

Центр цифрового образования детей «IT- куб»
(структурное подразделение АНО ДТ «Красноярский Кванториум»)

РЕКОМЕНДОВАНО
методическим советом

Протокол № 1
от «31» мая 2022 г.

УТВЕРЖДАЮ
Генеральный директор
АНО ДТ «Красноярский
Кванториум»


Кениг С.Р.

Приказ № 31
от «31» мая 2022 г.



Дополнительная общеобразовательная общеразвивающая программа
технической направленности

«Программирование на языке Python»

Срок реализации:

1 год

Возраст детей:

11-17 лет

Составитель программы:

Нагуслаев Н.Т.

г. Красноярск, 2022 г.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная общеобразовательная общеразвивающая программа «Программирование на языке Python» (далее - программа) имеет техническую направленность, базовый уровень сложности и ориентирована на обучающихся 11-17 лет. Программа в объеме 144 часа рассчитана на один год из расчета 4 часа в неделю.

Актуальность программы

В настоящее время компьютеры окружают современного человека повсюду, и не многие понимают то, как работают сервисы и программы, которыми мы пользуемся ежедневно. Программирование - это не просто написание программ для компьютера, это целое искусство воплощения идеи в свою программу. Знание основ программирования может помочь не только начинающим программистам продолжить свой путь в этой перспективной области, но и тем, чья жизнь в дальнейшем не будет тесно связана с написанием программ. Имея эти знания человек в дальнейшем будет способен автоматизировать решение рутинных задач и бытовых проблем, с которыми сталкивается в жизни.

Педагогическая целесообразность

Выбор своего первого языка программирования — это очень сложная задача и поэтому курс построен таким образом, чтобы помочь учащимся заняться программированием и понять в каком направлении им хотелось бы развиваться дальше. Изучая программирование, обучающиеся получают глубокое понимание принципов работы компьютера, организации ввода, вывода и хранения информации, принципов построения диалоговых приложений, познают азы профессии программиста.

В рамках образовательной программы «Программирование на языке Python» обучающийся познакомится с средой программирования PyCharm, затем он познакомится, пожалуй, с одним из самых универсальных языков программирования Python.

Цель и задачи программы

Целью программы «Программирование на языке Python» является изучение основ программирования на языке Python, рассмотрение различных парадигм программирования, предлагаемых этим языком (процедурная, функциональная, объектно-ориентированная); подготовка к использованию как языка программирования, так и методов программирования на Python в учебной и последующей профессиональной деятельности в различных предметных областях.

Задачи:

-сформировать и развить навыки алгоритмического и логического мышления, грамотной разработки программ;

-познакомить с принципами и методами функционального программирования;

-познакомить с принципами и методами объектно-ориентированного программирования;

-сформировать навыки работы в интегрированной среде разработки на языке Python;

-сформировать знания о конструкции языка программирования Python;

-познакомить с основными структурами данных и типовыми методами обработки этих структур;

-сформировать навыки разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;

-сформировать навыки поиска информации в сети интернет, анализа выбранной информации на соответствие запросу, использования информации при решении задач;

-развить у обучающихся интерес к программированию;

-сформировать навыки самостоятельности и творческого подхода к решению задач с использованием средств вычислительной техники;

-расширить кругозор обучающихся в области программирования.

Отличительные особенности данной дополнительной образовательной программы от существующих образовательных программ.

Отличительной особенностью данной образовательной программы является то, что процесс обучения выстроен с учетом используемых современных образовательных технологий. Программа предполагает использование и реализацию таких форм организации материала, которые допускают освоение специализированных знаний и языка программирования, обеспечивают трансляцию общей и целостной картины в рамках содержательно-тематического направления программ. Итоговое задание позволяет проявить творческий подход к решению задачи.

Возраст обучающихся, участвующих в программе

Программа «Программирование на языке Python» рассчитана на обучающихся 11-17 лет.

Условия вхождения в программу:

Набор на Программу осуществляется в соответствии с Правилами приема и отчисления обучающихся в АНО «Красноярский детский технопарк «Кванториум».

