

Муниципальное бюджетное общеобразовательное учреждение
«Средняя общеобразовательная школа № 71»
Городского округа «Город Лесной» Свердловской области

ПРИНЯТО
Педагогическим советом
МБОУ СОШ № 71
Протокол № 323 от 26.02.2024



**Дополнительная общеобразовательная
общеразвивающая программа технической направленности
«Азбука программирования»
(с использованием оборудования центра образования естественно-научной
и технологической направленностей «Точка Роста»)**

Возраст учащихся: 14-16 лет

Срок реализации: 1 год

г. Лесной, 2024

Пояснительная записка

Дополнительная образовательная общеразвивающая программа технической направленности «Азбука программирования» с использованием оборудования центра «Точка Роста» разработана в соответствии со следующими документами:

1. Федеральный закон от 29.12.2012 № 273-ФЗ.
2. Паспорт национального проекта «Образование».
3. Профессиональный стандарт «Педагог (педагогическая деятельность в дошкольном, начальном общем, основном общем, среднем общем образовании) (воспитатель, учитель)».
4. Профессиональный стандарт «Педагог дополнительного образования детей и взрослых».
5. Федеральный государственный образовательный стандарт основного общего образования.
6. Федеральный государственный образовательный стандарт среднего общего образования.
7. Методические рекомендации по созданию и функционированию в общеобразовательных организациях, расположенных в сельской местности и малых городах, центров образования естественно-научной и технологической направленностей («Точка роста»).

В качестве основного инструмента обучения выбран язык программирования Python. Его использование способствует формированию у учащихся более прочных и глубоких знаний, умений и навыков при составлении различных алгоритмов и написании программ со сложной структурой.

Программа «Азбука программирования» имеет выраженную практическую направленность и способствует приобщению школьников к алгоритмической культуре, а также дает им возможность познать азы профессии программиста. Кроме того, выполняемые на занятиях задания способствуют развитию творчества учащихся, и формированию у них аналитического мышления, в том числе умения анализировать, систематизировать, визуализировать информацию, работать с большими массивами данных, что является одним из приоритетных требований многих современных работодателей.

Цель: создать условия для развития алгоритмического мышления учащихся, творческих способностей, аналитических и логических компетенций, формирования основных приемов программирования на современном языке программирования.

Задачи:

1. Обучающие:

- способствовать освоению учащимися всевозможных методов решения задач, реализуемых на языке Python;
- научить применять структурный подход для решения практических задач с использованием компьютера;
- расширить знания, умения и навыки решения задач по программированию и алгоритмизации;

- сформировать у учащихся навыки практической исследовательской деятельности.

2. Развивающие

• развивать стремление к самообразованию, обеспечить в дальнейшем социальную адаптацию в информационном обществе и успешную профессиональную и личную самореализацию;

- раскрыть креативные способности;
- способствовать развитию алгоритмического, творческого, логического и критического мышления.

3. Воспитательные

- формировать информационную культуру учащихся;
- воспитывать толерантное отношение в группе;
- добиться максимальной самостоятельности детского творчества;
- воспитывать умение планировать свою работу;
- сформировать интерес к профессиям, связанным с программированием.

Объем и сроки реализации: программа предназначена для учащихся 14-16 лет (8-9 классов). Объем – 34 часа, количество учебных часов в неделю – 1 час. Сроки реализации: в течение учебного года.

Прогнозируемый результат:

По окончании изучения данного курса прогнозируется, что учащиеся будут обладать следующими знаниями, умениями и навыками:

- знать место языка Python среди языков программирования высокого уровня,
- знать особенности структуры программы, представленной на языке Python,
- знать основные операторы языка Python, их синтаксис
- знать, что такое алгоритм, свойства и типы алгоритмов, способы записи алгоритмов;
- знать назначение вспомогательных алгоритмов, технологии построения простых и сложных алгоритмов: метод последовательной детализации и сборочный (библиотечный) метод;

• уметь составлять линейные, ветвящиеся и циклические алгоритмы управления в среде учебных исполнителей;

- уметь выделять подзадачи; определять и использовать вспомогательные алгоритмы;
- знать правила описания процедур в Python и построение вызова процедуры;
- решать различные задачи по программированию;
- иметь представление о таких структурах данных, как множество, запись, файл, стек, очередь, строка;

- знать, как формально определять в программе тип «массив»,
- знать свойства данных типа «массив»,
- создавать алгоритмы сортировки линейных числовых массивов и поиска в упорядоченном массиве;

• создавать программы и изображения в среде программирования Python.

