**Муниципальное бюджетное общеобразовательное учреждение**

**Костомукшского городского округа «Гимназия»**

**УТВЕРЖДЕНО:**

**Директор МБОУ КГО «Гимназия»**

**\_\_\_\_\_\_\_\_\_\_\_ /И.В. Осипенко/**

**«01» сентября 2020 г.**

**Рабочая программа**

**по внеурочной деятельности**

**Общекультурное направление**

**Кружок по информатике «Приручи Питона!»**

**Для 7-8 класса**

**1 год**

Срок реализации

 **Разработчик:**

**Доромейчик Ю.А.,**

**учитель информатики.**

**Обсуждена и согласована на**

**заседании методического объединения**

**Протокол № 1 от ­­31 августа 2020 г.**

**г. Костомукша**

**2020**

**Пояснительная записка.**

Компьютеры занимают очень важное место в современном мире, и мы принимаем это как должное. Вместо звонков по телефону мы посылаем текстовые сообщения или используем социальные сети. Однако мы можем не только использовать эти технологии – научившись программировать, мы можем развивать их, создавать собственные произведения цифрового искусства.

Преподавание программирования в школе имеет очень старые традиции. Собственно, основу курса информатики на первых порах его введения в школьную программу, составляло обучение программированию. Затем, в связи с широким внедрением в жизнь информационных технологий и поставками в школу нового оборудования, наиболее важной составляющей курса информатики стало обучение информационным технологиям.

Сегодня наука и технология развиваются столь стремительно, что образование зачастую не успевает за ними. Например, для того, чтобы успешно выступать на Российских олимпиадах по информатике, надо серьёзно заниматься, начиная с начальной школы. Для этого проводятся конкурсы, викторины, олимпиады и другие мероприятия.

Уже в младшем школьном возрасте интересы многих ребят претерпевают существенные изменения, и большинство из них сильно удаляются от учебной деятельности вообще и научно-познавательной – в частности. Это можно объяснить разными причинами (что и делают психологи и педагоги), но одной из наиболее серьёзных таких причин, несомненно, является неуспешность наших детей в учебной деятельности или боязнь такой неуспешности. Как показывают исследования психологов, боязнь потерпеть неудачу в школе дети ставят на второе место (по силе стресса), сразу после смерти родителей. В такой ситуации одной из важнейших задач педагогов следует считать создание комфортной учебно-воспитательной среды, в которой возможна наиболее полная самореализация ребёнка.

Другой причиной снижения интереса учеников, имеющей отношение собственно к информатике, является очень небольшое количество часов, предусмотренное федеральными образовательными стандартами на изучение данного предмета. Более того, стандарты второго поколения вообще не включают дисциплину «Информатика», заменяя её «Информационными технологиями». Основной акцент предполагается сделать на приобретении учениками информационной грамотности, подразумевающей общие навыки обработки информации различных видов. Теоретическая же подготовка по информатике вынесена во внеучебную деятельность и, таким образом, носит факультативный характер.

Поэтому наиболее остро стоит проблема изучения возможных подходов к организации кружковой образовательной деятельности по информатике в начальной школе. Опыт таких видных педагогов как М.П. Лапчик, И.Г. Семакин, Е.К. Хеннер свидетельствует, что такая деятельность позволяет, с одной стороны, организовать среду для самореализации и самоутверждения учеников, и, с другой стороны, сформировать у них тягу к творчеству и знаниям и дать подходящие средства её реализации. Решение выше указанной проблемы может базироваться на использовании в обучении информатике языка программирования Python.

Python – это текстовый язык программирования. Он универсален, пригоден для создания самых разных программ, от текстовых процессоров до веб-браузеров. Вот несколько причин, почему именно этот язык я предлагаю изучать в рамках кружковых занятий:

1. Python – простой и удобный язык. По сравнению со многими другими языками читать и составлять программы на Python совсем не сложно;
2. В Python есть библиотеки готовых процедур для использования в своих программах. Это позволяет создавать сложные программы быстро;
3. Python используется серьёзными фирмами. Например, его используют в Google.

Курс построен таким образом, чтобы помочь учащимся заинтересоваться программированием вообще и найти ответы на вопросы, с которыми им приходится сталкиваться в повседневной жизни при работе с большим объемом информации; научиться общаться с компьютером, который ничего не умеет делать, если не умеет человек.