Срок реализации программы:

Программа рассчитана на 1 год обучения. Годовая нагрузка на обучающегося составляет 144 часа.

Режим занятий, формы и методы обучения

Учебные занятия проходят в очной форме. Режим занятий – 2 раза в неделю по 2 академических часа (1 академический час - 40 минут) с обязательным перерывом, что определяется Санитарно-эпидемиологическими правилами и нормативами СанПиН 2.4.4.3172-14.

При проведении занятий используются комбинированные занятия – изложение нового материала, проверка пройденного материала, закрепление полученных знаний, самостоятельная работа.

При проведении занятий используются три формы работы:

- демонстрационная, когда обучающиеся слушают объяснения педагога и наблюдают за демонстрационным экраном или экранами компьютеров на ученических рабочих местах;
- фронтальная, когда обучающиеся синхронно работают под управлением педагога;
- самостоятельная, когда обучающиеся выполняют индивидуальные задания в течение части занятия.

Повторение и усвоение пройденного материала осуществляется через прикладную работу обучающегося, использующего на практике приобретенные знания.

Ожидаемые образовательные результаты и эффекты, способы предъявления и отслеживания результатов

По результатам обучения, обучающиеся овладевают основами программирования на языке Python.

Ожидаемые результаты.

Личностные результаты:

- сформировано умение самостоятельной деятельности;
- сформировано умение работать в команде;
- сформированы навыки анализа и самоанализа.

Метапредметные результаты:

-сформировано умение самостоятельно планировать пути достижения целей, соотносить свои действия с планируемыми

результатами, осуществлять контроль своей деятельности, определять способы действий в рамках предложенных условий, корректировать свои действия в соответствии с изменяющейся ситуацией; оценивать правильность выполнения учебной задачи;

-сформировано умение самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности;

-сформированы навыки самостоятельности и творческого подхода к решению задач с использованием средств вычислительной техники;

-сформировано умение выступления с компьютерным сопровождением.

Предметные результаты:

-сформированы знания основ современных языков программирования, знания о принципах и методах функционального программирования, принципах и методах объектно-ориентированного программирования;

-развиты навыки алгоритмического и логического мышления, грамотной разработки программ

-расширен кругозор обучающихся в области программирования.

-развит интерес к программированию;

-сформировано умение объяснять и использовать на практике как простые, так и сложные структуры данных, конструкции для работы с ними и типовые методы обработки этих структур;

-сформированы навыки работы в интегрированной среде разработки на языке и знания о конструкции языка программирования Python;

-сформированы навыки разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;

-сформировано умение анализа, поиска и обработки как своего, так и чужого кода;

-сформировано умение работы с информацией: нахождение, оценки и использования информации из различных источников, необходимой для решения профессиональных задач (в том числе на основе системного подхода).

Механизм оценки результативности

По итогам каждого этапа проводится промежуточная аттестация в виде презентации полученных результатов и разработанных программ.

Итоговая аттестация проводится в конце года и представляет собой защиту итогового кейсового задания с использованием знаний, полученных за курс.

УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

№	Наименование разделов	Всего часов	Теория	Практика
1	Введение в программирование	18	7	11
1.1	Знакомство со средой	2	1	1
1.2	Условный оператор	2	1	1
1.3	Простые встроенные функции	2	1	1
1.4	Знакомство с циклом while	4	1	3
1.5	Отладчик	2	1	1
1.6	Знакомство с циклом for	4	1	3
1.7	True и False, break и continue	2	1	1
2	Базовые конструкции в Python	30	10	20
2.1	Вложенные циклы	4	1	3
2.2	Множества	2	1	1
2.3	Строки. Индексация	2	1	1
2.4	Строки. Срезы	2	1	1
2.5	Знакомство со списками	2	1	1
2.6	Кортежи. Преобразование коллекций	2	1	1
2.7	Методы split и join. Списочные выражения	4	1	3
2.8	Методы списков и строк	4	1	3
2.9	Вложенные списки	4	1	3
2.10	Знакомство со словарями	4	1	3
3	Решение прикладных задач в Python	38	19	19
3.1	Функции	4	2	2
3.2	Возвращение значений из функции	2	1	1
3.3	Области видимости переменных функции	2	1	1
3.4	Передача параметров функции	2	1	1
3.5	Функция с переменным числом аргументов	2	1	1
3.6	Функции как объект. Лямбда-функции	4	2	2
3.7	Рекурсия	4	2	2
3.8	Обработка коллекций. Поточковый ввод sys.stdin	4	2	2