Основные формы и методы работы с учащимися

Формы работы: индивидуальные (диагностические, групповые (практические работы с использованием цифровой лаборатории по физике ЦО Точка Роста», семинары), пленарные (мастер-класс, коллективное обсуждение, тренинг).

Активные методы обучения: моделирование процессов; наблюдение, анализ, обобщение; проблемно-поисковые ситуации, кейс технология.

Интерактивные методы: применением аудио- и видеоматериалов, ИКТ в режиме онлайн, работа с электронными учебниками, учебными сайтами.

Формы контроля

Осуществляется текущий, промежуточный и итоговый контроль. Текущий контроль осуществляется регулярно во время проведения каждого лабораторного занятия, заключается в ответе учащихся на контрольные вопросы, демонстрации полученных результатов, фронтальных опросов учителем. В Модулях 1 и 2 предусмотрены две промежуточные контрольные работы. Итоговая контрольная работа проводится после прохождения всех модулей.

Использование оборудования «Точка роста» при реализации данной образовательной программы позволяет создать условия:

- для расширения содержания школьного образования по информатике;
- для повышения познавательной активности обучающихся в технической области;
- для развития личности ребёнка в процессе обучения информатики, его способностей, формирования и удовлетворения социально значимых интересов и потребностей;
- для работы с одарёнными школьниками, организации их развития в различных областях образовательной, творческой деятельности.

Учебно-тематический план

| | всего | теории | практики | контроль |
|---|-------|--------|----------|----------|
| Модуль №1. Стартовый уровень. Раздел 1. Основы в Python | 6 | 1 | 4 | 1 |
| Модуль №2. Базовый уровень. Раздел 2. Алгоритмические структуры | 10 | 3 | 6 | 1 |
| Раздел 3. Подпрограммы | 4 | 1 | 2 | 1 |
| Модуль №3. Продвинутый уровень. Раздел 4. Массивы | 10 | 3 | 6 | 1 |
| Раздел 5. Использование графики в программировании | 4 | 0 | 2 | 2 |
| ИТОГО: | 34 | 8 | 20 | 6 |

Содержание изучаемого курса

| Модули | Содержание модуля | Средства ЦО «Точка Роста» |
|--|--|---|
| <p>Модуль №1. Стартовый уровень.</p> | <p>Раздел 1. Основы в Python (6 часов) Введение в Python. Структура программы на языке Python. Рекомендации по стилю записи программы, использование комментариев. Алфавит языка. Типы данных. Константы. Переменные. Организация ввода-вывода. Оператор присваивания. Общий вид программы на языке Python. Стандартные функции. Простейшая программа. Арифметические выражения. Правила записи арифметических выражений.</p> <p>Учащийся, освоив модуль № 1, должен знать</p> <ul style="list-style-type: none"> • алфавит языка • целый и вещественный, логический и символьный типы данных • структуру программы на языке Python <p>и уметь:</p> <ul style="list-style-type: none"> • организовывать ввод-вывод данных • использовать оператор присваивания • определять необходимый тип у различных данных • записывать математические выражения на языке Python • исправлять ошибки, допущенные при написании простейших программ | <p>Образовательный конструктор для практики блочного программирования с комплектом датчиков</p> <p>Среда программирования Python</p> <p>Среда программирования Scratch</p> <p>Компьютер, проектор, интерактивная доска.</p> |
| <p>Модуль №2, базовый уровень</p> | <p>Раздел 2. Алгоритмические структуры (10 часов) Организация программ разветвляющейся структуры. Условный оператор. Ветвление алгоритма на три рукава и более. Виды операторов цикла. Вложенные циклы.</p> <p>Раздел 3. Подпрограммы (4 часа) Подпрограммы (процедуры и функции), их описание и вызов в программе. Решение задач с математическим содержанием на использование подпрограмм. Файловые переменные. Ввод и вывод данных с использованием текстовых файлов.</p> | <p>Образовательный конструктор для практики блочного программирования с комплектом датчиков</p> <p>Среда программирования Python</p> <p>Среда программирования Scratch</p> |

