

1. **Пояснительная записка**

Программа составлена на основе дополнительной общеобразовательной общеразвивающей программы «Введение в программирование на Python» автор Резванов Р.,Алгоритмика, Москва 2020г.

Современный мир предъявляет новые требования к молодому поколению, вступающему в жизнь, так как будущее сегодняшних детей — это информационное общество. Психологическая готовность жизни в мире информационных технологий (ИТ) необходима каждому человеку.

Информатизация школьного образования открывает новые возможности для развития методов и организационных форм воспитания и обучения детей. Для успешного обучения в школе важен не столько набор знаний, сколько развитое мышление, умение получать знания, использовать имеющиеся навыки для решения различных учебных задач. Большие возможности при этом раскрываются при работе с компьютером.

**1.1. Направленность и уровень Программы**

*Направленность* Программы — техническая, заключается в обобщение и углубление знаний языка программирования Python, а также развитии логического и алгоритмического мышления для дальнейшего освоения программирования с использованием языка Python.

Техническое и творческое программирование у детей средних и старших классов, формирует у них первичные представления о сфере программирования в целом и проводит профориентацию ученика на профессию - программист. Также развивает новые общеинтеллектуальные умения и навыки: составлять план деятельности, разделять задачу на этапы решения, проявлять познавательную и исследовательскую активность и другое.

Творческие и практические задания, связанные с цифровыми инструментами и программированием являются одним из способов формирования интереса к техническим направлениям, а также стимулирует рационализаторские, изобретательские способности.

**1.2. Актуальность**

Компьютерное обучение — это новый способ обучения, одной из разновидностей которого можно считать использование обучающих игровых программ и приложений. Школьник овладевает новыми инструментами, что позволяет более эффективно и просто получать и обрабатывать информацию, а также открывает новые возможности для развития творческого потенциал.

Использование компьютерных технологий в работе с детьми эффективно решает образовательные задачи, которые будут способствовать повышению мотивации ребенка.

**Актуальность Программы заключается в:**

* востребованности развития широкого кругозора школьника средних и старших классов;
* формировании и развитии навыков начального программирования в условиях модернизации образования;
* развитии логического мышления, творчества посредством проектной деятельности;
* профориентировании ученика на актуальную и востребованную сегодня и ближайшем будущем профессию;
* формировании и развитии эмпатии и навыков командной работы, как важных составляющих успешной деятельности в будущем.

**Новизна Программы** заключается в технической направленности обучения, которое базируется на новых ИТ, что способствует развитию информационной культуры. Программа рассчитана на детей в возрасте 12–17 лет, проявляющих интерес к IT-технологиям, желающих совершенствовать свои навыки программирования, имеющих первичный опыт объектно-ориентированного и функционального программирования в различных интегрированных средах разработки на языке Python.

Во время обучения школьники познакомятся с новыми инструментами и конструкциями для программирования на языке Python. Тренажер помогает запомнить синтаксис языка, а продуманные практические и творческие задания делают этот процесс легче и увлекательней. Освоение языка проходит в увлекательном формате с использованием учебно-методического комплекса.

**1.3. Цель и задачи Программы**;

**Цель программы —** углубление и структурирование знаний основ языка программирования Python; умение на практике использовать сложные структуры данных; работать со специальными средствами и библиотеками языка Python, а также познакомить учеников с модулем turtle.

Достижение цели обеспечивается решением следующих основных **задач** программы:

*Обучающие:*

1. Познакомить ученика с современными информационными технологиями.
2. Познакомить с использованием сложных структур данных – списков, функций и т.д.
3. Познакомить ученика с синтаксисом модуля turtle языка Python при помощи визуального программирования.
4. Научить использовать различные компиляторы для разработки проектов на языке Python.
5. Закрепить и развить навык программирования на языке Python при помощи практических и творческих задач.
6. Сформировать навыки решения прикладных задач.
7. Сформировать практический навык разработки простых приложений.

*Развивающие:*

1. Формировать и развивать алгоритмическое и логическое мышление.
2. Развивать познавательный интерес, навык планирования, способность к рефлексии и самооценке.
3. Развивать интерес учащихся к изучению программирования.

