

РАБОЧАЯ ПРОГРАММА
курса внеурочной деятельности
«Программирование»
7–9 классы
170 часов

Оглавление

1. Пояснительная записка	3
2. Содержание курса внеурочной деятельности «Программирование»	4
7 класс	4
8 класс	5
9 класс	6
3. Планируемые результаты освоения курса внеурочной деятельности «Программирование»	7
4. Тематическое планирование курса внеурочной деятельности «Программирование» с указанием ЭОР	8
7 класс	8
8 класс	9
9 класс	10
5. Форма организации учебной деятельности	11

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Рабочая программа курса внеурочной деятельности «Программирование» разработана для реализации в 7—9 классах образовательных организаций, подведомственных Департаменту образования и науки города Москвы, городского образовательного проекта «ИТ-вертикаль». Программой курса предусмотрено изучение двух языков программирования: Python и C++.

Программа составлена на основе требований Федерального государственного образовательного стандарта основного общего образования к результатам освоения основной программы основного общего образования (Приказ Министерства просвещения Российской Федерации от 31.05.2021 № 287 «Об утверждении федерального государственного образовательного стандарта основного общего образования»), с учётом Примерной программы воспитания (протокол Федерального учебно-методического объединения по общему образованию № 3/22 от 23.06.2022) и Приказа Департамента образования и науки города Москвы «Об утверждении стандарта(ов) городских образовательных проектов ».

Программа курса даёт представления о целях, задачах, общей стратегии обучения, воспитания и развития обучающихся средствами курса внеурочной деятельности «Программирование», устанавливает содержание курса, предусматривает его структурирование по разделам и темам; предлагает распределение учебных часов по разделам и темам курса, а также последовательность их изучения с учётом межпредметных и внутрипредметных связей, логики учебного процесса, возрастных особенностей обучающихся, включает описание форм организации занятий.

Программа определяет количественные и качественные характеристики учебного материала для каждого года изучения, в том числе планируемые результаты освоения обучающимися программы курса внеурочной деятельности на уровне основного общего образования.

Общая характеристика курса внеурочной деятельности «Программирование»

В современном мире программирование играет ключевую роль и имеет широкий спектр применения:

- разработка программного обеспечения;
- автоматизированная обработка данных и автоматизация задач;
- кибербезопасность;
- инженерия и робототехника и др.

Целью освоения программы курса внеурочной деятельности «Программирование» является:

- овладение основами алгоритмизации и программирования на языках Python и C++;
- приобретение навыков разработки программного кода с использованием современных инструментальных средств для платформ.

Основными задачами курса является:

- формирование знаний, умений, навыков и компетенций, обучающихся 7 –9 классов для решения теоретических и практико-ориентированных задач в области информационных технологий;
- формирование систематических знаний о современных методах прикладной информатики и программирования, их месте и роли в системе наук;
- формирование умений и навыков программирования
- развитие систематического и абстрактного мышления, вычислительной, алгоритмической культур и общей математической и информационной культуры.

Курс внеурочной деятельности отражает и расширяет содержание тематического раздела информатики алгоритмы и программирование информатики на уровне основного общего образования.

2. СОДЕРЖАНИЕ КУРСА ВНЕУРОЧНОЙ ДЕЯТЕЛЬНОСТИ «ПРОГРАММИРОВАНИЕ»

7 класс

Python

Простейшие программы. Консольный ввод и вывод данных. Конкатенация строк. Форматированный вывод. Переменные. Оператор. Операции с файлами и каталогами. Файловый ввод и вывод данных.

Компьютерная графика. Модуль Graph и графические примитивы. Система координат.

Модуль Tkinter. Система координат. Графические примитивы: линии, прямоугольники, окружности. Параметры графических примитивов. Анимация.

Модуль Turtle. Система координат. Построение графических примитивов. Управление пикселями. Изменение координат.

Пользовательские процедуры. Функции и процедуры. Подпрограмма. Рефакторинг. Параметры процедуры. Глобальные и локальные переменные.

Целые числа. Целые числа и арифметические операции. Деление нацело. Остаток от деления.

Вещественные числа. Особенности представления вещественных чисел в памяти компьютера. Операции над вещественными числами. Библиотека `math`: основные функции.

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. Циклы с условием. Алгоритм Евклида. Обработка потока данных. Бесконечные циклы.

Циклы по переменной. Циклы по переменной. Шаг изменения переменной цикла.

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

8 класс

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.

C++

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

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

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.

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

9 класс

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.

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

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ КУРСА ВНЕУРОЧНОЙ ДЕЯТЕЛЬНОСТИ «ПРОГРАММИРОВАНИЕ»

3.1. Личностные результаты:

- готовность и способность к самостоятельной, творческой и ответственной деятельности;
- навыки сотрудничества со сверстниками, взрослым в образовательной, общественно полезной, учебно-исследовательской, проектной и других видах деятельности;
- готовность и способность к образованию, в том числе самообразованию, на протяжении всей жизни;
- сознательное отношение к непрерывному образованию как условию успешной профессиональной и общественной деятельности;
- эстетическое отношение к миру, включая эстетику научного и технического творчества;
- осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов; отношение к профессиональной деятельности как возможности участия в решении личных, общественных, государственных, общенациональных проблем.

