Комитет по образованию и делам молодежи Администрации
Алтайского района Алтайского края
Муниципальное бюджетное общеобразовательное учреждение
«Алтайская средняя общеобразовательная школа № 2»
имени Почетного гражданина Алтайского края И.А. Яркина
(МБОУ «Алтайская СОШ №2»)

ПРИНЯТА
Методическим подразделением педагогов
дополнительного образования
Протокол от 26.08.2023 № 1

УТВЕРЖДЕНА
Приказом от 30.08 № 140/ОД
Директор МБОУ «Алтайская СОШ №2»
_____Т.В.Бауэр

СОГЛАСОВАНА
И.о. заместителя директора по ВР
МБОУ «Алтайская СОШ №2»
_____Н. А. Панченко

Дополнительная общеобразовательная общеразвивающая
программа
«Учимся программировать на языке Python»
технической направленности
Возраст учащихся 13-17 лет
Срок реализации: 1 год

составитель: Тимашёва Н.Н.
педагог дополнительного образования

с. Алтайское
2023

Пояснительная записка

Данная дополнительная общеобразовательная общеразвивающая программа соотносится с тенденциями развития дополнительного образования и согласно Концепции развития дополнительного образования способствует формированию и развитию имеющихся навыков операционного и логического стиля мышления, исследовательских и творческих качеств.

Актуальность

Актуальность программы отвечает современным потребностям детей и родителей в получении необходимых теоретических знаний и навыков в сфере программирования для личностного развития и успешной адаптации к требованиям жизни в современном информационном обществе.

Изучение программирования развивает логическое мышление, креативность, исследовательские и творческие качества. Писать программы — занятие увлекательное и временами непростое, однако полученный опыт пригодится и в школе, и дома, да и на рынке труда профессия программиста востребована.

Новизна Для реализации программы выбран язык Python – это один из самых популярных, быстро развивающихся языков в современном мире, при этом он имеет достаточно простой синтаксис и подходит для начального обучения.

Читать код на этом языке программирования легко, т.к. в нем используется минимум вспомогательных элементов, а правила языка заставляют программистов делать отступы. В Python есть инструменты, заметно ускоряющие обучение и позволяющие создавать несложные анимации для своих игр, поэтому большинство тем изучается на примерах написания игр. С другой стороны, Python – это универсальный язык программирования. Это язык так называемого сверх высокого уровня – он поддерживает несколько парадигм программирования: структурное, объектно-ориентированное, функциональное и аспектно-ориентированное. Язык Python с легкостью можно применять в любых целях – как для написания небольших скриптов, так и для создания полноценных приложений, от веб-программирования до разработки игр.

Для языка Python создано множество доступных сред разработки, сервисов и фреймворков. В стандартных библиотеках Python есть средства для работы с электронной почтой, протоколами Интернета, FTP, HTTP, базами данных и другие инструменты.

Хорошая модульность языка дает возможность составлять программу из готовых элементов, выполняющих различные действия. В некоторых случаях для написания программы достаточно лишь найти подходящие модули и правильно их скомбинировать.

Цель и задачи

Основная цель курса «Учимся программировать на языке Python» – формирование и развитие имеющихся навыков операционного и логического стиля мышления, исследовательских и творческих качеств.

Задачи

Обучающие:

- обучить синтаксису языка и основным базовым алгоритмическим конструкциям;

- обучить навыкам разработки, тестирования и отладки несложных программ;
- познакомить с понятием проекта, его структуры, дизайна и разработки;
- формировать мотивы к познавательной и творческой деятельности.

Воспитательные:

- создание условий для формирования навыков работы в группе;
- создание условий для формирования культуры общения между учащимися;
- создание условий для самоопределения учащихся в профессиональном выборе.

Развивающие:

- развитие логического мышления;
- развитие творческих способностей;
- развитие интереса учащихся к выбранному ими профилю деятельности;
- развития трудолюбия и целеустремлённости.
-

Уровень сложности – программа «Учимся программировать на языке Python» является разноуровневой. Предполагается последовательное освоение уровней, при этом овладение стартовым (базовым) уровнем является достаточным.

Направленность программы

Дополнительная общеобразовательная программа «Учимся программировать на языке Python» имеет техническую направленность.

Программа направлена на развитие и поддержку детей, проявивших интерес и определенные способности к программированию.

Категория учащихся

Возраст детей, участвующих в реализации программы с 13 до 17 лет (учащиеся 7-11 классов). Дети данного возраста способны на общекультурном уровне выполнять предлагаемые задания по образцу и создавать свои простые программы.

Сроки реализации программы

Общее количество учебных часов, запланированных на весь период обучения и необходимых для освоения программы – 70 часов.

Срок освоения программы - 1 год.

Формы и режим занятий

Занятия проводятся в очной форме на русском языке. Группа формируется из 10-ти человек — по количеству рабочих мест (компьютеров).

процессе обучения используются различные формы занятий: традиционные, комбинированные и практические занятия, конкурсы, но основной тип занятий — практикум. Большинство заданий курса выполняется с помощью персонального компьютера и необходимых программных средств.

Преподавание нового теоретического материала рекомендуется проводить в форме беседы, используя проблемный подход, закрепляя его иллюстрацией на различных примерах и написанием игр.

Планируемые результаты.

Программа обеспечивает формирование личностных, предметных, метопредметных результатов.

Личностными результатами:

формирование ответственного отношения к учению, готовности и способности обучающихся к саморазвитию и самообразованию на основе мотивации к обучению и познанию, осознанному выбору и построению дальнейшей индивидуальной траектории образования на базе ориентировки в мире профессий и профессиональных предпочтений, с учётом устойчивых познавательных интересов;

формирование целостного мировоззрения, соответствующего современному уровню развития науки и общественной практики, учитывающего социальное, культурное, языковое, духовное многообразие современного мира;

формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками, детьми старшего и младшего возраста, взрослыми в процессе образовательной, общественно полезной, учебно-исследовательской, творческой и других видов деятельности.