**Цель кружка**: изучение пропедевтики программирования в игровой, увлекательной форме, используя язык Python.

**Задачи кружка:**

- Приобретение положительного опыта. Часто дети теряют интерес к предмету в процессе изучения синтаксиса и грамматики языка. Синтаксические проблемы описания циклов и ветвлений многим кажутся непреодолимыми. Много времени занимает просто кодирование – не все быстро работают на клавиатуре. Учитывая простоту языка Python эти проблемы легко решаются.

- На основе изучения программирования и алгоритмизации развивать мышление, формирование научного мировоззрения школьников именно этой возрастной группы.

- Подготовить к более успешному усвоению базового и профильного курса «Информатика и ИКТ» в старших классах.

**Место кружка в учебном плане.**

Программа кружка «Компьютерная графика» предназначена для обучающихся 7-8 классов и рассчитана на один год обучения. Ребята занимаются по одному часу в неделю, что составит 34 учебных часа на год.

**Планируемые результаты освоения учебного предмета.**

Python – серьёзный язык программирования. В то же время учащиеся в полной мере могут раскрыть свои творческие таланты, так как с его помощью можно легко создавать игры и другие приложения.

**Способы определения результативности занятий.**

- Наблюдение.

- Устный контроль

- Практическая работа.

**Содержание учебного предмета.**

**Знакомство учащихся с устройством языка Python.**

Техника безопасности на занятии. Знакомство с интерфейсом программы.

**Установка Python. Среда разработки IDLE. Сохранение программы. Ошибки в окне программы. Ошибки в окне консоли. Поиск ошибок.**

Скачивание и установка на компьютер Python 3. Знакомство со средой IDLE – это ориентированная на начинающих среда разработки, в которой есть несложный текстовый редактор для написания и отладки Рython- программ.

**Проект №1. «Дом с привидениями». Структура программы: подготовка к игре, основной цикл, конец игры.**

В этой игре показаны некоторые основные возможности Python. Программа разбирается по частям для понимания её устройства: подготовка к игре, основной цикл, конец игры. Основные понятия: ввод программы, запуск программы, структурирование программы, понятие переменных, отображение текста.

**Ветвление на языке Python. Использование ветвления в основном цикле игры.**

Понятие «ветвления». Программа выполняет разные действия в зависимости от того, выполняется ли условие. Оператор «ветвления».

**Два вида окон в IDLE: окно программы и окно консоли.**

Два вида окон в IDLE: окно программы и окно консоли. Сравнение этих окон и их возможностей.

**Переменные в Python. Создание переменной. Использование переменных.**

Создание переменной, изменение значения переменной, имена переменных, использование переменных, печать переменных.

**Типы данных: числа, строки, булевы значения. Преобразование типов данных.**

Данные, которыми пользуются компьютеры (и языки программирования), представлены несколькими типами, и это вполне очевидно. Числа бывают дробными, целыми, могут состоять из множества цифр или быть весьма массивными из-за дробной части. Чтобы интерпретатору было проще работать с ними, и он мог понять, с чем имеет дело, следует задать определенный тип. Более того, он необходим, чтобы числа поместились в отведенную ячейку памяти. Наиболее распространенные типы данных, которым пользуется язык программирования Python: Integer. Речь идет о целых числах, имеющих как отрицательное, так и положительное значение. Ноль также входит в данный тип. Для того чтобы интерпретатор понял, что работает с дробными частями, следует задать тип float point. Как правило, им пользуются в случае использования чисел с варьирующейся точкой. Следует помнить, что при написании программы нужно придерживаться записи «3.25», а не использовать запятую «3,25». В случае добавления строк язык программирования Python позволяет добавить тип string. Зачастую слова или фразы заключаются в одинарные или двойные кавычки.

**Вычисления в Python. Простые вычисления. Использование скобок. Результат в переменной. Случайные числа.**

Простые вычисления: сложение, вычитание, умножение, деление. Использование скобок, результат в переменной, случайные числа. Функция randint().

**Строки в Python. Создание строк. Сложение строк. Длина строки. Нумерация символов. Апострофы.**

 Строки в переменных. Сложение двух строк. Добавление третьей строки в середине. Длина строки. Нумерация символов. Отсчёт символов. Срез символов. Апострофы. Экранирование апострофа.

**Ввод и вывод. Использование input(). Функция print(). Вывод в окне консоли.**

Команда input(). Использование input(). Вывод в окне консоли. Функция print(). Использование print(). Два способа разделять вывод. Три способа завершить вывод: точка в конце, вывод без перевода строки, с новой строки. Пустые строки в конце.

