

РАБОЧАЯ ПРОГРАММА

курса внеурочной деятельности «Технология проектирования программ на языке Python»

АННОТАЦИЯ

Рабочая программа курса внеурочной деятельности составлена на основании следующих нормативных документов:

- Федеральный закон «Об образовании в Российской Федерации» от 29.12.2012 года № 273-ФЗ (с изменениями и дополнениями);
- Федеральный государственный образовательный стандарт среднего общего образования (утвержден приказом Министерства образования и науки РФ № 413 от 17 мая 2012 года);
- Приказ Министерства просвещения Российской Федерации от 12.08.2022 № 732 “О внесении изменений в федеральный государственный образовательный стандарт среднего общего образования, утвержденный приказом Министерства образования и науки Российской Федерации от 17 мая 2012 г. № 413” (Зарегистрирован 12.09.2022 № 70034)
- Основная образовательная программа среднего общего образования МБОУ СШ № 35;
- Примерная программы курса «Программирование на языках Python и C++» Поляков К.Ю. М.: БИНОМ. Лаборатория знаний, 2019

Срок реализации программы – один учебный год
На изучение курса отводится по 2 часа в неделю.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Цели изучения курса внеурочной деятельности:

- знакомление учащихся с принципами и методами современного объектно – ориентированного программирования высокого уровня;
- формирование знаний, умений и навыков в области программирования на языке Python
- развитие алгоритмических способностей учащихся по программированию.

Отличительными особенностями данной программы являются:

- Данный курс имеет выраженную практическую направленность, которая и определяет логику построения материала учебных занятий. Основной формой обучения является практикум.
- Для поддержки курса К.Ю. Поляковым разработано электронное учебное пособие в формате PDF, которое содержит теоретический материал, задания для выполнения практических работ, презентации, которое используется во время уроков для самостоятельной работы и в качестве справочника. Это позволяет успешно организовывать занятия в группах, в которых есть ученики с разным темпом усвоения материала.
- Знания, полученные при изучении элективного курса учащиеся могут применить для решения прикладных задач разного рода, повышения качества подготовки к успешной сдаче ЕГЭ.

- Полученные знания и умения являются основой для последующего изучения программирования в высших профессиональных образовательных учреждениях.
- образовательный процесс может осуществляться удаленно, через сеть Интернет в режиме реального времени посредством общения в программе Zoom, Discord. Это позволяет вести данный курс в дистанционном формате, когда очная форма обучения по каким-либо причинам недоступна.
- составлена с учетом рабочей программы воспитания МБОУ СШ №35.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ВНЕУРОЧНОЙ ДЕЯТЕЛЬНОСТИ.

Личностные результаты:

1. Сформированность мировоззрения, соответствующего современному уровню развития науки и общественной практики. Данная учебная дисциплина формирует представления учащихся о науках, развивающих информационную картину мира, вводит их в область информационной деятельности людей. Ученики узнают о месте, которое занимает программирование в современном информационном пространстве, современной системе наук, его связи с другими научными областями. Ученики получают представление о современном уровне и перспективах развития этой отрасли, в реализации которых в будущем они, возможно, смогут принять участие.

2. Сформированность навыков сотрудничества со сверстниками, взрослыми в образовательной, общественно полезной, учебно-исследовательской, проектной и других видах деятельности. Эффективным методом формирования данных качеств является учебно-проектная деятельность. Работа над проектом требует взаимодействия между учениками – исполнителями проекта, а также между учениками и учителем – формулирующим задание для проектирования, контролирующим ход его выполнения, принимающим результаты работы. В завершение работы предусматривается процедура защиты проекта перед коллективом класса, которая также требует наличия коммуникативных навыков у детей.

3. Бережное, ответственное и компетентное отношение к физическому и психологическому здоровью как собственному, так и других людей, умение оказывать первую помощь. Всё большее время у современных детей занимает работа за компьютером (не только над учебными заданиями). Поэтому для сохранения здоровья очень важно знакомить учеников с правилами безопасной работы за компьютером, с компьютерной эргономикой.

4. 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; Данное качество формируется в процессе развития навыков самостоятельной учебной и учебно-исследовательской работы учеников. Выполнение проектных заданий требует от ученика проявления самостоятельности в изучении нового материала, в поиске информации в различных источниках. Такая деятельность раскрывает перед учениками возможные перспективы в изучении предмета, в дальнейшей профориентации в этом направлении. В процессе изучения курса ученики узнают о использовании программирования в различных профессиональных областях и перспективы их развития.

5. Осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов; Важное место в изучении данного курса занимает знакомство учащихся с современными профессиями в IT-отрасли.

Метапредметные результаты:

1. Умение самостоятельно определять цели и составлять планы; самостоятельно осуществлять, контролировать и корректировать учебную и внеучебную (включая внешкольную) деятельность; использовать все возможные ресурсы для достижения целей;

выбирать успешные стратегии в различных ситуациях. Данная компетенция формируется при изучении курса в нескольких аспектах, таких как:

- учебно-проектная деятельность: планирование целей и процесса выполнения проекта и самоконтроль за результатами работы;
- изучение основ системологии: способствует формированию системного подхода к анализу объекта деятельности;
- алгоритмическая линия курса: алгоритм - это план достижения цели исходя из ограниченных ресурсов (исходных данных) и ограниченных возможностей исполнителя (системы команд исполнителя).

2. Умение продуктивно общаться и взаимодействовать в процессе совместной деятельности, учитывать позиции другого, эффективно разрешать конфликты. Формированию данной компетенции способствуют следующие аспекты методической системы курса:

- формулировка многих вопросов и заданий к теоретическим разделам курса стимулирует к дискуссионной форме обсуждения и принятия согласованных решений;
- ряд проектных заданий предусматривает коллективное выполнение, требующее от учеников умения взаимодействовать; защита работы предполагает коллективное обсуждение ее результатов.

3. Владение навыками познавательной, учебно-исследовательской и проектной деятельности, навыками разрешения проблем; способность и готовность к самостоятельному поиску методов решения практических задач, применению различных методов познания.

Большое место в методике углубленного изучения информатики занимает учебно-исследовательская и проектная деятельность. Предусматриваются проекты как для индивидуального, так и для коллективного исполнения. В частности, в рамках коллективного проекта ученик может быть как исполнителем, так и руководителем проекта. В методике учебно-проектной работы предусматриваются коллективные обсуждения с целью поиска методов выполнения проекта.

4. Готовность и способность к самостоятельной информационно познавательной деятельности, включая умение ориентироваться в различных источниках информации, критически оценивать и интерпретировать информацию, получаемую из различных источников. Данное направление стремительно развивается, поэтому успешная учебная и производственная деятельность в этой области невозможна без способностей к самообучению, к активной познавательной деятельности, без умения получения, отбора и систематизации информации. Интернет является важнейшим современным источником информации, ресурсы которого постоянно расширяются

5. Владение навыками познавательной рефлексии как осознания совершаемых действий и мыслительных процессов, их результатов и оснований, границ своего знания и незнания, новых познавательных задач и средств их достижения. Формированию этой компетенции способствует методика индивидуального подхода, практические задания разделены на три уровня сложности: репродуктивный, продуктивный и творческий, такое разделение станет для некоторых учеников стимулирующим фактором к переоценке и повышению уровня своих знаний и умений.

Предметные результаты:

- научится составлять и отлаживать простые диалоговые программы;
- узнает особенностей машинных вычислений с целыми и вещественными числами;
- научится 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 проектирования программ «сверху вниз» и «снизу вверх»;
- научится использовать вспомогательные алгоритмы (процедуры и функции) для структуризации программ;

- научиться применять рефакторинг для улучшения читаемости программ;
- научиться использовать символьные строки;
- овладеет основными алгоритмами обработки одномерных и двумерных массивов;
- познакомится с понятием сложности алгоритма;
- научиться применять различные алгоритмы сортировки массивов;
- научиться использовать двоичный поиск;
- научиться обрабатывать данные, записанные в текстовые и двоичные файлы, и сохранять в файлах результаты работы программы;
- научиться использовать структуры для объединения данных;
- научиться применять словари, стеки, очереди, деки для решения задач обработки данных;
- научиться использовать деревья для организации данных;
- 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;

СОДЕРЖАНИЕ ВНЕУРОЧНОЙ ДЕЯТЕЛЬНОСТИ С УКАЗАНИЕМ ФОРМ ЕЕ ОРГАНИЗАЦИИ И ВИДОВ ДЕЯТЕЛЬНОСТИ.

Основными формами организации занятий являются традиционные занятия с фронтальной формой работы, практические занятия по решению задач, групповая работа над проектами.

