

**Федеральное государственное автономное образовательное учреждение
высшего образования «Национальный исследовательский
технологический университет «МИСиС»**



УТВЕРЖДАЮ

Проректор по образованию

А.А. Волков

«*сентябрь*» 2022 г.

Дополнительная общеобразовательная общеразвивающая программа

«Алгоритмизация и программирование. (Python)»

НАПРАВЛЕННОСТЬ: ТЕХНИЧЕСКАЯ

Уровень: вводный

Возраст обучающихся 14 - 18 лет

Срок реализации: 24 академических часа

Составитель (разработчик):

О.И. Ремизова

к.ф.-м. н, доцент кафедры «Инженерная кибернетика»

А.И. Широков

доцент кафедры «Инженерная кибернетика»

Москва

2022

1. Пояснительная записка

Дополнительная общеобразовательная (общеразвивающая) программа «Алгоритмизация и программирование (Python)» (далее – Программа) является вводной образовательной программой для школьников ориентированных на развитие своих навыков. Актуальность программы определяется динамичным развитием Информационных технологий и большой востребованностью специалистов в этой области. Программа является дополнением к школьному курсу информатики, развивающей знания и навыки в области программирования и алгоритмизации, а так же кругозор в вопросах командной разработки. Знание основ программирования позволит слушателям курса сформировать для себя требования к современному разработчику программного обеспечения. Программа реализуется Федеральным государственным автономным образовательным учреждением высшего образования «Национальный исследовательский технологический университет «МИСиС» (далее – НИТУ «МИСиС», Университет). Программа разработана и утверждена в соответствии с Уставом НИТУ «МИСиС» с целью формирования у школьников позитивного восприятия инженерных специальностей.

Программа имеет **техническую направленность**.

Уровень освоения – вводный. Программа предполагает в познавательной форме и на понятном языке пояснить новые синтаксические конструкции языка программирования Python, развить навык реализации эффективных алгоритмов.

Новизна Программы состоит в применении современных средств программирования.

Актуальность. Современный уровень развития информационных технологий предполагает освоение новых методов разработки программ. Содержание обучения ориентировано на пробуждение интереса учащихся к алгоритмизации, а также к существующим задачам программирования.

Педагогическая целесообразность. Обучение данному курсу направлено на выработку способностей к творческой деятельности участников, практическое применение полученных навыков и знаний в области программирования, на развитие научно-технического способа мышления обучающимися.

Цель и задачи программы

Цель - формирование и развитие у обучающихся интеллектуальных и практических компетенций в области разработки программ на языке Python, их дальнейшему практическому применению в проектной деятельности.

Развитие интереса к современным технологиям. Это предполагает формирование у школьников навыка самостоятельного формулирования методов решения поставленных задач, выбор методов их реализации, практике отладки программ.

Среди **задач** решаемых при реализации Программы следует выделить **обучающие, развивающие и воспитательные задачи**.

Каждый обучающийся расширит знания о языке программирования Python, разберется в особенностях синтаксических и алгоритмических конструкций языка, получит представление о современных методах тестирования программ (**обучающие задачи**);

при этом предполагается практическое применение знания современных технологий программирования (**развивающие задачи**);

формирование практики командной работы, творческого отношения к задачам, необходимым для реализации проекта (**воспитательные задачи**).

Отличительная особенность Программы заключается в том, что она позволяет обучающимся в короткие сроки познакомиться с задачами, которые развивают алгоритмическое

мышление и знание конструкций языка программирования Python.

Возраст обучающихся: 14-18 лет.

Сроки реализации: 24 академических часов.

Наполняемость группы: 20-25 человек.

Формы и режим занятий

Режим занятий: 1 занятие в неделю по 3 академических часа.

Формы организации деятельности: групповые, индивидуально-групповые.

Методы обучения: словесные, комбинированные, теоретические, практические.

Ожидаемые результаты и способы их определения

В результате освоения программы обучающиеся
будут знать:

- основные понятия информатики;
- основы программирования;
- алгоритмы и структуры данных;

будут уметь:

- разрабатывать и отлаживать нестандартные алгоритмы;
- подбирать наиболее подходящие методы и средства реализации алгоритмов;
- работать в команде и принимать решения;
- защищать подготовленный за время обучения проект.