3.9	Библиотеки Python. Встроенные модули	2	1	1
3.10	Библиотеки Python. Работа с графическими файлами	2	1	1
3.11	Библиотеки Python. Работа с графическими файлами и звуком	2	1	1
3.12	Библиотеки Python. Морфология	2	1	1
3.13	Библиотеки Python. Работа с документами	2	1	1
3.14	Библиотеки Python. Numpy	4	2	2
4	Объектно-ориентированное программирование	28	10	18
4.1	Введение в ООП.	4	2	2
4.2	Полиморфизм	4	2	2
4.3	Определение операторов	2	1	1
4.4	Наследование	4	2	2
4.5	Проектирование и разработка классов	4	0	4
4.6	Графический интерфейс пользователя с библиотекой tkinter	6	2	4
4.7	Работа с графикой с библиотекой tkinter	4	1	3
5	Выполнение индивидуального кейса	30	0	30
5.1	Выполнение индивидуального кейса	28	0	28
5.2	Итоговая аттестация	2		2
Итого		144	46	98

Содержание программы

1. Введение в программирование

1.1. Тема: Знакомство со средой

Теоретическая работа: программирование и программа, виды языков программирования.

Практическая работа: знакомство с виртуальной средой взаимодействия: регистрация, организация личного кабинета, поиск и выкладывание материалов. Знакомство с системой автоматизированной проверки задач.

1.2. Тема: Условный оператор

Теоретическая работа: конструкция ветвления. Разветвляющиеся программы. Условные выражения. Операции отношения. Логические операции. Полная и сокращенная форма условного оператора. Блоки и составные операторы. Оператор множественного выбора.

Практическая работа: работа с логическим типом данных. Объявление и использование в программе переменных, констант и выражений логического типа. Использование простых и составных условий. Полная и сокращенная формы условного

оператора. Создание разветвляющейся программы с помощью условных операторов. Оператор множественного выбора. Организация множественного разветвления в программе.

1.3. Тема: Простые встроенные функции

Теоретическая работа: простые типы данных: целый, вещественный, символьный, логический и их представление в ПК. Приведение типов. Работа с переменными и константами. Использование математических функций. Организация ввода и вывода данных.

Практическая работа: работа с простыми типами данных: целым, вещественным, символьным и логическим. Организация ввода и вывода данных.

1.4. Тема: Знакомство с циклом while

Теоретическая работа: циклы. Циклические программы. Понятия итерации, счетчика и параметров цикла. Циклов со счетчиком. Циклы с предусловием.

Практическая работа: циклы. повтор команд. Переменные цикла. Операторы while.

1.5. Тема: Отладчик

Теоретическая работа: отладка программы. Шаг вперед и назад. Вход в функции.

Практическая работа: разбор ошибок в программе. Первые шаги к тестированию.

1.6. Тема: Знакомство с циклом for

Теоретическая работа: синтаксис, реализация и способы применения всех видов циклов.

Практическая работа: циклы. Повтор команд. Переменные цикла. Оператор for.

1.6. Тема: True и False, break и continue

Теоретическая работа: операторы принудительного завершения итерации цикла. Оператор завершения цикла. Оператор безусловного перехода. Синтаксис, реализация и правила использования.

Практическая работа: бесконечный цикл. Выход из цикла.

