



*ПРИНЯТО*

на заседании Педагогического совета

Протокол №

от

*УТВЕРЖДЕНО*

приказом директора

АНО ОШ ЦПМ

от №

## **РАБОЧАЯ ПРОГРАММА**

по предмету **«Информатика»**

для обучающихся 10-11 классов

(углублённый уровень)

для очно-заочной формы обучения

на 2023 – 2024 учебный год

Составители:

В. В. Быкова, Д. Д. Врубель

Москва, 2023 год

## Оглавление

Планируемые результаты освоения учебного предмета	3
Содержание учебного предмета	5
Тематическое планирование учебного предмета	10

## Планируемые результаты освоения учебного предмета

### Обучающийся научится:

- Работать с переменными на языке Python. Использовать арифметические операции с целыми и вещественными числами;
- Считывать входные данные. Выводить данные;
- Использовать и применять условные операторы if. Использовать и применять условные операторы elif и else;
- Использовать цикл while. Обрабатывать последовательности. Использовать однопроходные алгоритмы. Искать минимум и максимум в последовательности. Прерывать работу цикла с помощью break и continue;
- Использовать цикл for. Использовать range для перебора чисел в диапазоне. Обрабатывать и генерировать последовательности с помощью цикла for. Прерывать работу цикла с помощью break и continue;
- Работать с вещественными значениями. Работать с точностью вещественных чисел. Представлять вещественные числа с помощью мантиссы и экспоненты.
- Считывать и обрабатывать строки на языке Python. Обрабатывать тексты и строки с помощью встроенных методов и функций. Использовать срезы строк;
- Считывать и работать со списками, обрабатывать их с помощью встроенных функций. Добавлять новые элементы в конец списка, получать срезы списка. Использовать кортежи;
- Использовать вложенные списки и работать с ними или с многомерными списками. Сохранять двумерные данные или таблицу в программе. Считывать и выводить двумерные данные;
- Использовать функции в языке Python и создавать собственные. Работать с локальными и глобальными переменными;
- Использовать рекурсию и применять ее, находить числа Фибоначчи с использованием рекурсии. Настраивать лимит рекурсии в Python;
- Сортировать списки и другие структуры. Использовать собственные ключи для сортировки, использовать lambda-функции;
- Использовать и применять структуру данных «множество» в языке Python. Добавлять и удалять элементы в структуре данных «множество», проверять, лежит ли элемент во множестве. Создавать множество из других структур данных. Объединять и пересекать множества;
- Использовать и применять структуру данных «словарь». Создавать словарь из других структур данных. Добавлять и удалять пары «ключ-значение» в словарь, искать ключи в словаре. Получать ключи и пары «ключ-значения» из словаря как список;
- Оценивать сложность алгоритма с помощью O-нотации, оценивать скорость работы алгоритма и использование памяти в нем;
- Использовать встроенные функции map, enumerate, zip и другие. Применять их в типичных случаях для упрощения понимания кода и его длины;
- Создавать собственные классы в языке Python. Создавать и использовать поля и методы классов. Использовать метод создания класса \_\_init\_\_.
- Различать понятия «поле класса» и «поля объекта класса». Переопределять операции на классах. Использовать наследование классов в языке Python;
- Импортировать и использовать библиотеки в языке Python. Использовать библиотеки math, datetime, collections и copy.
- Эффективно работать с одномерными массивами с помощью библиотеки numpy. Использовать стандартные методы и функции для обработки массивов numpy и их анализа;



- Генерировать случайные числа с помощью библиотеки numpy. Генерировать значения из разных распределений, создавать случайные списки;
- Работать с массивом данных pandas.Series, применять эффективные функции к нему;
- Эффективно работать с датафреймами данных при помощи библиотеки pandas. Использовать стандартные методы и функции для обработки дата фреймов. Создавать, считывать и сохранять датафреймы из/в разных форматах. Анализировать данные датафреймов, обрабатывать пропуски в данных;
- Рисовать линейные графики с помощью библиотеки matplotlib. Настраивать параметры графика: легенду, размер, цвет. Сохранять рисунки и загружать их;
- Основным способом обработки естественного языка при помощи Python;
- Основам машинного и глубокого обучения;
- Классифицировать и кластеризовать данные;
- Работать над задачами программирования в команде.

## Содержание учебного предмета

### Раздел 1. Основы программирования на языке Python.

#### Тема 1.1. Понятие программирования и языка программирования.

- Что такое программирование и какие задачи оно решает;
- Прикладное программирование в смежных науках;
- Установка среды программирования и работа в ней.

#### Тема 1.2. Переменные и типы данных.

- Типы данных: целые числа и строки;
- Ввод-вывод чисел и строк;
- Операции сложения, вычитания, умножения, деления, целочисленного деления, возведения в степень и взятия остатка.