Метапредметные результаты:

умение самостоятельно определять цели своего обучения, ставить и формулировать для себя новые задачи в учёбе и познавательной деятельности, развивать мотивы и интересы своей познавательной деятельности;

умение самостоятельно планировать пути достижения целей, в том числе альтернативные, осознанно выбирать наиболее эффективные способы решения учебных и познавательных задач;

умение соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности в процессе достижения результата, определять способы действий в рамках предложенных условий и требований, корректировать свои действия в соответствии с изменяющейся ситуацией;

• умение оценивать правильность выполнения учебной задачи, собственные возможности её решения;

• владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности;

• умение определять понятия, создавать обобщения, устанавливать аналогии, классифицировать, самостоятельно выбирать основания и критерии для классификации, устанавливать причинно-следственные связи, строить логическое рассуждение, умозаключение (индуктивное, дедуктивное и по аналогии) и делать выводы;

• умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебных и познавательных задач;

• умение организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками; работать индивидуально и в группе: находить общее решение и разрешать конфликты на основе согласования позиций и учёта интересов; формулировать, аргументировать и отстаивать своё мнение;

• формирование и развитие компетентности в области использования информационно-коммуникационных технологий.

Предметные результаты:

формирование представления об основных изучаемых понятиях курса;

формирование информационной и алгоритмической культуры; формирование представления о компьютере как универсальном устройстве обработки информации; развитие основных навыков и умений использования компьютерных устройств;

развитие алгоритмического мышления, необходимого для профессиональной деятельности в современном обществе; развитие умений составить и записать алгоритм для решения конкретной задачи;

формирование умений формализации и структурирования информации, умения выбирать способ представления данных в соответствии с поставленной задачей, с использованием соответствующих программных средств обработки данных;

формирование представления о том, что значит “программировать” на примере графического языка Blockly;

знакомство с базовыми конструкциями языка Python; формирование умения придумывать алгоритмы и их реализовывать на языке Python;

знакомство с основными алгоритмическими структурами — линейной, условной и циклической;

формирование умений работы с дополнительными библиотеками языка Python (tkinter, pygame, etc);

формирование умения создавать реальные приложения с помощью языка Python, формирование умения применять накопленные знания для решения практических задач;

использование готовых прикладных компьютерных программ по выбранной специализации;

развитие умений применять изученные понятия, результаты, методы для решения задач практического характера и задач из смежных дисциплин с использованием при необходимости справочных материалов, компьютера;

формирование навыков и умений безопасного и целесообразного поведения при работе с компьютерными программами и в Интернете, умения соблюдать нормы информационной этики и права.

Содержание

Раздел 1. Техника безопасности. Вводное занятие (1 часа).

Урок №1. Теория: **Техника безопасности. Вводное занятие -1ч.**

Знакомство с планом работы объединения, инструктаж по ТБ. Практика: Опрос. Тренинг на командообразование.

Раздел 2. Введение в язык программирования Python (30 часа)

История создания языка. Установка Python. Структура программы. Типы данных. Ввод-вывод. Линейные алгоритмы. Алгоритмы с ветвлением. Циклы. Вложенные циклы. Циклы for и while. Списки. Функции. Модули. Работа с текстовыми файлами. Графический модуль PyTurtle. Графика с модулем tkinter. Создание приложения Painter. Разработка несложных консольных приложений.

Урок № 2. Теория: Знакомство с языком программирования Python – 1ч.

История создания.

Python - это интерпретируемый язык программирования.

Python - это полноценный язык программирования.

Python – свободно распространяемый язык программирования.

Дзэн Питона. Команда «import this»

Установка языка Python.

Интерактивный режим.

Создание первой программы: «Hello, world»

Урок № 3. Практика: Структура программы. Типы данных. Переменные – 1ч.

Ввод и вывод данных.

Операции.

Создание и редактирование программ в среде IDLE.
Типы данных: целые числа, числа с плавающей точкой, строки.
Запуск программ.
Кейс: создайте учебную задачу для соседа на ввод/вывод в Python или на последовательность выполнения арифметических операций.
Урок № 4. Практика: Линейные алгоритмы – 1ч.
Блок-схема линейного алгоритма.
Ввод данных с клавиатуры.
Правила записи арифметических выражений.
Множественное присваивание.
Случайные числа.
Порядок выполнения операций.
Вывод на экран.
Кейс. Работа в парах: запишите выражение по правилам языка Python и предложите напарнику вычислить его, не используя интерпретатора Python. Проверьте себя в среде разработки.
Урок № 5. Теория: Ветвящиеся алгоритмы – 1ч.
Инструкция ветвления if.
Ветвление. Полная и сокращенная формы
Логический тип данных.
Логическое высказывание.

Урок № 6. Практика: Ветвящиеся алгоритмы - -1ч.
Множественное ветвление.
Инструкция ветвления if … else.
Мини-кейс:
Программа «Открыто / закрыто»
Урок № 7. Теория: Циклические алгоритмы – 1ч.
Цикл с предусловием.
Блок-схема цикла с предусловием.
Инструкция while.
Мини-кейс:
Подсчет суммы цифр целого числа.
Урок № 8. Практика: Циклические алгоритмы – 1ч.
Мини-кейс:
Поиск НОД двух целых чисел.
Урок № 9. Практика: Циклические алгоритмы – 1ч.
Цикл с параметром.
Блок-схема цикла с параметром.
Генерация диапазона значений. Функция range().
Мини-кейс: вычисление факториала N.
Урок № 10. Практика: Вложенные циклы - 1ч.
Цикл в цикле.
Время работы программы, содержащей вложенные циклы.
Мини-кейс: написать программу, выводящую на экран таблицу Пифагора.
Урок 11. Практика: Цикл for в Python – 1ч.
Использование цикла для перебора объектов (конструкция for elem in object, где object – строка, кортеж, иной итерируемый объект). Использование цикла для выполнения заданного количества операций: три способа записи for i in range(n); for i in range(a, b); for i in range(a, b, d)). Простые программы, например:

• Как напечатать слово привет столько раз, какое число введет пользователь (каждый “привет” должен быть в отдельной строке)?
• Напечатайте числа от 1 до n (n определяется пользователем) в прямом и в обратном порядке, только четные, только кратные трем, и т.п. задает число n, программа выводит n строк, в первой строке одна звездочка, во второй две и т.д.