2. Базовые конструкции в Python

2.1. Тема: Вложенные циклы

Теоретическая работа: циклы. Циклические программы. Понятия итерации, счетчика и параметров цикла. Циклов со счетчиком. Циклы с предусловием. Циклы с постусловием. Синтаксис, реализация и способы применения всех видов циклов. Вложенные циклы. Операторы принудительного завершения итерации цикла. Оператор завершения цикла. Оператор безусловного перехода. Синтаксис, реализация и правила использования.

Практическая работа: циклы. Повтор команд. Переменные цикла. Вложенные циклы. Операторы while, for. Бесконечный цикл. Выход из цикла.

2.2. Тема: Множества

Теоретическая работа: основы множеств и алгебры логики. Круги Эйлера.

Практическая работа: создание простейших множеств и основные встроенные функции. Как объединять множества и делать выборку.

2.3. Тема: Строки. Индексация.

Теоретическая работа: новые типы данных. Для чего они нужны и как с ними работать.

Практическая работа: работа типами данных: Строки, списки, кортежи и словари. Организация ввода и вывода данных.

2.4. Тема: Строки. Срезы

Теоретическая работа: новые типы данных, срезы. Обратная индексация.

Практическая работа: работа типами данных: Строки, срезы. Организация ввода и вывода данных.

2.5. Тема: Знакомство со списками

Теоретическая работа: списки. Индексация и обратная индексация, схожесть со строками.

Практическая работа: работа типами данных: Списки. Организация ввода и вывода данных.

2.6.Тема: Кортежи. Преобразование коллекций

Теоретическая работа: отличия кортежей от списков. Практическое использование списков.

Практическая работа: работа с типами данных. Кортежи, tuple(). Организация ввода и вывода данных.

2.7.Тема: Методы split и join. Списочные выражения

Теоретическая работа: преобразование данных.

Практическая работа: работа со списком, преобразование данных.

2.8.Тема: Методы списков и строк

Теоретическая работа: методы и функции для разных типов данных. Различия между изменяемыми и неизменяемыми типами данных.

Практическая работа: работа со списком, преобразование данных. Сортировка, поиск максимального и минимального элемента. Создание выборок.

2.9.Тема: Вложенные списки

Теоретическая работа: создание структуры в структуре.

Практическая работа: создание списка в списке, кортежа в списке. Правильный вызов и вывод.

2.10.Тема: Знакомство со словарями

Теоретическая работа: отличия словарей от списков, кортежей и множеств.

Практическая работа: создание словарей, использование словарей для создания базы данных.

3. Решение прикладных задач в Python

3.1.Тема: Функции

Теоретическая работа: подходы к программированию, деление программы на функции.

Практическая работа: создание функций. Аргументы. Правильный вывод и ввод данных.

3.2.Тема: Возвращение значений из функции

Теоретическая работа: вывод из функции. Аргументы.

Практическая работа: функция return. Присваивание функций.

3.3.Тема: Области видимости переменных функции

Теоретическая работа: отличия глобальных и локальных данных.

Практическая работа: создание глобальных и локальных переменных и структур данных. Вызов глобальных переменных и ограничения локальных.

3.4.Тема: Передача параметров функции

Теоретическая работа: ввод данных через аргументы и их отработка.

Практическая работа: работа с данным как локальными, так и глобальными. Передача данных между функций.

3.5.Тема: Функция с переменным числом аргументов

Теоретическая работа: создание функций с переменными аргументами.

Практическая работа: разница между статичными и переменными функциями.

3.6.Тема: Функции как объект. Лямбда-функции

Теоретическая работа: лямбда-функции.

Практическая работа: создание и использование функции. Работа с некорректным вызовом.

3.7.Тема: Рекурсия

Теоретическая работа: создание рекурсий. Принцип бесконечности.

Практическая работа: определение условий окончания рекурсий. Контроль бесконечности.