#### Тема 1.3. Условный оператор и алгебра логики.

- Переменные логического типа;
- Условный оператор if, else и elif;
- Логические связки and, or и not;
- Вложенные условные операторы;
- Алгебра логики

#### Тема 1.4. Вещественные числа.

- Хранение вещественных чисел в памяти;
- Операции над вещественными числами;
- Точность операций над вещественными числами;
- Представление вещественных чисел с использованием мантиссы.

#### Тема 1.5. Цикл while.

- Понятие цикла. Цикл while;
- Использование break/continue в цикле;
- Решение задач с использованием цикла while.

#### Тема 1.6. Цикл for.

- Понятие цикла for;
- Использование range в циклах;
- Вложенные циклы;
- Решение задач с использованием цикла for.

#### Тема 1.7. Строки.

- Операции над строками;
- Многострочный текст;

- Срезы;
- Встроенные методы для работы со строками;

### Тема 1.8. Списки и кортежи.

- Понятие списков и кортежей;
- Хранение списка в памяти;
- Считывание списков;
- Добавление элементов в список;
- Срезы в списках;
- Решение задач с использованием списков.
- Работа с кортежами

### Тема 1.9. Вложенные списки.

- Создание вложенных списков;
- Считывание вложенных списков;
- Решение задач на вложенные списки.

### Тема 1.10. Простые алгоритмы.

- Алгоритм Евклида;
- Поиск НОД и НОК;
- Проверка на простоту;
- Разложение числа на множители.

### Тема 1.11. Функции.

- Понятие функции;
- Передача параметров в функцию в Python. Передача параметров по ссылке;
- Значения аргументов по умолчанию;
- Локальные и глобальные переменные;
- Использование функций в решении задач.

### Тема 1.12. Рекурсия.

- Понятие рекурсии;
- Решение задач на рекурсию.

### Тема 1.13. Обработка файлов.

- Чтение из файлов;
- Запись в файл;
- Библиотеки os и pathlib;
- Основы работы с бинарными файлами;
- Кодировки.

### Тема 1.14. Понятие асимптотики. Алгоритмы сортировки.

- Сложность алгоритма и его время работы;
- O-нотация;
- Время работы пройденных функций и методов;
- Сортировка пузырьком;
- Сортировка слиянием;
- Быстрая сортировка;
- Решение задач при помощи алгоритмов.

### Тема 1.15. Бинарный поиск.

- Алгоритм бинарного поиска;
- Бинарный поиск по ответу;
- Решение задач при помощи алгоритмов.

### Тема 1.16. Множества.

- Понятие хеширования;
- Структура данных «множество»;
- Создание множества из других структур данных;
- Добавление, поиск и удаление во множествах;
- Объединение и пересечение множеств;
- Решение задач на множествах
- frozenset.

### Тема 1.17. Словари.

- Структура данных «словарь». Понятие «ключ-значение»;
- Создание словарей;
- Поиск в словаре, добавление и удаление элементов;
- Проход по всем значениям словаря;
- Использование defaultdict;
- Решение задач на словари.

### Тема 1.18. Классы.

- Что такое классы и зачем они нужны;
- Простейшее создание классов в языке Python;
- Методы и поля класса, использование \_\_init\_\_.
- Поля класса и поля объекта класса;
- Переопределение операций над классом: \_\_add\_\_;
- Наследование. Простейшее понимание;
- Решение задач на классы;

### Тема 1.19. Стандартная библиотека Python.

- Что такое модули и пакеты и зачем они нужны;
- Установка библиотек;
- Модули math, datetime и другие.
- Модуль collections;
- Модуль сорю;

- Map, enumerate, zip и другие;
- Паттерны использования встроенных функций;
- Решение задач с помощью модулей.

## Раздел 2. Использование языка Python для обработки данных.

### Тема 2.1. Графы и простые алгоритмы для работы с ними. Деревья.

- Бинарные деревья;
- Куча;
- Сортировка кучей;
- Алгоритмы на графах;

### Тема 2.2. Работа с форматом json.

- Модуль json;
- Сохранение и запись в файлы json.

### Тема 2.3. Работа в Jupiter, Google colab.

- Особенности работы с расширением .ipynb
- Оформление кода, Markdown.

### Тема 2.4. Морфологические анализаторы.

- Работа с языковым материалом при помощи nltk, pymorphy, mystem.

### Тема 2.5. Регулярные выражения.

- Библиотека re;
- Практика написания регулярных выражений.

### Тема 2.6. Парсинг данных.

- Автоматическое извлечение данных с сайтов;
- Работа при помощи beautifulsoup4.

### Тема 2.7. numpy.

- Массивы numpy – одномерные и многомерные;
- Создание массивов – array, arange, linspace, ones, zeros, eye;
- Математические операции над массивами;
- Операции над массивами разных размерностей;
- Сравнение numpy array и list;
- Размер массива – shape. Метод reshape;



- Объединение массивов – concatenate;
- Применение функций к элементам массива: np.sqrt, np.log и другие;
- Применение функций к массиву: np.sum, np.prod и другие.
- Множественное присваивание;
- Получение элементов через маску;
- Функции any и all.