*Воспитательные:*

1. Воспитывать у учеников потребность в сотрудничестве, взаимодействии со сверстниками, умение подчинять свои интересы определенным правилам.
2. Формировать у детей постоянное стремления к развитию профессиональных способностей и мастерства.
3. Формировать информационную культуру.

**1.4. Формы** **и режим занятий***;*

Занятия проходят 1 раз в неделю и длятся 1 академический час. Дети занимаются в классе с учителем (групповая форма занятий). Занятие состоит из следующих частей: вводное повторение и разминка, проблематизация, новый материал, развитие умений, рефлексия.

Методика обучения ориентирована на индивидуальный подход. Для того чтобы каждый подросток получил наилучший результат обучения, программой предусмотрены индивидуальные задания для самостоятельной работы. Для данного типа работы используются разные по уровню сложности задания, которые носят репродуктивный и творческий характер. Количество таких заданий в работе может варьироваться. В ходе обучения проводится промежуточное тестирование по темам для определения уровня знаний обучающихся.

*Формы обучения:*

* Игровая, задачная и проектная.
* Обучение от общего к частному.
* Поощрение вопросов и свободных высказываний по теме.
* Уважение и внимание к каждому ученику.
* Создание мотивационной среды обучения.
* Создание условий для дискуссий и развития мышления учеников при достижении учебных целей вместо простого одностороннего объяснения темы преподавателем.

**1.5. Срок реализации Программы**

Содержание Программы для учеников 7–11 классов рассчитано на обучение в течение 16 недель (16 занятий — 16 академических часов).

**1.6. Планируемые результаты**

Достижение цели и задач образовательной программы предполагает получение следующих результатов:

|  |  |  |
| --- | --- | --- |
| **Планируемый результат** | **Способ достижения** | **Критерий достижения образовательного результата** |
| **Предметные** | | |
| Развитие логического и алгоритмического мышления. | Ученик продумывает способ решения творческих и практических заданий. | Проявляет проектное мышление, строит логическую цепь рассуждений. Придумывает решения для творческих проектов с использованием логических и алгоритмических конструкций. |
| Развитие навыка создания кода на языке Python для решения практических и творческих задач. | Ученик решает тестовые и практические задания для программирования на языке Python. | Умеет использовать среду программирования Python. Умеет составлять правильный формат записи функций. Умеет подбирать корректные имена переменных. Умеет использовать оператор условия в программах. Умеет писать программы с вложенным оператором условия. Умеет формулировать и записывать условия проверки, используя операторы сравнения. Умеет писать конструкцию цикла и программы. Умеет использовать основные команды модуля turtle: forward(), backward(), left(), right(), clear(), penup(), pendown(), color(), width. Уметь использовать команды модуля turtle: begin\_fill(), end\_fill(). Умеет строить многоугольники и звезды в программе. Умеет строить графики в Python. Самостоятельно использует базовые конструкции языка Python и реализует групповые и индивидуальные проекты. |
| Развитие творческих способностей. | Ученик решает творческие задачи, поставленные на проектных уроках. | Придумывает, обсуждает, планирует и реализует свой проект. |

|  |  |  |
| --- | --- | --- |
| **Метапредметные** | | |
| Развитие и формирование учебных действий. | Создание благоприятных условий для участия в диалоге, в коллективном обсуждении. Ученик строит продуктивное взаимопонимание со сверстниками и взрослыми в процессе коллективной деятельности. | Легко общается, не боится просить помощь или оказать ее товарищу. Корректирует свое поведение при необходимости. |
| Умеет презентовать свою работу. | Ученик презентует свои проекты, учитель и другие ученики дают обратную связь. Учитель также дает советы, каким образом это лучше делать. | Во время презентации своих проектов пользуется вниманием аудитории. |

