

**Приложение 1.2.8**  
к основной образовательной программе  
среднего общего образования  
МБОУ СОШ № 10  
от 28.08.2024 № 227

**РАБОЧАЯ ПРОГРАММА**  
учебного курса  
«Программирование»  
10-11 класс

## РАЗДЕЛ I. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ

Программа позволяет добиться следующих результатов:

### ***Первый уровень***

*В результате изучения курса на первом уровне учащийся*

- 1) научится составлять и отлаживать простые диалоговые программы;
- 2) узнает особенностей машинных вычислений с целыми и вещественными числами;
- 3) научится использовать основные алгоритмические конструкции: условные операторы, циклы с условием, циклы по переменной;
- 4) овладеет методами построения графических изображений программными средствами;
- 5) овладеет простыми методами программирования компьютерной анимации.

### ***Второй уровень***

*В результате изучения курса на втором уровне учащийся*

- 6) познакомится с методами проектирования программ «сверху вниз» и «снизу вверх»;
- 7) научится использовать вспомогательные алгоритмы (процедуры и функции) для структуризации программ;
- 8) научится применять рефакторинг для улучшения читаемости программ;
- 9) научится использовать символные строки;
- 10) овладеет основными алгоритмами обработки одномерных и двумерных массивов;
- 11) познакомится с понятием сложности алгоритма;

### ***Третий уровень***

*В результате изучения курса на третьем уровне учащийся*

- 12) научится применять различные алгоритмы сортировки массивов;
- 13) научится использовать двоичный поиск;
- 14) научится обрабатывать данные, записанные в текстовые и двоичные файлы, и сохранять в файлах результаты работы программы;
- 15) научится использовать структуры для объединения данных;
- 16) научится применять словари, стеки, очереди, деки для решения задач обработки данных;
- 17) научится использовать деревья для организации данных;

- 18) познакомится с методами описания графов и некоторыми популярными алгоритмами на графах;
- 19) научится использовать динамическое программирование для решения комбинаторных и оптимизационных задач;
- 20) познакомится с понятием выигрышных и проигрышных позиций в играх с полной информацией;

#### **Четвёртый уровень**

*В результате изучения курса на четвёртом уровне учащийся*

- 21) познакомится с объектно-ориентированным подходом к разработке программ;
- 22) научится выполнять объектно-ориентированный анализ задачи, выделять свойства и методы объектов;
- 23) научится использовать инкапсуляцию для защиты данных объектов;
- 24) познакомится с понятиями «класс» и «абстрактный класс»;
- 25) познакомится с понятиями «инкапсуляция», «наследование», «полиморфизм»;
- 26) научится проектировать несложные иерархии классов для прикладных задач;
- 27) познакомится с принципами разработки событийно-ориентированных программ;
- 28) научится создавать программы с графическим интерфейсом на языках Python и C#;
- 29) научится использовать готовые и создавать новые компоненты (виджеты) для сред быстрой разработки программ.

## РАЗДЕЛ II. СОДЕРЖАНИЕ УЧЕБНОГО ПРЕДМЕТА

### *Первый уровень*

#### **Раздел 1. Программирование на языке Python**

Простейшие программы. Диалоговые программы. Переменные. Консольный ввод и вывод данных.

Компьютерная графика. Система координат. Управление пикселями. Графические примитивы: линии, прямоугольники, окружности. Изменение координат. Анимация.

Процедуры. Процедуры с параметрами. Рефакторинг.

Обработка целых чисел. Арифметические выражения. Деление нацело. Остаток от деления.

Обработка вещественных чисел. Особенности представления вещественных чисел в памяти компьютера. Операции с вещественными числами.

Случайные и псевдослучайные числа. Генераторы случайных чисел.

Ветвления. Условный оператор. Полная и неполная формы условного оператора. Вложенные условные операторы. Логические переменные. Экспертные системы. Сложные условия. Логические операции И, ИЛИ, НЕ. Порядок выполнения операций.

Циклы с условием. Алгоритм Евклида. Обработка потока данных. Бесконечные циклы. Циклы по переменной. Шаг изменения переменной цикла.

Циклы в компьютерной графике. Узоры. Вложенные циклы. Штриховка.

#### **Раздел 2. Программирование на языке C++**

Структура программы на языке C++. Компиляция программы. Препроцессор. Директива *include*.

Вывод текста на экран. Диалоговые программы. Ввод и вывод данных.

Компьютерная графика. Библиотека `TX Library`. Управление пикселями. Линии и фигуры. Замкнутые фигуры.

Процедуры. Процедуры с параметрами. Рефакторинг.

Обработка целых чисел. Ограниченность значений целых чисел. Арифметические выражения. Деление и остаток.

Обработка вещественных чисел. Операции с вещественными числами.

Случайные и псевдослучайные числа.

Ветвления. Условный оператор. Вложенные условные операторы. Логические переменные. Сложные условия.

Цикл с предусловием. Циклы с постусловием. Циклы по переменной.

Анимация. Обработка нажатия клавиш.

### *Второй уровень*

#### **Раздел 1. Программирование на языке Python**

Этапы создания программ. Методы проектирования программ «сверху вниз» и «снизу вверх». Интерфейс и реализация. Документирование программы.