Урок 12. Практика: Цикл while – 1ч.

Цикл с предусловием. Блок-схема. Бесконечный цикл. Оператор break для выхода из цикла. Переменная-счетчик для подсчета количества операций в цикле. Простые программы, например, такие, как приведены ниже:

Программа 1. Сладкоежка

```
candies = 0
ans = ‘да’
while ans == ‘да’:
candies = candies + 1
print(‘Съедено конфет:’, candies)
ans = input(‘Хочешь еще конфетку? да/нет\n’)
```

Программа 2. Бесконечный цикл:

```
while True:
print(‘я буду работать вечно!’)
```

Программа 3. Надоедливая программа:

```
while True:
ans = input(‘я тебе не надоела? да/нет\n’)
if ans == ‘да’:
print(‘Как некрасиво! Ухожу…’)
break
```

Программа 4. Угадай, как меня зовут?

```
print(‘Угадай, как меня зовут!’)
ans = input()
while ans != ‘python’:
print(‘Не угадал! Попробуй еще раз’)
ans = input()
print(‘Правильно, я python!’)
```

Урок № 13. Теория: Списки – 1ч.

Массивы и списки.
Индекс (номер элемента списка).
Создание, ввод и вывод списков.
Просмотр элементов списка.
Добавление удаление элементов списка. Количество элемен тов списка может меняться во время выполнения программы.

Мини-кейс:

Программа «Рекорды»

Урок № 14. Практика: Списки – 1ч.

Копирование списков.
Линейный поиск.
Сортировка списка.

Мини-кейс:

Создать список учеников класса, отсортировать его и написать программу, которая спрашивает у пользователя число N и выводит фамилию и имя N-ого по алфавиту ученика, или N-ого с конца алфавита.

Урок № 15. Практика: Функции – 1ч.
Вспомогательные алгоритмы.
Объявление функции.
Правила описания функций.
Параметры и возвращаемые значения.
Области видимости.
Чтение глобальной переменной внутри функции.
Урок № 16. Практика: Модули – 1ч.
Модули из стандартной библиотеки.
Инструкция import.
Инструкция from.
Использование псевдонимов
Создание своего модуля.
Урок № 17. Практика: Работа с текстовыми файлами – 1ч.
Функция open и её аргументы.
Чтение из файла.
Запись в файл.
Урок № 18. Теория: Практическая работа – 1ч.

Кейс:
Создание игры «Викторина».
Вопросы и варианты ответов с указанием правильного хранятся в текстовом файле. Вопросы задаются последовательно. Игроку предлагается выбрать правильный ответ из предложенных вариантов. В случае правильно ответа результат увеличивается на единицу, и игроку выводится сообщение: «верно». В случае неверного ответа выводится сообщение: «не верно».
После ответа на последний вопрос на экран выводится результат (количество верных ответов).
Урок № 19. Практика: Практическая работа – 1ч.
Кейс:
Завершение работы над Викториной. Работа в парах: ребята разбиваются на пары, и тестируют игру, написанную напарником.
Урок № 20. Теория: Графический модуль PyTurtle – 1ч.
Импорт модуля turtle.
Внешность, формы «черепашки».
Размер «черепашки».
Цвет «черепашки».
Отпечаток «черепашки».
Урок № 21. Практика: Графический модуль PyTurtle - 1ч.
Команды перемещения.
Команды поворота.
Рисуем домики, машинки и т.п.
Раскрашиваем картинки.
Кейс. Нарисуй самый необычный домик.
Урок № 22. Практика: Графический модуль PyTurtle – 1ч.
Возможности рисования в PyTurtle.
Цвет и размер пера.
Решение задач с использованием линейных алгоритмов.

Урок № 23. Практика: Графический модуль PyTurtle – 1ч.
Положение «черепашки».

Координатная плоскость.
Направление «черепашки».
Команда поставить точку.
Циклические конструкции в PyTurtle
Урок № 24. Практика: Графический модуль PyTurtle – 1ч.
Создание геометрических фигур в PyTurtle.
Рисование спиралей в PyTurtle.
Конструкции ветвления алгоритмов в PyTurtle.
Урок № 25. Практика: Практическая работа – 1ч.
Кейс:
Рисуем снежинки. Конкурс на самую необычную снежинку.
Урок № 26. Теория: Графика с модулем tkinter в Python – 1ч.
Создание графического интерфейса.
Импорт модуля tkinter.
Создание базового окна.
Создание рамки.
Создание метки.
Создание кнопок.
Урок № 27. Практика: Графика с модулем tkinter в Python -1ч.
Менеджер размещения Grid.
Текстовые поля и области.
Применение флажков.
Применение переключателей.
Урок № 28. Практика: Виджет Canvas – 1ч.
Холст.
Линии.
Цвета.
Формы.
Рисуем изображение.