**Логические операции. Виды операций сравнения. Проект № 2 «Проверка на день рождения».**

Понятие логических операций. Виды операций сравнения. Проект 2: программа запрашивает номер дня и месяца, а затем с помощью логических операций выясняет, не пора ли праздновать. Сравнение строк. Строковые операции (операция in).

**Ветвление. Условие if. Условие if-else. Условие if-elif-else**

Условие if. Программа: спрашивается у пользователя, не день ли рождения у него. Если тот ответит «Y», программа напечатает поздравление Условие if-else. Программа: если ввести «Y» программа напечатает новогоднее поздравление, если ввести что-нибудь другое – появится другое сообщение. Условие if-elif-else.

**Циклы в Python. Повтор команд. Цикл for. Переменные цикла. Вложенные циклы.**

Управление Черепашкой. Используем цикл for. Переменные цикла: отсчёт, отсчёт по двойкам, обратный отсчёт. Вложенные циклы: цикл внутри цикла.

**Цикл while. Блок repeat until. Бесконечный цикл. Выход из цикла.**

Понятие цикла while. Использование цикла while. Бесконечный цикл : создание бесконечного цикла. Остановка цикла. Команда break внутри цикла. Команда continue(продолжить).

**Списки. Что такое списки? Пример списка. Использование списков.**

Понятие списка. Пример списка. Использование списков. Списки списков. Списки и циклы. Сложение списков.

**Функции. Полезные функции. Создание и вызов функций. Передача данных в функцию. Получение данных из функции**.

Понятие функции. Создание и вызов функций: определение функции, вызов функции. Параметры функции. Передача значений. Получение данных из функции. Определение функции, которая возвращает число.

**Итоговая работа. Проект-игра «Охота за пузырями»**

 Описание игры: игрок управляет подводной лодкой. Стрелки на клавиатуре перемещают подлодку. Справой стороны экрана на лодку налетают пузыри. За протыкание пузырей начисляются очки. Первоначально время игры равно 30 секунд. Каждая 1000 очков добавляет ещё времени. Игра заканчивается, когда истекает время.

**Календарно-тематический план**

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **темы занятий** | **Всего часов** | **В том числе** |
| **Теория** | **Практика** |
| **1** |  Знакомство учащихся с устройством языка Python | **2** | **1** | **1** |
| **2** | Установка Python. Среда разработки IDLE. Сохранение программы. Ошибки в окне программы. Ошибки в окне консоли. Поиск ошибок. | **2** | **1** | **1** |
| **3** | Проект №1. «Дом с привидениями». Структура программы: подготовка к игре, основной цикл, конец игры. | **2** | **1** | **1** |
| **4** | Ветвление на языке Python. Использование ветвления в основном цикле игры. | **2** | **1** | **1** |
| **5** | Два вида окон в IDLE: окно программы и окно консоли.  | **2** | **1** | **1** |
| **6** | Переменные в Python. Создание переменной. Использование переменных. | **2** | **1** | **1** |
| **7** | Типы данных**:** числа, строки, булевы значения. Преобразование типов данных. | **2** | **1** | **1** |
| **8** | Вычисления вPython. Простые вычисления. Использование скобок. Результат в переменной. Случайные числа. | **2** | **1** | **1** |
| **9** | Строки вPython. Создание строк. Сложение строк. Длина строки. Нумерация символов. Апострофы. | **2** | **1** | **1** |
| **10** | Ввод и вывод. Использование input(). Функция print(). Вывод в окне консоли. | **2** | **1** | **1** |
| **11** | Логические операции. Виды операций сравнения. Проект № 2 «Проверка на день рождения». | **2** | **1** | **1** |
| **12** | Ветвление. Условие if. Условие if-else. Условие if-elif-else | **2** | **1** | **1** |
| **13** | Циклы в Python. Повтор команд. Цикл for. Переменные цикла. Вложенные циклы. | **2** | **1** | **1** |
| **14** | Цикл while. Блок repeat until. Бесконечный цикл. Выход из цикла. | **2** | **1** | **1** |
| **15** | Списки. Что такое списки? Пример списка. Использование списков. | **2** | **1** | **1** |
| **16** | Функции. Полезные функции. Создание и вызов функций. Передача данных в функцию. Получение данных из функции. | **2** | **1** | **1** |
| **17** | Итоговая работа.Проект-игра «Охота за пузырями» | **2** |  | **2** |