**2. Содержание Программы**

**2.1. Календарный учебный график**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **№**  **п/п** | **Наименование темы** | **Количество часов** | | | **Формы**  **аттестации (контроля)** | **Оборудование** |
| **всего** | **теория** | **практика** |
| 1 | Модуль 1. Понятие алгоритма и его характеристики | 1 | 0,5 | 0,5 | *Тестирование с*  *автопроверкой.* | *ИК*  *ПК* |
| 2 | Модуль 2. Повторение синтаксиса языка Python. Переменные. Операторы сравнения | 1 | 0,5 | 0,5 | *Тестирование с*  *автопроверкой.* |
| 3 | Модуль 3. Условный оператор. Вложенные условия. | 1 | 0,5 | 0,5 | *Тестирование с*  *автопроверкой.* |
| 4 | Модуль 4. Сложные структуры данных (списки, функции) | 2 | 0,5 | 1,5 | *Тестирование с*  *автопроверкой.* |
| 5 | Модуль 5. Подключение модуля turtle. Основные команды. | 2 | 0,5 | 1,5 | *Задания с*  *автопроверкой.* |
| 6 | Модуль 6. Создание многоугольников. Цикл while. | 2 | 0,5 | 1,5 | *Задания с*  *автопроверкой.* |
| 7 | Модуль 7. Цикл со счетчиком | 1 | 0,5 | 0,5 | *Задания с*  *автопроверкой.* |
| 8 | Модуль 8. Координаты. Получение и установка текущей позиции. | 2 | 0,5 | 1,5 | *Задания с*  *автопроверкой.* |
| 9 | Модуль 9. Проектируем задачу. Решение творческих задач. | 4 | 0 | 4 | *Представление решения группе.* | *МФУ*  *ИК*  *ПК* |
|  | **Всего** | **16** | **4** | **12** |  |  |

**2.2. Содержание учебного (тематического) плана**

**Модуль 1. Понятие алгоритма и его характеристик**

**1.1. Анонс Программы**

*Демонстрируем примерный проект, который ученики смогут реализовать в процессе обучения.*

**1.2. Повторение синтаксиса языка Python.**

*Демонстрируем: процесс запуска компилятора, интерфейс, выполнение простейших заданий.*

**1.3. Развиваем умения**

*Закрепление пройденного материала при помощи тестирования.*

**Модуль 2. Синтаксис языка Python. Переменные. Операторы сравнения**

**2.1. Поиск и отладка ошибок. Написание кода**

*Вводим и обсуждаем новые понятия урока: «программа», «переменная», «функция» и три правила синтаксиса языка: правило порядка, использование скобок и роль кавычек.*

**2.2. Разбор переменных и функций**

*Разбираем пример кода и находим в нем синтаксические ошибки. Демонстрируем написание кода на языке Python.*

**2.3. Развиваем умения**

*Закрепление пройденного материала при помощи тестирования.*

**Модуль 3. Условный оператор**

**3.1. Правила именования переменных**

*Формат записи. Разрешенные символы. Логика именования. Написание программы с использованием переменных.*

**3.1. Условный оператор**

*Понятие «полное ветвление» и «неполное ветвление», «вложенного условия». Разбираем блок-схемы. Демонстрируем пример синтаксиса условного оператора. Создание программы с использованием условного оператора.*

**3.3. Развиваем умения**

*Закрепление пройденного материала при помощи тестирования.*

**Модуль 4. Методы списков и строк. Функции**

**4.1. Списочные выражения.**

*Методы split и join. Другие методы списков и строк. Знакомство с функциями. Работа со списками, строками, множествами и кортежами в Python.*

**4.2. Функции**

*Понятие итератора, подпрограммы, процедуры, функции. Функции в Python.**Создание программы с использованием функций.*

**4.3. Развиваем умения**

*Закрепление пройденного материала при помощи тестирования.*

**Модуль 5. Подключение модуля turtle. Основные команды**

**5.1. Новые понятия**

*Вводим и обсуждаем новые понятия: «виды информации», «графическая информация», «растровая графика», «пиксель», «модуль».*

**5.2. Реализация графики в Python**

*Демонстрация примеров графики, созданной на Python. Знакомство с модулем turtle. Демонстрируем подключение и использование модуля turtle. Осваиваем команды управления в модуле turtle.*

