порталов. Информационно-коммуникационные технологии в образовании. <http://www.ict.edu.ru/lib/>

4. Российская национальная библиотека (РНБ). [www. hbl-russia.ru](http://www.hbl-russia.ru) <http://www.nlr.ru>.

5. Российская государственная библиотека (РГБ). [http://www. rsl. ru](http://www.rsl.ru).

6. ЭБС «Университетская библиотека онлайн» <http://www.biblioclub.ru>

7. ЭБС «Znanium.com» <http://znanium.com/>

8. ЭБС «Юрайт» <https://biblio-online.ru/>

## **8. Материально-техническое обеспечение дисциплины**

Для освоения дисциплины требуется свободный доступ к сети Интернет, а также:

А. Помещения: аудитория для проведения лекционных и семинарских занятий, текущего контроля и промежуточной аттестации, компьютерный класс.

Б. Оборудование: наборы ученической мебели, рабочее место преподавателя, ученическая доска, персональные компьютеры, проекторы, экраны, доска интерактивная.

### **Разработчики:**

Научный сотрудник А. В. Якушин, факультет ВМиК МГУ.