## Тема 2.8. Pandas

- Создание Series, параметры при создании;
- Создание Series из других структур данных;
- Работа с элементами Series, связь со словарем и numpy array.
- Создание Dataframe и зачем он нужен;
- Скачивание и сохранение Dataframe-ов;
- Работа со строками и столбцами Dataframe-ов;
- Работа со строками и столбцами Dataframe-ов. Продолжение;
- Обработка пропусков данных в Dataframe.

## Тема 2.9. Визуализация данных.

- Знакомство с библиотеками matplotlib и seaborn;
- Построение графиков.

## Раздел 3. Работа с терминалом Linux и системой контроля версий git.

### Тема 3.1. Основы работы в терминале.

- Запуск терминала;
- Команды в терминале;
- Запуск исполняемых файлов;
- Управляющие операторы &&, || и другие;
- Команды для управления файлами: ls, cat, cd, pwd, mkdir, cp, mv, rm, rmdir.
- Команды для работы с текстом: less/more, head/tail, grep, sort, wc, diff, sed.
- Установка пакетов при помощи терминала.

### Тема 3.2. Работа с системой контроля версий git.

- Платформы gitlab и github;
- Совместная работа над одним проектом при помощи git.

### Тема 4.1. Теоретический обзор классического машинного обучения.

- Задачи машинного обучения;
- Задачи классификации, регрессии, кластеризации;
- Оценка работы модели;

### Тема 4.2. Линейная классификация и кластеризация.

- Библиотека scikit-learn;
- Предобработка данных;

- Практика по теме.

### **Тема 4.3. Нейронные сети (теория).**

- Отличия глубокого и машинного обучения;
- Понятие градиентного спуска;
- Оценка качества модели;
- Обработка различных видов данных.

### **Тема 4.4. Основные задачи NLP сегодня**

- Распознавание речи;
- Анализ текста;
- Анализ тональности текста;
- Вопросно-ответные системы;
- Генерирование текста;
- Синтез речи;
- Машинный перевод.



## Тематическое планирование учебного предмета

Раздел/тема	Количество академических часов
<b>Раздел 1. Основы программирования на языке Python. Основы алгоритмов. 1 семестр</b>	<b>68 часов</b>
Тема 1.1. Понятие программирования и языка программирования.	1
Тема 1.2. Переменные и типы данных	1
Тема 1.3. Условный оператор и алгебра логики	2
Тема 1.4. Вещественные числа	4
Тема 1.5. Цикл while	3
Тема 1.6. Цикл for	3
Тема 1.7. Строки	4
Тема 1.8. Списки и кортежи	4
Тема 1.9. Вложенные списки	4
Тема 1.10. Простые алгоритмы	4
<b>Промежуточное тестирование</b>	<b>2</b>
Тема 1.11. Функции	6
Тема 1.12. Рекурсия	4
Тема 1.13. Обработка файлов	4



Тема 1.14. Понятие асимптотики. Алгоритмы сортировки	2
Тема 1.15. Бинарный поиск	2
Тема 1.16. Множества	2
Тема 1.17. Словари	4
Тема 1.18. Классы	4
Тема 1.19. Стандартная библиотека Python.	6
<b>Контрольная работа по разделу</b>	<b>2</b>
<b>Раздел 2. Использование языка Python для обработки данных. Визуализация.</b>	<b>34 часа</b>
Тема 2.1. Графы и простые алгоритмы для работы с ними. Деревья.	6
Тема 2.2. Работа с форматом json	2
Тема 2.3. Работа в Jupiter, Google colab	1
Тема 2.4. Морфологические анализаторы	2
Тема 2.5. Регулярные выражения	4
Тема 2.6. Парсинг данных	4
Тема 2.7. Библиотека numpy	4
Тема 2.8. Библиотека pandas	4
Тема 2.9 Визуализация данных.	5
<b>Контрольная работа по разделу</b>	<b>2</b>



<b>Раздел 3. Работа с терминалом Linux и системой контроля версий git</b>	<b>6 часов</b>
Тема 3.1. Работа с командной строкой	3
Тема 3.2. Работа с git	3
<b>Раздел 4. Машинное обучение, глубокое обучение, вопросы современного NLP</b>	<b>14 часов</b>
Тема 4.1. Теоретический обзор классического машинного обучения	4
Тема 4.2. Линейная классификация и кластеризация	2
Тема 4.3. Нейронные сети (теория)	4
Тема 4.4. Основные задачи NLP сегодня	4
<b>Работа над итоговым проектом</b>	<b>14 часов</b>
Работа над проектом	5
Предзащита проекта	2
Доработка проекта с учетом предложенных правок	5
Защита проекта	2
<b>Итого</b>	<b>136</b>