Рисуем текст.
Урок № 29. Практика: Практическая работа – 1ч.
Кейс: Командная работа.
«Создаем приложение «Painter», позволяющее рисовать мышкой на экране (с палитрой цветов и ластиком)»
Обучающийся научится:
Составлять математическую модель, алгоритм и программу для решения простых задач;
Записывать математические выражения на языке Python;
Анализировать готовую программу и предсказывать результат;
Выделять вспомогательные алгоритмы в сложных задачах;
Обрабатывать списки;
работать с графическим модулем PyTurtle;
работать с графическим модулем tkinter;
Урок 30. Практика: Кейс: Создание программы-теста из нескольких вопросов – 1ч.
Создание тематического теста. Программа ведет диалог с пользователем, задавая ему 1-3 вопроса, каждый по одной и той же схеме (задается вопрос, далее дается три попытки на ответ, если одна из попыток заканчивается верным ответом, выводится строка “Ответ верный!”, если ни в одной попытке не было дано верного ответа, просто показывается верный ответ.) Программа должна

засчитывать верный ответ без учета регистра, для реализации этого удобно использовать встроенную
функцию lower или встроенную функцию upper.
Пример подобной программы, задающей пользователю вопрос про футбол, приведен ниже.

```
print(‘Тест про футбол’)
ans = input(‘Вопрос 1. В какой стране проходил последний чемпионат мира по футболу\n’)
cheker = True
attempt = 0
while cheker and attempt < 3:
if ans.lower() == «россия» or ans.lower() == «в россии»:
print(‘Ответ верный!’)
cheker = False
else:
if attempt < 2:
ans = input(‘Попробуй еще раз!\n’)
attempt = attempt + 1
if attempt == 3:
print(‘Попытки закончились... Верный ответ: Россия или В России’)
```

Урок 31. Практика: Завершение работы над программой-тестом – 1ч.
Добавление в тест еще одного-двух вопросов. Добавление счетчика очков (за каждый верный ответ пользователю начисляется определенное количество очков, после прохождения теста это количество очков выводится на экран). Обсуждение со школьниками того, как использование функций могло бы помочь сделать код короче (все вопросы теста однотипны), реализация при достаточном уровне подготовленности школьников.

Раздел 3. Создание приложений с помощью tkinter (15 часов)
Работа с модулем tkinter. Виджеты. Конфигурация виджетов. Реакция на события. Упаковщики виджетов. Рисование на холсте canvas. Управление нарисованными объектами с помощью клавиатуры. Разработка и создание GUI-приложений “пинг- понг”, “сапер”.

Урок 32. Теория: Знакомство с модулем tkinter – 1ч.
Подключение модуля tkinter. Краткий обзор возможностей, демонстрация уже готовых приложений, разработанных с помощью tkinter. Создание простого GUI-приложения, состоящего из одного виджета, например, кнопки. Обсуждение общей последовательности действий, необходимой для создания GUI-приложения. Совершенствуем наше приложение, добавив реакцию на событие, например, щелчка левой кнопки мыши (например, виджет меняет цвет).
Урок 33. Теория: Что такое виджеты, конфигурация виджетов – 1ч.
Некоторые виджеты: кнопка, текстовое поле, поле ввода, метка. Конфигурация виджетов, методы config и configure. Получение информации о состоянии виджета.

Урок 34. Теория: События в tkinter – 1ч.
Как научить виджет реагировать на события – различные способы. Методы bind и unbind.
Кейс: Создайте интеллект-карту на тему “События в tkinter”.
Урок 35. Теория: Создание простых приложений – 1ч.
Продолжаем знакомиться с tkinter на простых примерах.
Создание приложения, состоящего из кнопки, подсчитывающей количество нажатий, и исчезающей после определенного количества нажатий. Создание простого приложения, состоящего из двух

кнопок “Привет!” и “Пока…” и текстового поля: при нажатии на одну из кнопок в текстовом поле печатается приветствие, при нажатии на вторую кнопку, приложение закрывается.

Урок 36. Практика: Создание простых приложений – 1ч.

Приложение “Радуга”: простое приложение, состоящее из кнопок, окрашенных в основные цвета радуги и текстового поля. При нажатии на кнопку, в текстовом поле появляется название цвета нажатой кнопки.

Урок 37. Практика: Создание простых приложений – 1ч.

Упаковщики виджетов: pack(), grid(), place(). Создание простого калькулятора из нескольких кнопок и текстового поля для вывода результата вычислений.

Урок 38. Практика: Продвинутое рисование в tkinter – 1ч.

Рисование на холсте canvas в tkinter. Создание различных геометрических фигур.

Урок 39. Практика: Движение нарисованных объектов – 1ч.

Движение нарисованных объектов с помощью клавиатуры.

Урок 40. Теория: Кейс: Создание игрового приложения “Пинг-понг” – 1ч.

Создание простой версии игры “пинг-понг”. Создаем мячик и ракетку с помощью canvas. “Учим” ракетку двигаться влево-вправо при нажатии кнопок со стрелками влево-вправо соответственно. Как сделать так, чтобы ракетка двигалась все время и при нажатии кнопки только меняла направление движения. Как сделать так, чтобы ракетка не выходила за левую и правую границы игрового поля.

Урок 41. Практика: Создание игрового приложения “Пинг-понг” – 1ч.

Доделываем приложение пинг-понг. Добавляем правильное движение мячика – с отскоком от стен и от ракетки. Добавляем счетчик количества ударов мяча о ракетку.

Урок 42. Практика: Завершение работы над приложением “пинг-понг” – 1ч.

Работа в парах: ребята делятся на пары программист-тестер, и тестируют созданные приложения.

Общая идея игры сапер. Двумерные списки, вложенные циклы. Как расположить виджеты Label в виде прямоугольника. Раскрасим минное поле в шахматном порядке.

Урок 43. Практика: Приложение “Сапер”. Информация о минах – 1ч.