Подпрограммы: процедуры и функции. Процедуры. Процедуры с параметрами. Локальные и глобальные переменные.

Функции. Логические функции.

Рекурсия. Рекурсивные процедуры и функции. Фракталы.

Символьные строки. Сравнение строк. Операции со строками. Обращение к символам. Перебор всех символов. Срезы. Удаление и вставка. Встроенные методы. Поиск в символьных строках. Замена символов. Преобразования «строка — число». Символьные строки в функциях. Рекурсивный перебор.

Массивы (списки). Массивы в языке Python. Создание массива. Обращение к элементу массива. Перебор элементов массива. Генераторы. Вывод массива. Ввод массива с клавиатуры. Заполнение массива случайными числами.

Алгоритмы обработки массивов. Сумма элементов массива. Подсчёт элементов массива, удовлетворяющих условию. Особенности копирования списков в Python.

Поиск в массивах. Линейный поиск. Поиск максимального элемента в массиве. Максимальный элемент, удовлетворяющий условию. Использование массивов в прикладных задачах.

Матриц. Создание и заполнение матриц. Вывод матрицы на экран. Перебор элементов матрицы. Квадратные матрицы.

Сложность алгоритмов. Асимптотическая сложность.

## **Раздел 2. Программирование на языке C++**

Процедуры. Процедуры с параметрами. Локальные и глобальные переменные. Процедуры, изменяющие аргументы. Рекурсивные процедуры. Построение простых фракталов.

Функции в C++. Логические функции. Рекурсивные функции.

Символьные строки. Сравнение строк. Сцепление строк. Обращение к символам. Перебор всех символов. Подстрока. Удаление и вставка. Поиск в символьных строках. Замена символов. Преобразования «строка — число». Символьные строки в функциях. Рекурсивный перебор.

Массивы в C++. Обращение к элементу массива. Перебор элементов массива. Вывод массива. Ввод массива с клавиатуры. Заполнение массива случайными числами. Алгоритмы обработки массивов. Использование массивов в прикладных задачах.

Матрицы. Размещение матрицы в памяти. Заполнение матрицы. Вывод матрицы на экран. Обработка матриц.

Системы управления версиями. Основные приёмы работы с Git. Операции с файлами. Восстановление версии. Работа с удалённым архивом. Ветки. Графические оболочки для Git.

### *Третий уровень*

## **Раздел 1. Программирование на языке Python**

Сортировка массивов. Метод пузырька (сортировка обменами). Метод выбора. Сортировка слиянием. Быстрая сортировка. Сортировка в языке Python.

Двоичный поиск в массиве данных. Двоичный поиск по ответу. Обработка файлов. Типы файлов. Чтение данных. Запись данных. Обработка данных из файла.

Целочисленные алгоритмы. Решето Эратосфена. Целочисленный квадратный корень.

Словари. Алфавитно-частотный словарь. Перебор элементов словаря.

Структуры. Классы. Создание структур. Работа с полями структур. Хранение структур в файлах. Сортировка структур.

Стек. Использование списка. Вычисление арифметических выражений. Скобочные выражения. Системный стек. Очередь. Дек.

Деревья. Деревья поиска. Обход дерева. Использование связанных структур. Вычисление арифметических выражений.

Графы. Описание графа. Жадные алгоритмы. Минимальное остовное дерево. Алгоритм Дейкстры. Алгоритм Флойда–Уоршелла. Использование списков смежности.

Динамическое программирование. Числа Фибоначчи. Количество программ для исполнителя. Двумерные задачи. Поиск оптимального решения.

Игровые модели. Выигрышные и проигрышные позиции.

## **Раздел 2. Программирование на языке C++**

Сортировка массивов. Метод пузырька (сортировка простыми обменами). Сортировка вставками. Массивы в подпрограммах. Сортировка слиянием. Быстрая сортировка. Стандартная сортировка в языке C++. Двоичный поиск.

Обработка файлов. Файловые потоки. Обработка данных из файла. Чтение текстовых файлов по словам. Построчная обработка файлов. Аргументы основной программы.

Целочисленные алгоритмы. Решето Эратосфена. «Длинные» числа.

Динамические массивы. Тип `vector` из библиотеки STL. Итераторы.

Словари. Перебор элементов словаря.

Структуры в C++. Обращение к полям структуры. Хранение структур в файлах. Сортировка структур.

Стек. Очередь. Хранение очереди в массиве. Дек.

Деревья в C++. Обходы дерева. Деревья поиска. Вычисление арифметических выражений. Хранение дерева в массиве.

Графы в языке C++. Задача коммивояжера. Жадные алгоритмы. Случайные перестановки. Передача данных по ссылке.

Динамическое программирование. Одномерные задачи. Редактирование строк. Оптимальная стратегия.

## **Четвёртый уровень**

### **Раздел 1. Программирование на языке Python**

Проблема сложности программ. Процедурный и объектно-ориентированный подходы к написанию программ.

Классы и объекты. Объектно-ориентированный анализ. Взаимодействие объектов. Свойства и методы.