| | | |
|---------------------------------------|---|---|
| | <p>Учащийся, освоив модуль № 2, должен знать</p> <ul style="list-style-type: none"> • Полное и неполное ветвление • Цикл с предусловием • Цикл с постусловием • Цикл с параметром • Понятия процедура и функция • Способы считывания данных из файлов <p>и уметь:</p> <ul style="list-style-type: none"> • Организовывать ветвления в программах на три рукава и более • Определять необходимый цикл в зависимости от поставленной задачи • Использовать вложенные циклы • Подключать файлы к программам • Определять необходимость использования функций и процедур | <p>Компьютер, проектор, интерактивная доска.</p> |
| <p>Модуль №3, продвинутый уровень</p> | <p>Раздел 4. Массивы (10 часов) Массивы (одномерные (линейные) и двумерные), различные способы их описания в программе. Обработка массивов (ввод и вывод элементов массива; поиск элементов в массиве; проведение математических операций с элементами массива; замена, удаление и вставка элементов в массиве; сортировка). Одномерные массивы: описание и ввод элементов, действия над ними. Поиск, замена в одномерном массиве. Сортировка массива. Способы сортировки. Понятие двумерного массива: описание и ввод элементов. Обработка элементов двумерных массивов. Сортировка массива. Способы сортировки.</p> <p>Раздел 5. Использование графики в программировании (4 часа) Графический режим. Примитивы в графическом режиме. Рисование с помощью примитивов.</p> <p>Учащийся, освоив модуль № 3, должен знать</p> <ul style="list-style-type: none"> • способы создания массивов • принципы сортировки элементов массива • типы поиска в массиве • понятие мерности элементов массива | <p>Образовательный конструктор для практики блочного программирования с комплектом датчиков Среда программирования Python Среда программирования Scratch Компьютер, проектор, интерактивная доска.</p> |

| | | |
|--|---|--|
| | <ul style="list-style-type: none"> • способы замены элементов массива • примитивы в графическом режиме Python <p>и уметь:</p> <ul style="list-style-type: none"> • использовать цикл ДЛЯ при создании массива и переборе элементов массива • выполнять поиск, замену, сортировку элементов массива • создавать многомерные массивы • помещать в массив из файла различные типы данных • включать графический режим • создавать изображения из графических примитивов | |
|--|---|--|

Методическое обеспечение

1. Приказ министерства просвещения российской федерации от 9 ноября 2018 г. № 196 "Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам".

2. Методическое пособие. Реализация образовательных программ по предмету "Информатика" с использованием оборудования центра «Точка роста» // Под редакцией С. Г. Григорьева. – М.: Министерство просвещения Российской Федерации, 2021. -180с.

3. Первин Ю. А. Методика раннего обучения информатике. – М.: «Бином», Лаборатория базовых знаний, 2008. – 228 с.

4. Практический Python 3 для начинающих <https://pythonworld.ru/samouchitel-python>.

5. Рафгарден Т. Совершенный алгоритм. Жадные алгоритмы и динамическое программирование. – СПб.: Питер, 2020. – 256 с.

6. Рейтц К., Шлюссер Т. Автостопом по Python. – СПб.: Питер, 2017. – 336 с.

Список литературы

1. Методическое пособие. Реализация образовательных программ по предмету "Информатика" с использованием оборудования центра «Точка роста» // Под редакцией С. Г. Григорьева. – М.: Министерство просвещения Российской Федерации, 2021. -180с.

2. Сайт ФИПИ. Открытый банк заданий для формирования естественно-научной грамотности: <https://fipi.ru/oge/otkrytyy-bank-zadaniy-oge>

3. Сайт Единая коллекция цифровых образовательных ресурсов: <http://school-collection.edu.ru/catalog>

4. Библиотека электронных материалов ФГИС «Моя школа» <https://myschool.edu.ru>

5. <https://scratch.mit.edu/> Сообщество Sctach.

6. Python для начинающих 2021 – уроки, задачи и тесты <https://pythonru.com/uroki/python-dlja-nachinajushhih>

7. Python/Учебник Python 3.1 https://ru.wikibooks.org/wiki/Python/%D0%A3%D1%87%D0%B5%D0%B1%D0%BD%D0%B8%D0%BA_Python_3.1

8. Буйначев С. К. Основы программирования на языке Python: учеб. пособие. – Екатеринбург: Изд-во Урал. ун-та, 2014. – 91 с.

9. Бхаргава А. Грокаем алгоритмы. Иллюстрированное пособие для программистов и любопытствующих. – СПб.: Питер, 2017. – 288 с.

10. Бэрри П. Изучаем программирование на Python. – М., 2017. – 624 с.

11. Голиков Д. В. Scratch для юных программистов. – СПб.: БХВ-Петербург, 2017. – 192 с.

12. Гэддис Т. Начинаем программировать на Python / Пер. с англ. – 4-е изд. – СПб.:БХВ-Петербург, 2019. – 768 с.

13. Луридад П. Алгоритмы для начинающих: теория и практика для разработчика. – М. : Эксмо, 2018. – 608 с.

14. Лутц М. Изучаем Python / Пер. с англ. – 3-е изд – СПб.: Символ Плюс, 2009. – 848 с.