Двумерный битовый список для хранения информации о минах в игре. Создаем и заполняем его случайным образом. Как с помощью модуля random можно контролировать количество мин в игре? (Например, выбираем случайное целое число от 1 до 100, если оно оказалось меньшим, чем 25, в соответствующую ячейку пишем 1 – это будет мина, иначе 0 – пустая ячейка. Обсуждение со школьниками, много или мало мин в этом случае стоит ожидать).

Урок 44. Практика: Приложение “Сапер”. Как узнать, сколько мин среди соседей данной клетки и как эту информацию хранить? – 1ч.

По созданному двумерному списку мин создаем список, в каждой ячейке которого находится информация о том, сколько мин среди соседей данной клетки (соседями считаются клетки, имеющие с данной клеткой общую вершину или сторону).

Урок 45. Практика: Создаем “мозг” игры – 1ч.

Пишем “мозг” игры сапер. Что происходит в тот момент, когда пользователь нажимает на клетку, описываем реакцию виджета на событие.

Урок 46. Практика: Завершение работы над приложением “Сапер” – 1ч.

Доделываем приложение, тестируем, убираем недостатки.

Кейс: Как представить свой проект на конференции? Ребята разбиваются на группы, выбирают лучшее из созданных ими приложений, рассказывают о процессе создания.

Раздел 4. Продвинутые библиотеки языка Python. Pygame (13 часов)

Обзор дополнительных библиотек для работы с графическим интерфейсом. Библиотека Pygame. Шаблон программы. Геометрические примитивы в Pygame. Простая анимация в Pygame.

Урок 47. Теория: Знакомство с продвинутыми фреймворками для разработки GUI-приложений – 1ч.

Обзор дополнительных библиотек, позволяющих создавать приложения с графическим интерфейсом (PyQt, wxPython, Pygame). Примеры приложений. Установка библиотеки Pygame.

Урок 48. Теория: Шаблон программы на Pygame – 1ч.

Как устроен шаблон программы на Pygame. Подключение модулей, инициализация, главный цикл, корректное завершение работы приложения. Функции init(), set_mode(), update(), quit(). События.

Урок 49. Практика: Геометрические примитивы в Pygame – 1ч.

Цветовые модели. Рисуем и раскрашиваем линии, прямоугольники, многоугольники, круги, овалы.

Урок 50. Практика: Кейс: Простая анимация в Pygame – 1ч.

Создание анимации движения геометрической фигуры от левой границы главного окна вправо. Как изменить программу так, чтобы фигура не исчезала, а отражалась от стен.

Урок 51. Практика: События клавиатуры – 1ч.

Модуль pygame.event.

Кейс: создание приложения, реализующего перемещение геометрической фигуры с помощью стрелок вправо/влево и вверх/вниз.

Урок 52. Практика: События мыши – 1ч.

Обработка событий нажатия кнопки мыши, отпускания кнопки мыши, движения мыши. Координаты мыши.

Кейс: “Звездное небо”. Создание приложения, в котором при щелчке мыши по экрану на экране появляются звездочка.

Урок 53. Практика: Дополнительные поверхности в Pygame – 1ч.

Surface и blit(). Простые примеры.

Урок 54. Практика: Кейс: Работа в команде. Создание приложения “Поймай звезду” – 1ч.

Создание игры, в которой в произвольном месте экрана появляются звездочки, падающие вниз. При щелчке мыши по звездочке она исчезает, а в соответствующем текстовом поле идет подсчет пойманных звезд. Классы и объекты. Реализация появления звезд. Ребята разбиваются на группы, распределяют задачи внутри группы.

Урок 55. Практика: Продолжение работы над игрой. Текст в Pygame – 1ч.

Реализация реакции звезды на событие щелчка мыши. Работа с текстом в Pygame. Модуль pygame.font.

Урок 56. Практика: Завершение работы над проектом. Музыка в Pygame – 1ч.

Добавление фоновой музыки и звуковых эффектов в игру. Модули pygame.mixer и pygame.mixer.music.

Представление проекта.

Урок 57. Теория: Кейс: Разработка и создание игры “Мемори” – 1ч.

Подготовка игрового поля. Использование символов Unicode в качестве картинок, например, u’\u2702’ это символ с изображением ножниц. Создание сетки из кнопок с закрытым текстом с помощью упаковщика grid. Конфигурация цвета поля и размера текста. Создание функции, открывающей текст-изображение при нажатии на кнопку. Обсуждение дальнейшей реализации.

Урок 58. Практика: Разработка и создание игры “Мемори” – 1ч.

Использование словарей для хранения информации о том, какая кнопка нажата. Использование grid_info()[‘row’] и grid_info() [‘column’] для получения информации о том, в какой строке и в каком столбце находится нажатая кнопка. Модификация функции, обрабатывающей событие нажатие кнопки – как сделать так, чтобы запоминалась информация о предыдущей нажатой кнопке и обработка этой информации. Использование задержки sleep из модуля time для отображения изображений, если при двух последовательных нажатиях были открыты разные картинки.

Урок 59. Практика: Разработка и создание игры “Мемори” – 1ч.

Завершение работы над проектом. Тестирование проекта и доработка мелочей (если изображения уже открыты, то на кнопки нельзя нажимать, если открыты все изображения, то игра завершена и т.п.). Тестирование проектов одноклассников. Добавление счетчика ходов. Работа с изображениями в Python. Замена текстовых надписей в приложении “Мемори” изображениями.

Раздел 5. Создание собственных проектов (11 часов)

Урок 60-67. Теория: Создание проектов по собственному замыслу – 2ч.
Практика: Создание проектов по собственному замыслу – 6ч.
Урок 68-69. Практика: Защита проектов – 2ч.
Урок 70. Теория: Итоговое занятие Подведение итогов года – 1ч.

Учебно-тематический план.