Этапы решения задач. Знакомство с языком и средой программирования. (19 часов)

Что такое проект. Виды проектов Структура проекта. Постановка проблемы. Требования к проекту Этапы работы над проектом: подготовка, планирование, моделирование, реализация, оформление, рецензирование, защита

. Сущность моделирования. Понятие модель. Виды модели. Информационное моделирование. Моделирование и формализация

Математическая модель задачи. Построение математических моделей задач

Понятия алгоритма. Свойства алгоритма.. Способы описания алгоритмов

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.

Этапы проектирования и разработки программ. Обработка массивов (17 часов)

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.

Программирование на языке Python (17 часов)

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.

Программирование на языке Python. Повышенный уровень (17 часов)

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.

ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ.

Номер урока	Тема занятия	Кол-во часов	
		теория	практика
	Этапы решения задач. Знакомство с языком и средой программирования (19 часов)		
1.	Проект. Структура проекта. Планирование проекта. Как защитить проект.	1	
2.	Этапы решения задач по программированию. Моделирование и формализация	1	
3.	Первые программы	0,5	0,5
4.	Диалоговые программы	0,5	0,5
5.	Компьютерная графика	0,5	0,5
6.	Процедуры	0,5	0,5
7.	Обработка целых чисел	0,5	0,5
8.	Обработка вещественных чисел	0,5	0,5
9.	Случайные и псевдослучайные числа	0,5	0,5
10.	Ветвления	0,5	0,5
11.	Сложные условия	0,5	0,5
12.	Циклы с условием	0,5	0,5
13.	Циклы с условием: практикум		1
14.	Анимация	0,5	0,5
15.	Циклы по переменной	0,5	0,5
16.	Циклы в компьютерной графике	0,5	0,5
17.	Выполнение проекта		1
18.	Выполнение проекта		1
19.	Выполнение проекта		1
	Этапы проектирования и разработки программ. Обработка массивов. (17 часов)		
20.	Проектирование программ	1	
21.	Процедуры	0,5	0,5
22.	Рекурсия	0,5	0,5
23.	Функции	0,5	0,5
24.	Символьные строки	0,5	0,5
25.	Обработка символьных строк	0,5	0,5
26.	Строки в функциях	0,5	0,5
27.	Массивы	0,5	0,5
28.	Ввод и вывод массивов	0,5	0,5
29.	Суммирование элементов массива	0,5	0,5
30.	Подсчёт элементов массива, удовлетворяющих условию	0,5	0,5

Номер урока	Тема занятия	Кол-во часов	
		теория	практика
31.	Поиск значения в массиве	0,5	0,5
32.	Поиск максимального элемента в массиве	0,5	0,5
33.	Игра «Стрельба по тарелкам»	0,5	0,5
34.	Игра «Стрельба по тарелкам»	0,5	0,5
35.	Матрицы	0,5	0,5
36.	Сложность алгоритмов	1	
	Программирование на языке Python (17 часов)		
37.	Простые алгоритмы сортировки	0,5	0,5
38.	Сортировка слиянием	0,5	0,5
39.	Быстрая сортировка	0,5	0,5
40.	Двоичный поиск	0,5	0,5
41.	Обработка файлов	0,5	0,5
42.	Обработка файлов: практикум		1
43.	Целочисленные алгоритмы	0,5	0,5
44.	Словари	0,5	0,5
45.	Структуры	0,5	0,5
46.	Структуры: практикум		1
47.	Стек, очередь, дек	0,5	0,5
48.	Деревья	0,5	0,5
49.	Графы	0,5	0,5
50.	Графы: практикум		1
51.	Динамическое программирование	0,5	0,5
52.	Динамическое программирование: практикум	0,5	0,5
53.	Игровые модели	0,5	0,5
54.	Игровые модели: практикум		1
	Программирование на языке Python. Повышенный уровень (17 часов)		
55.	Что такое ООП?	1	
56.	Модель задачи: классы и объекты	0,5	0,5
57.	Классы и объекты в программе	0,5	0,5
58.	Классы и объекты в программе: практикум		1
59.	Скрытие внутреннего устройства	0,5	0,5
60.	Иерархия классов	0,5	0,5
61.	Классы-наследники (I)	0,5	0,5
62.	Классы-наследники (II)	0,5	0,5
63.	Доработка игры		1
64.	Событийно-ориентированное программирование	0,5	0,5
65.	Использование компонентов (виджетов)	0,5	0,5
66.	Использование компонентов (виджетов)	0,5	0,5
67.	Создание компонентов	0,5	0,5
68.	Модель и представление	0,5	0,5
69.	Выполнение проекта		1
70.	Выполнение проекта		1