3.2. Метапредметные результаты:

- оставлять и отлаживать простые диалоговые программы;
- использовать основные алгоритмические конструкции: условные операторы, циклы с условием, циклы по переменной;
- использовать поставленные вопросы в качестве инструмента познания, а также в качестве основы для формирования гипотезы по проверке правильности высказываемых суждений;

- использовать вспомогательные алгоритмы (процедуры и функции) для структуризации программ;
- использовать символьные строки;
- использовать двоичный поиск;
- обрабатывать данные, записанные в текстовые и двоичные файлы, и сохранять в файлах результаты работы программы;
- использовать структуры для объединения данных;
- применять словари, стеки, очереди, деки для решения задач обработки данных;
- использовать деревья для организации данных;
- использовать динамическое программирование для решения комбинаторных и оптимизационных задач;
- создавать программы с графическим интерфейсом на языках Python и C++;
- использовать готовые и создавать новые компоненты (виджеты) для сред быстрой разработки программ.

3.3. Предметные результаты:

- научиться составлять и отлаживать простые диалоговые программы;
- узнает особенностей машинных вычислений с целыми и вещественными числами;
- научиться использовать основные алгоритмические конструкции: условные операторы, циклы с условием, циклы по переменной;
- овладеет методами построения графических изображений программными средствами;
- овладеет простыми методами программирования компьютерной анимации.
- познакомится с методами описания графов и некоторыми популярными алгоритмами на графах;
- научиться использовать динамическое программирование для решения комбинаторных и оптимизационных задач;
- познакомится с понятием выигрышных и проигрышных позиций в играх с полной информацией;
- познакомится с объектно-ориентированным подходом к разработке программ;
- познакомится с понятиями «класс» и «абстрактный класс»;
- познакомится с понятиями «инкапсуляция», «наследование», «полиморфизм»;
- познакомится с принципами разработки событийно-ориентированных программ;

4. ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ С УКАЗАНИЕМ ЭОР

7 класс

№	Тема	Количество часов	Модуль рабочей программы воспитания
1.	Простейшие программы	4	
2.	Компьютерная графика	4	
3.	Пользовательские процедуры	4	
4.	Целые числа. Арифметические операции над целыми числами	2	
5.	Вещественные числа. Операции над вещественными числами	2	
6.	Случайные и псевдослучайные числа	2	
7.	Ветвления	4	
8.	Цикл с условием	4	
9.	Цикл по переменной	4	
10.	Циклы в компьютерной графике	4	
11.	Проектная деятельность	30	
12.	Повторение и обобщение	4	

8 класс

№	Тема	Количество часов	Модуль рабочей программы воспитания
Python			
1.	Повторение. Документирование программы	4	
2.	Функции. Логические функции	2	
3.	Рекурсии. Рекурсивные процедуры и функции. Фракталы	2	
4.	Символьные строки. Сравнение строк. Операции со строками	2	
5.	Символы, операции с символами	2	
6.	Преобразование строка-число. Рекурсивный перебор	2	
7.	Массивы	4	
8.	Алгоритмы обработки массивов	4	

9.	Сортировка массивов	4	
C++			
10.	Структура программы на языке C++. Компиляция программы. Препроцессор. Директива include.	2	
11.	Вывод текста на экран. Диалоговые программы. Ввод и вывод данных	2	
12.	Процедуры. Процедуры с параметрами. Рефакторинг	2	
13.	Целые и вещественные числа. Арифметические выражения. Деление и остаток	2	
14.	Ветвление. Условный оператор. Вложенный условный оператор. Логические переменные. Сложные условия	2	
15.	Цикл с условием. Цикл с постусловием. Цикл с переменной. Вложенный цикл	2	
16.	Локальные и глобальные переменные. Процедуры, изменяющие аргументы	2	
17.	Функции в C++. Логические и рекурсивные функции	2	
18.	Символьные строки	4	
19.	Массивы в C++	4	
20.	Матрицы в C++	4	
21.	Сортировка массивов в C++	2	
22.	Повторение и обобщение	4	

9 класс

№	Тема	Количество часов	Модуль рабочей программы воспитания
Python			
1.	Двоичный поиск по массиву	1	
2.	Двоичный поиск по ответу	1	
3.	Обработка файлов	4	
4.	Целочисленные алгоритмы	2	

5.	Словари	2	
6.	Структуры и классы	6	
7.	Деревья	4	
8.	Графы	4	
9.	Динамическое программирование	2	
10.	Событийно-ориентированное программирование. Программы с графическим интерфейсом	8	

5. ФОРМА ОРГАНИЗАЦИИ УЧЕБНОЙ ДЕЯТЕЛЬНОСТИ

В рамках курса внеурочной деятельности «Программирование» приоритет следует отдавать тем формам работы, в которых ребенок занимает активную позицию: обсуждения, дискуссии, мозговые штурмы, решения кейсов, олимпиады, эксперименты, конкурсы, коммуникативные, деловые, интеллектуальные игры и т.п.

Формы внеурочной деятельности должны сочетать индивидуальную и групповую работу обучающихся, а также предоставлять им возможность проявить и развить свою самостоятельность.

Выбор конкретных форм реализации внеурочной деятельности образовательная организация определяет самостоятельно.