№ п/п	Название раздела	Количество часов			Формы контроля
		Всего	Теория	Практика	
1.	Техника безопасности. Вводное занятие	1	1	0	собеседование, фронтальный опрос
2.	Введение в язык программирования Python	30	7	23	опрос, самостоятельные работы
3.	Создание приложений с помощью tkinter	15	5	10	опрос, самостоятельные работы
4.	Продвинутые библиотеки языка Python. Pygame	13	3	10	опрос, самостоятельные работы
5.	Создание собственных проектов	11	3	8	опрос, самостоятельные работы от
	Итого:	**70**	**19**	**51**	

Учебно-календарный график.

№ п/п	Месяц	Число	Форма занятия	Тема занятия	Кол-во часов	Место проведения	Формы контроля
1	сентябрь			**Раздел 1. Техника безопасности. Вводное занятие**	**1**		Фронтальный опрос
1.1		04.09	Теоретическое	Техника безопасности. Вводное занятие	1	МБОУ АСОШ №2	
2.				**Раздел 2. Введение в язык программирования Python**	**30**		Фронтальный опрос. Решение практических задач
2.1		04.09	Теоретическое	Знакомство с языком программирования Python.	1	МБОУ АСОШ №2	
2.2.		11.09	Практическое	Структура программы. Типы данных. Переменные.	1	МБОУ АСОШ №2	
2.3		11.09	Практическое	Линейные алгоритмы.	1	МБОУ АСОШ №2	
2.4-2.5		18.09 18.09	Комбинированное	Ветвящиеся алгоритмы.	2	МБОУ АСОШ №2	
2.6-2.8		25.09 25.09 02.10	Комбинированное	Циклические алгоритмы.	3	МБОУ АСОШ №2	
2.9	октябрь	02.10	Практическое	Вложенные циклы.	1	МБОУ АСОШ №2	
2.10		09.10	Практическое	Цикл for в Python.	1	МБОУ АСОШ №2	
2.11		09.10	Практическое	Цикл while	1	МБОУ АСОШ №2	
2.12-2.13		16.10 16.10	Комбинированное	Списки.	2	МБОУ АСОШ №2	
2.14		23.10	Практическое	Функции.	1	МБОУ АСОШ №2	
2.15		23.10	Практическое	Модули.	1	МБОУ АСОШ №2	
2.16		30.10	Практическое	Работа с текстовыми файлами.	1	МБОУ АСОШ №2	
2.17-2.18		30.10 06.11	Комбинированное	Практическая работа.	2	МБОУ АСОШ №2	

2.19-2.23	ноябрь	06.11 13.11 13.11 20.11 20.11	Комбинированное	Графический модуль PyTurtle.	5	МБОУ АСОШ №2	
2.24		27.11	Практическое	Практическая работа.	1	МБОУ АСОШ №2	
2.25-2.26		27.11 04.12	Комбинированное	Графика с модулем tkinter в Python.	2	МБОУ АСОШ №2	
2.27	декабрь	11.12	Практическое	Виджет Canvas.	1	МБОУ АСОШ №2	
2.28		11.12	Практическое	Практическая работа	1	МБОУ АСОШ №2	
2.29		18.12	Практическое	Создание программы-теста из нескольких вопросов.	1	МБОУ АСОШ №2	
2.30		18.12	Практическое	Завершение работы над программой-тестом.	1	МБОУ АСОШ №2	
3				**Раздел 3. Создание приложений с помощью tkinter**	**15**		Фронтальный опрос. Решение практических задач
3.1		25.12	Теоретическое	Знакомство с модулем tkinter.	1	МБОУ АСОШ №2	
3.2		25.12	Теоретическое	Что такое виджеты, конфигурация виджетов.	1	МБОУ АСОШ №2	
3.3	январь	15.01	Теоретическое	События в tkinter.	1	МБОУ АСОШ №2	
3.4-3.6		15.01 22.01 22.01	Комбинированное	Создание простых приложений.	3	МБОУ АСОШ №2	
3.7		29.01	Практическое	Продвинутое рисование в tkinter.	1	МБОУ АСОШ №2	
3.8		29.01	Практическое	Движение нарисованных объектов.	1	МБОУ АСОШ №2	
3.9-3.10	февраль	05.02 05.02	Комбинированное	Создание игрового приложения “Пинг-понг”.	2	МБОУ АСОШ №2	
3.11		12.02	Практическое	Приложение “Сапер”. Информация о минах.	1	МБОУ АСОШ №2	
3.12		12.02	Практическое	Приложение “Сапер”. Как узнать, сколько мин среди	1	МБОУ АСОШ №2	

				соседей данной клетки и как эту информацию хранить?			
3.14		19.02	Практическое	Создаем “мозг” игры.	1	МБОУ АСОШ №2	
3.15		19.02	Практическое	Завершение работы над приложением “Сапер”.	1	МБОУ АСОШ №2	
4				**Раздел 4. Продвинутые библиотеки языка Python. Pygame**	**13**		
4.1		26.02	Теоретическое	Знакомство с продвинутыми фреймворками для разработки GUI-приложений.	1	МБОУ АСОШ №2	
4.2		26.02	Теоретическое	Шаблон программы на Pygame	1	МБОУ АСОШ №2	
4.3		05.03	Практическое	Геометрические примитивы в Pygame	1	МБОУ АСОШ №2	
4.4		05.03	Практическое	Простая анимация в Pygame.	1	МБОУ АСОШ №2	
4.5		12.03	Практическое	События клавиатуры.	1	МБОУ АСОШ №2	Фронтальный опрос. Решение практических задач
4.6		12.03	Практическое	События мыши.	1	МБОУ АСОШ №2	
4.7		19.03	Практическое	Дополнительные поверхности в Pygame.	1	МБОУ АСОШ №2	
4.8		19.03	Практическое	Работа в команде. Создание приложения “Поймай звезду”.	1	МБОУ АСОШ №2	
4.9		26.03	Практическое	Продолжение работы над игрой. Текст в Pygame.	1	МБОУ АСОШ №2	
4.10	март	26.03	Практическое	Завершение работы над проектом. Музыка в Pygame.	1	МБОУ АСОШ №2	
4.11-4.13		02.04 02.04 09.04	Комбинированное	Разработка и создание игры “Мемори”	3	МБОУ АСОШ №2	
5				**Раздел 5. Создание собственных проектов**	**11**		Фронтальный опрос. Решение практических задач
5.1-5.8	апрель	09.04 16.04	Комбинированное	Создание проектов по собственному	8	МБОУ АСОШ №2	