**5.3. Развиваем умения**

*Закрепление пройденного материала при помощи заданий с автопроверкой.*

**Модуль 6. Создание многоугольников. Цикл while**

**6.1. Конструкция while**

*Демонстрируем устройство и принцип работы цикла while.*

**6.2. Использование цикла while для рисования**

*Рисование многоугольника и его заливка при помощи цикла while.*

**6.3. Развиваем умения**

*Закрепление пройденного материала при помощи заданий с автопроверкой.*

**Модуль 7. Цикл со счетчиком**

**7.1. Конструкция for**

*Демонстрируем устройство и принцип работы конструкции for.*

**7.2. Использование конструкции for для рисования**

*Рисование разнообразных фигур при помощи конструкции for*

**7.3. Развиваем умения**

*Закрепление пройденного материала при помощи заданий с автопроверкой.*

**Модуль 8. Координаты. Получение и установка текущей позиции**

**8.1. Напоминаем математические понятия**

*Разбор математических понятий, необходимых для урока: «координатная плоскость», «математические уравнения».*

**8.2. Рисование по координатам при помощи модуля turtle**

*Разбор синтаксиса команды goto(), pos(), xcor(), ycor(). Использование математический уравнений в Python для рисования. Рисование фигуры по координатам.*

**8.3. Развиваем умения**

*Закрепление пройденного материала при помощи заданий с автопроверкой.*

**Модуль 9. Проектируем задачу. Решение творческих задач.**

**8.1. Развиваем умения**

*Работа над мини-проектом. Представление выполненного задания группе.*

**2.3 Формы аттестации и оценочные материалы**

Текущий контроль сформированности результатов освоения программы осуществляется с помощью нескольких инструментов на нескольких уровнях:

* **на каждом занятии:** опрос, выполнение заданий на платформе, взаимоконтроль учеников в парах, самоконтроль ученика;
* **в конце каждого модуля:** тестирование, проведение презентации (по желанию) финальных проектов модуля и их оценка.

Для контроля сформированности результатов освоения программы с помощью цифровых инструментов используются формы Google. В каждом модуле ученики проходят тестовые задания (с автопроверкой), выполняют практические и творческие задания (проверяются учителем).

**3. Организационно-педагогические условия реализации Программы**

**3.1.** **Организационно-педагогические условия реализации Программы содержат:**

# **Материально-техническое обеспечение**

Цифровое оборудование, используемое на занятиях, кабинета центра образования цифрового и гуманитарного профилей «Точка роста»:

* МФУ (принтер, сканер, копир) Pantum M6550
* Ноутбуки мобильного класса HP ProBook x 360 (ПК)
* Интерактивный комплект на базе интерактивной панели Newline TT-7519RS (ИК)

Минимальные требования к техническим характеристикам каждого компьютера следующие:

 процессор – с тактовой частотой 2 ГГц;

 оперативная память – не менее 1 Гб;

 жидкокристаллический монитор с диагональю не менее 15 дюймов;

 жѐсткий диск – не менее 250 Гб;

 клавиатура;

 мышь;

 аудио-карта и акустическая система (наушники или колонки).

Требования к программному обеспечению компьютеров:

 операционная система Windows

 текстовый редактор (Блокнот) и текстовый процессор (MS Word);

 среда программирования Python 3+.

**учебно-методическое и информационное обеспечение Программы:**

* поурочные методические рекомендации к занятиям;
* задания на платформе с автопроверкой;
* бонусные практические и творческие задания на платформе;

**Список литературы**

1. Turtle graphics [Электронный ресурс]. – URL:: <https://docs.python.org/3/library/turtle.html> (дата обращения: 07.08.2022).
2. Подсадников А.В., Розов К.В., Русецкий А.В. «Основы программирования на языке Python 3» [Электронный ресурс]. – URL:: <https://mooc.nspu.ru/pluginfile.php/33787/mod_resource/content/0/Теория.pdf>  (дата обращения: 07.08.2022).
3. Курс «Язык программирования Python» http://www.intuit.ru/studies/courses/49/49/info .
4. Национальный открытый университет, курс «Введение в программирование на Python», <http://www.intuit.ru/studies/courses/12179/1172/info>