3.8. Тема: Обработка коллекций. Поточковый ввод sys.stdin

Теоретическая работа: потоковый ввод-вывод, классификация, общие подходы и методы, конвертация

Практическая работа: обобщение типов данных. Функция Counter(). Общие подходы к работе с любой коллекцией

3.9.Тема: Библиотеки Python. Встроенные модули

Теоретическая работа: виды библиотек.

Практическая работа: установка и подключение библиотек. Вызов функций из модулей и подмодулей.

3.10.Тема: Библиотеки Python. Работа с графическими файлами

Теоретическая работа: особенности работы с графическими файлами.

Практическая работа: библиотеки по работе с данными и графическими файлами. Обработка потокового изображения.

3.11.Тема: Библиотеки Python. Работа с графическими файлами и звуком

Теоретическая работа: обработка звуковых файлов. Особенности и сложности.

Практическая работа: работа с библиотеками подключения звука. Создание будильника.

3.12.Тема: Библиотеки Python. Морфология

Теоретическая работа: работа с текстовыми данными. Семантика и парсинг документов.

Практическая работа: считывание данных из файлов и запись. Редактирование и парсинг.

3.13.Тема: Библиотеки Python. Работа с документами

Теоретическая работа: работа с текстовыми данными. Семантика и парсинг документов.

Практическая работа: считывание данных из файлов и запись. Редактирование и парсинг.

3.14.Тема: Библиотеки Python. Numpy

Теоретическая работа: анализ данных и обработка больших данных.

Практическая работа: обработка больших данных. Особенности библиотеки NumPy. Разница между списками и структурами в NumPy.

4. Объектно-ориентированное программирование

4.1.Тема: Введение в ООП.

Теоретическая работа: понятия объекта и класса. Поля и методы класса. Иерархия классов. Принципы инкапсуляции, наследования и полиморфизма. Преимущества и недостатки объектно-ориентированного подхода. Этапы объектно-ориентированного программирования.

Практическая работа: создание классов, объектов, полей и методов.

4.2.Тема: Полиморфизм

Теоретическая работа: первый принцип ООП.

Практическая работа: изучение синтаксиса описания класса, его компонентов и спецификаторов доступа к ним. Изучение правил описания и вызова конструкторов, деструкторов и других методов класса. Объявление объектов и работа с ними.

Создание программы, выполняющей работу с классами и объектами.

4.3.Тема: Определение операторов

Теоретическая работа: свертка. Сильные стороны сверточных нейронных сетей в задачах компьютерного зрения.

Практическая работа: нейросетевая библиотека Keras. Классификация изображения.

4.4. Тема: Наследование

Теоретическая работа: второй принцип ООП.

Практическая работа: изучение синтаксиса описания класса, его компонентов и спецификаторов доступа к ним. Изучение правил описания и вызова конструкторов, деструкторов и других методов класса. Объявление объектов и работа с ними. Создание программы, выполняющей работу с классами и объектами.

4.5. Тема: Проектирование и разработка классов

Практическая работа: изучение синтаксиса описания класса, его компонентов и спецификаторов доступа к ним. Изучение правил описания и вызова конструкторов, деструкторов и других методов класса. Объявление объектов и работа с ними. Создание программы, выполняющей работу с классами и объектами

4.6. Тема: Графический интерфейс пользователя с библиотекой tkinter

Теоретическая работа: понятия GUI. Модуль Tkinter и библиотека PyQt.

Практическая работа: создание своего первого графического интерфейса. Создание виджета Label. Настройка размера и шрифта текста. Настройка размеров окна приложения. Добавление виджета Button.

4.7. Тема: Работа с графикой с библиотекой tkinter

Теоретическая работа: модуль Tkinter и библиотека PyQt. Основы графики.

Практическая работа: создание крестиков-ноликов. Отрисовка изображений.

5. Выполнение индивидуального кейса

5.1 Выполнение индивидуального кейса

Практическая работа: выполнение индивидуального кейса.

5.2 Итоговая аттестация.

Практическая работа: защита решения индивидуального кейса.

МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА, НЕОБХОДИМАЯ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ПРОГРАММЕ

№ п/п	Наименование оборудования (ФПО)	Примерная модель (РВПО)	Единица измерения	Количество
1	"Презентационное оборудование"			
1.1	Напольная мобильная стойка для интерактивных досок или универсальное настенное крепление	ONKRON TS1330	шт	1
1.2	Моноблочное интерактивное устройство	SMART SBID-MX265-V2	шт	1
2	"Дополнительное оборудование"			
2.1	Комплект кабелей и переходников	Atcom High speed HDMI - HDMI MOST Lite LRG FAZA FOP-05GS-500	шт	4
2.2	Флипчарт магнитно-маркерный на треноге	Attache	шт	1
3	"Профильное оборудование"			
3.1	WEB-камера (рабочее место педагога)	logitech C922	шт	1
3.2	Наушники (рабочее место обучающегося)	SVEN AP-320M/321M	шт	12
3.3	МФУ (принтер, сканер, копир) (рабочее место педагога)	Epson L14150	шт	1
3.4	Ноутбук тип 2 (рабочее место обучающегося)	Acer TravelMate P2 TMP215-52	шт	12
3.5	Ноутбук тип 1 (рабочее место педагога)	Acer TravelMate P2 TMP215-52	шт	1

СПИСОК ИСТОЧНИКОВ

1. Босова, Л.Л. Занимательные задачи по информатике [Электронный ресурс] / Л.Л. Босова, А.Ю. Босова, Ю.Г. Коломенская. – 5-е изд. – М.: БИНОМ. Лаборатория знаний, 2013.
2. Гаско, Рик. Простой Python просто с нуля [Электронный ресурс] / Рик Гаско. – М.: СОЛОН-Пресс, 2019.
3. Златопольский, Д.М. Программирование: типовые задачи, алгоритмы, методы [Электронный ресурс] / Д.М. Златопольский. – М.: БИНОМ. Лаборатория знаний, 2007.
4. Златопольский, Д. М. Сборник задач по программированию [Электронный ресурс] / Д.М. Златопольский. – 3-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2011.
5. Меньшиков, Ф.В. Олимпиадные задачи по программированию [Электронный ресурс] / Ф.В. Меньшиков. – СПб.: Питер, 2006.
6. Мюллер, Джон Пол. Python для чайников [Электронный ресурс] / Джон Пол Мюллер. - 2-е изд.: Пер с англ. – СПб.: ООО «Диалектика», 2019.
7. Окулов, С.М. Основы программирования: учебное пособие [Электронный ресурс] / С.М. Окулов. – 10-е изд., электрон. – Москва : Лаборатория знаний, 2020.
8. Прохоренок, Н.А. Python 3. Самое необходимое [Электронный ресурс] / Н. А. Прохоренок, В. А. Дронов. – 2-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2019.
9. Пупышев, В.В. 128 задач по началам программирования [Электронный ресурс] / В.В. Пупышев. – М.: БИНОМ. Лаборатория знаний, 2009.
10. Пэйн, Брайсон. Python для детей и родителей [Электронный ресурс] / Брайсон Пейн [пер. с англ. М.А. Райтмана]. - М.: Издательство «Э», 2017.
11. Свейгарт, Эл. Учим Python, делая крутые игры [Электронный ресурс] / Эл. Свейгарт. - М: Эксмо, 2018.
12. Седер, Наоми. Python. Экспресс-курс [Электронный ресурс] / Наоми Седер. - 3-е изд. – СПб.: Питер, 2019.
13. Столяров, А.В. Оформление программного кода [Электронный ресурс] / А.В. Столяров. – 2-е изд., испр. и доп. – Москва: МАКС Пресс, 2019.
14. Шень, А. Программирование: теоремы и задачи [Электронный ресурс] / А. Шень. - 6-е изд., дополненное. М.: МЦНМО, 2017.
15. Шуман, Х.Г. Python для детей [Электронный ресурс] / Х.Г. Шуман [пер. с нем. М.А. Райтман]. – М.: ДМК Пресс, 2019.