		16.04 23.04 23.04 30.04 30.04 07.05		замыслу.			
5.9- 5.10		07.05 14.05	Практическое	Защита проектов.	2	МБОУ АСОШ №2	
5.11	май	14.05	Теоретическое	Итоговое занятие Подведение итогов года.	1	МБОУ АСОШ №2	
				Итого:	**70**		

Материально-техническое обеспечение

Для реализации настоящей программы требуется:

- наличие компьютерного класса, с оборудованием, соответствующим санитарным нормам;
- 10 ученических мест;
- каждое учебное место должно быть оборудовано 1 компьютером с установленным программным обеспечением;
- среда программирования языка Python
- принтер, сканер;
- маркерная доска;
- видеопроектор.

МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

✓ **Методы обучения**

Методическое обеспечение – это методы и технологии, используемые для реализации образовательного процесса.

В процессе реализации программы используются следующие методы организации занятий:

✓ словесные методы (лекция, объяснение);
✓ демонстративно-наглядные (демонстрация работы в программе, схем, скриптов, таблиц);
✓ исследовательские методы;
✓ работа в парах;
✓ работа в малых группах;
✓ проектные методы (разработка проекта по спирали творчества, моделирование, планирование деятельности)
✓ работа с Интернет-сообществом (публикация проектов в Интернет-сообществе скретчеров).

Практическая часть работы – работа в среде программирования со скриптами и проектирование информационных продуктов. Для наилучшего усвоения материала практические задания рекомендуется выполнять каждому за компьютером. При выполнении глобальных проектов рекомендуется объединять школьников в пары. Текущий контроль уровня усвоения материала осуществляется по результатам выполнения рефлексивных упражнений и практических заданий. Итоговый контроль осуществляется по результатам разработки проектов. Формы подведения итогов: презентация проекта, испытание квеста, игры.

Для успешной реализации программы используются следующие методические материалы:

✓ учебно-тематический план;
✓ календарно-тематический план;
✓ теоретический материал по изучаемым темам;

- ✓ инструкции по технике безопасности и правилам поведения в учреждении;
- ✓ справочники и переводчики в электронном виде;
- ✓ методическая литература для педагогов дополнительного образования.
- ✓ Дидактическое обеспечение программы включает в себя следующие материалы:
- ✓ учебные презентации по темам;
- ✓ материалы для практических и самостоятельных заданий;
- ✓ материалы для проведения конкурсных мероприятий.

Форма организации образовательного процесса – групповая

Формы проведения занятий:

- ✓ занятие с использованием игровых технологий;
- ✓ занятие-игра;
- ✓ занятие-исследование;
- ✓ творческие практикумы (сбор скриптов с нуля);
- ✓ занятие-испытание игры;
- ✓ занятие-презентация проектов;
- ✓ занятие с использованием тренинговых технологий (работа на редактирование готового скрипта в соответствии с поставленной задачей).

Формы организации учебного занятия:

Форма организации деятельности групповая, при этом отдельные вопросы и ошибки рассматриваются в индивидуальном порядке с каждым обучающимся, исходя из особенностей каждого обучающегося в усвоении пройденного материала.

Первая часть занятия предполагает получение обучающимся нового материала. Во время второй части занятия обучающийся пытается самостоятельно реализовать полученную теоретическую базу в рамках собственного проекта. Оценка результатов производится коллективно всей группой.

Некоторые занятия полностью отведены на реализацию проектной работы.

Общение на занятии ведётся в свободной форме — каждый обучающийся в любой момент может задать интересующий его вопрос без поднятия руки. Данный момент очень важен в процессе обучения, так как любой невыясненный вопрос, может превратиться в препятствие для получения обучающимся последующих знаний и реализации им собственных проектов.

- ✓ **Педагогические технологии**

овременные бразовательные ехнологии /или методики	**Цель использования технологий и/или методик**	**Описание внедрения технологий и/или методик в практической профессиональной деятельности**	**Результат использования технологий и/или методик**
ичностно-риентированная	Максимальное развитие, а не информирование заранее данных индивидуальных познавательных способностей ребенка на основе использования имеющегося у него	Раскрытие возможностей каждого обучающегося, организация совместной, познавательной, творческой деятельности каждого ребенка.	Раскрытие и использование субъективного опыта каждого обучающегося, становление личности путем организации познавательной деятельности.

	опыта		
доровьесбере-ающая	Сохранение, формирование и укрепление здоровья обучающихся.	Профилактика сколиоза, укрепление мышечного корсета, коррекция недостатков осанки.	Формирование у детей необходимых знаний, навыков по здоровому образу жизни, использование обучающимися полученных знаний в повседневной жизни.
рупповая	Выявить, учесть, развить творческие способности детей и приобщить их к многообразной творческой деятельности с выходом на конкретный продукт, который можно фиксировать (комбинация, этюд, номер и т.д.) воспитание общественно-активной творческой личности и способствует организации социального творчества, направленного на служение людям в конкретных	Обучение осуществляется путем общения в динамических группах, когда каждый учит каждого. Обучение есть общение обучающихся и обучаемых.	Воспитывает чувство товарищества, взаимовыручки, взаимопомощи и понимания, что влияет на сплоченность коллектива и способность ребенка найти свое место в детском обществе.