Классы и объекты в программе. Объявление класса. Поля класса. Конструктор класса. Данные и методы класса.

Скрытие внутреннего устройства. Доступ к полям через методы. Свойства (property). Свойство «только для чтения»

Иерархия классов. Наследование. Базовый класс. Доступ к полям. Классы-наследники. Полиморфизм. Разработка модулей.

Событийно-ориентированное программирование. Программы с графическим интерфейсом. Форма. Свойства формы. Обработчики событий.

Использование компонентов (виджетов). Ввод и вывод данных. Обработка ошибок с помощью исключений.

Создание компонентов. Добавление свойств и методов. Составные компоненты.

Модель и представление.

## **Раздел 2. Программирование на языке C++**

Классы и объекты в языке C++. Объектно-ориентированный анализ задачи. Конструкторы классов. Разбиение на модули.

Инкапсуляция. Возможность изменения внутреннего устройства объектов. Свойства «только для чтения».

Наследование. Иерархия классов. Базовый класс. Абстрактный класс. «Чистые» виртуальные методы. Защищённые поля и методы (protected).

Полиморфизм. Указатели на базовый класс. Виртуальные методы. Позднее связывание. Деструктор.

Организация взаимодействия объектов.

«Умные» указатели.

RAD-среды для разработки программ. Язык C# и среда .NET. Проект в C#. Свойства объектов. Обработчики событий.

Использование компонентов. Ввод и вывод данных. Обработка ошибок с помощью исключений.

Создание новых классов. Статические методы класса. Создание новых компонентов.

### РАЗДЕЛ III. ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ,

в том числе с учетом рабочей программы воспитания, \*с указанием количества часов, отводимых на освоение каждой темы

*\*Курсивом выделены темы, разработанные с учетом рабочей программы воспитания*

**10**

**класс**

№ раздела (темы)	Название раздела (темы)	Кол-во часов
<b>1. Первый уровень ( Программирование на языке Python)</b>		<b>17</b>
1	Первые программы	1
2	Диалоговые программы	1
3	Компьютерная графика	1
4	Процедуры	1
5	Обработка целых чисел	1
6	Обработка вещественных чисел	1
7	Случайные и псевдослучайные числа	1
8	Ветвления <i>День Интернета. Всероссийский урок безопасности школьников в сети Интернет</i>	1
9	Сложные условия	1
10	Циклы с условием	1
11	Циклы с условием: практикум	1
12	Анимация	1
13	Циклы по переменной	1
14	Циклы в компьютерной графике	1
15	Выполнение проекта часть 1	1
16	Выполнение проекта часть 2.	1
17	Защита проекта	1
<b>2. Второй уровень (Программирование на языке Python)</b>		<b>17</b>
18	Проектирование программ	1
19	Процедуры	1
20	Рекурсия <i>День российской науки.</i>	1
21	Функции	1
22	Символьные строки	1



23	Обработка символьных строк	1
24	Строки в функциях	1
25	Массивы	1
26	Ввод и вывод массивов	1
27	Суммирование элементов массива	1
28	Подсчёт элементов массива, удовлетворяющих условию	1
29	Поиск значения в массиве	1
30	Поиск максимального элемента в массиве	1
31	Игра «Стрельба по тарелкам»	1
32	Игра «Стрельба по тарелкам»	1
33	Матрицы	1
34	Сложность алгоритмов	1
	<b>Итого:</b>	<b>34</b>

**11**

**класс**

<b>№ раздела (темы)</b>	<b>Название раздела (темы)</b>	<b>Кол-во часов</b>
<b>1. Третий уровень (Программирование на языке Python)</b>		<b>18</b>
1	Простые алгоритмы сортировки	1
2	Сортировка слиянием	1
3	Быстрая сортировка	1
4	Двоичный поиск	1
5	Обработка файлов	1
6	Обработка файлов: практикум	1
7	Целочисленные алгоритмы	1
8	Словари <i>День Интернета. Всероссийский урок безопасности школьников в сети Интернет</i>	1
9	Структуры	1
10	Структуры: практикум	1
11	Стек, очередь, дек	1
12	Деревья	1
13	Графы	1
14	Графы: практикум	1
15	Динамическое программирование	1

16	Динамическое программирование: практикум	1
17	Игровые модели	1
18	Игровые модели: практикум	
<b>2. Четвертый уровень (Программирование на языке Python)</b>		<b>16</b>
19	Что такое ООП? <i>День российской науки.</i>	1
20	Модель задачи: классы и объекты	1
21	Классы и объекты в программе	1
22	Классы и объекты в программе: практикум	1
23	Скрытие внутреннего устройства	1
24	Иерархия классов	1
25	Классы-наследники (I)	1
26	Классы-наследники (II)	1
27	Доработка игры	1
28	Событийно-ориентированное программирование	1
29	Использование компонентов (виджетов)	1
30	Использование компонентов (виджетов)	1
31	Создание компонентов	1
32	Модель и представление	1
33	Выполнение проекта часть 1	1
34	Выполнение проекта часть 2.	1
	<b>Итого:</b>	<b>34</b>