	социальных ситуациях		
КТ	Развитие мышления, развитие коммуникативных способностей, развитие навыков исследовательской деятельности, формирование умений принимать решения в сложных ситуациях, формирование информационной культуры.	Использование компьютерных технологий, как основной компонент для организации учебного процесса.	Формирование и развитие базовых знаний использования новых информационно-коммуникативных технологий как в учёбе, так и в другой деятельности человека

Система контроля результативности

Оценочные материалы

По завершении изучения крупных тем или в конце учебного года целесообразно проведение нескольких занятий в форме конференции, где бы каждый ученик или группа учеников могли представить свою работу, по заинтересовавшей их тематике.

№	Критерий	Оценка (в баллах)
1	Актуальность поставленной задачи	3 – имеет большой интерес (интересная тема) 2 – носит вспомогательный характер 1 – степень актуальности определить сложно 0 – не актуальна
2	Новизна решаемой задачи	3 – поставлена новая задача 2 – решение данной задачи рассмотрено с новой точки зрения, новыми методами 1 – задача имеет элемент новизны 0 – задача известна давно
3	Оригинальность методов	3 – задача решена новыми оригинальными

	решения задачи	методами 2 – использование нового подхода к решению идеи 1 – используются традиционные методы решения
4	Практическое значение результатов работы	2 – результаты заслуживают практического использования 1 – можно использовать в учебном процессе 0 – не заслуживают внимания
5	Насыщенность элементами мультимедийности	Баллы суммируются за наличие каждого критерия 1 – созданы новые объекты или импортированы из библиотеки объектов 1 – присутствуют текстовые окна, всплывающие окна, в которых приводится пояснение содержания проекта 1 – присутствует музыкальное оформление проекта, помогающего понять или дополняющего содержание (музыкальный файл, присоединенный к проекту) 1 – присутствует мультипликация
6	Наличие скриптов (программ)	2 – присутствуют самостоятельно, созданные скрипты 1 – присутствуют готовые скрипты 0 – отсутствуют скрипты
7	Уровень проработанности решения задачи	2 – задача решена полностью и подробно с выполнением всех необходимых элементов 1 – недостаточный уровень проработанности решения 0 – решение не может рассматриваться как удовлетворительное
8	Красочность оформления работы	2 – красочный фон, отражающий (дополняющий) содержание, созданный с помощью встроенного графического редактора или импортированный из библиотеки рисунков 1 – красочный фон, который частично отражает содержание работы 0 – фон тусклый, не отражает содержание работы
9	Качество оформления работы	3 – работа оформлена изобретательно, применены нетрадиционные средства, повышающие качество описания работы 2 – работа оформлена аккуратно, описание четко, последовательно, понятно, грамотно 1 – работа оформлена аккуратно, но без «изысков», описание непонятно, неграмотно
	Максимальное количество баллов	**24 балла**

Воспитательная работа с обучающимися

Воспитательная работа является неотъемлемой частью образовательного процесса. Можно выделить два основных направления воспитательной работы: формирование мировоззрения и воспитание нравственных качеств, таких как ответственность, трудолюбие, вежливость, терпение и др. Реализация воспитательной работы осуществляется через ряд мероприятий.

№	Проводимые мероприятия	Сроки проведения
	Беседа о правилах поведения в компьютерном классе. Проведение инструктажей по технике безопасности.	В течение года
	Организация взаимопомощи в учебе	На каждом занятии
	Беседа «Мы и компьютер»-охрана зрения ,осанки.	В течение года
	Организация минуты отдыха на учебных занятиях	На каждом занятии
	Проектная деятельность	В течение года
	Участие в конкурса, олимпиадах, а так же разработка и проведение собственных конкурсов и олимпиад	В течение года
	Представление достижений, результатов, способностей учащихся родителям, педагогам, сверстникам.	В течение года
	Беседы о правилах дорожного движения	В течение года

Литература для педагога

1. Бизли, Дэвид М. Python. Подробный справочник. – М.–СПб.: Символ-Плюс, 2010.
2. Лутц, Марк Python. Справочник. – М.: Вильямс, 2015.
3. Официальный сайт программы [Электронный ресурс] – Режим доступа: https://docs.python.org/, свободный.
4. Сайт, среда разработки для языка Python. [Электронный ресурс] – Режим доступа: https://www.jetbrains.com/pycharm/?fromMenu, свободный.

Литература для родителей и учащихся

1. Сайт / справочные материалы [Электронный ресурс] – Режим доступа: https://metanit.com/python/, свободный.
2. Сайт / интерактивный сборник задач для практики программирования [Электронный ресурс] – Режим доступа: http://pythontutor.ru/, свободный.
3. Сайт / Адаптивный тренажер Python [Электронный ресурс] – Режим доступа: https://stepik.org/course/431 , свободный.
4. Сайт / среда разработки для языка Python [Электронный ресурс] – Режим доступа: https://www.jetbrains.com/pycharm/?fromMenu , свободный.

Сайт проекта Open Book Project. Практические примеры на Python Криса Мейерса [Электронный ресурс] – Режим доступа: openbookproject.net , свободный

Интернет-ресурсы

1. Официальный сайт языка Python https://www.python.org/
2. Интерактивный Python. Trinket https://trinket.io/python
3. Интерактивный учебник языка Питон http://pythontutor.ru/
4. Python 3 для начинающих https://pythonworld.ru/
5. Python GUI Programming (Tkinter) https://www.tutorialspoint.com/python/python_gui_programming.htm http://www.tkdocs.com/tutorial/ и http://effbot.org/tkinterbook/
6. Python IDE for beginners http://thonny.org/
7. Бесплатный, простой и профессиональный инструмент для изучения программирования с помощью Python https://www.jetbrains.com/pycharm-edu/

8. Образовательный портал pythonicway.com http://pythonicway.com