2. Учебно-тематический план

№ п/ п	Раздел / Тема	Количество часов			Форма аттестации/ контроля
		Всего	Теория	Практические занятия	
1	Блок 1. Ввод – вывод данных				
1.1	Введение. Знакомство с Python. Команды print и input. Параметры sep и end	4	1	3	Практическая работа
2	Блок 2. Типы данных				
2.1	Числовые типы данных: int, float. Целочисленная арифметика. Модуль math. Строковый тип данных	3		3	Практическая работа
3	Блок 3. Условный оператор				
3.1	Выбор из двух. Логические операции. Вложенные и каскадные условия	2		2	Практическая работа
4	Блок 4. Циклы for и while				
4.1	Цикл for. Функция range. Частые сценарии. Цикл while. Обработка цифр числа. Break, continue, else. Поиск ошибок и ревью кода. Вложенные циклы	2		2	Практическая работа
5	Блок 5. Строковый тип данных				
5.1	Индексация. Срезы. Методы строк. Строки в памяти компьютера. Таблица символов Unicode	3	1	2	Практическая работа
6	Блок 6. Списки				

6.1	Введение в списки. Основы работы со списками. Методы списков. Вывод элементов списка. Методы строк: <code>split</code> , <code>join</code> . Списочные выражения. Сортировка списков	4	1	3	Практическая работа
7	Блок 7. Функции				
7.1	Функции без параметров. Функции с параметрами. Локальные и глобальные переменные. Функции с возвратом значения	3	-	3	Практическая работа
8	Блок 8. Работа над мини – проектом	3	-	3	Практическая работа
Итоговая аттестация проводится на основании совокупности выполненных промежуточных практических работ					
Всего		24	3	21	

3. Содержание образовательной программы

Блок 1. Ввод – вывод данных

1.1. Введение. Знакомство с Python. Команды `print` и `input`. Параметры `sep` и `end`

Теория (1 а.ч.) Python. Команды `print` и `input`. Параметры `sep` и `end`

Практика (3 а.ч.) Разработка алгоритмов с использованием команд `print` и `input`

Планируемые результаты: слушатели научатся работать с командами `print` и `input`; смогут улучшать визуальный результат работы кода, используя параметры `sep` и `end`; получают навыки работы с целочисленной арифметикой.

Блок 2. Типы данных

2.1 Числовые типы данных: `int`, `float`. Целочисленная арифметика. Модуль `math`. Строковый тип данных

Практика (3 а.ч.) Разработка алгоритмов с использованием модуля `math` и числовых и строковых типов данных.

Планируемые результаты: слушатели познакомятся с числовыми и типами данных; научатся определять различие между `int` и `float`; получают навыки работы с модулем `math`.

Блок 3. Условный оператор

3.1 Выбор из двух. Логические операции. Вложенные и каскадные условия

Практика (2 а.ч.) Разработка алгоритмов с использованием логических операций

Планируемые результаты: слушатели научатся осуществлять выбор из двух предложенных вариантов; научатся использовать логические операции; получают навыки работы с вложенными и каскадными условиями.

Блок 4. Циклы `for` и `while`.

4.1 Циклы `for`. Функция `range`. Частые сценарии. Цикл `while`. Обработка цифр числа. `Break`, `continue`, `else`. Поиск ошибок и ревью кода. Вложенные циклы

Практика (2 а.ч.) Разработка алгоритмов с использованием циклов разной структуры.

Планируемые результаты: слушатели познакомятся с циклами `for` и `while`; получают навыки работы с функцией `range`, вложенными циклами; научатся обрабатывать цифры числа, искать ошибки и получать ревью кода; использовать операторы `break`, `continue`, `else`.

Блок 5. Строковый тип данных

5.1 Индексация. Срезы. Методы строк. Строки в памяти компьютера. Таблица символов Unicode

Теория (1а.ч.) Индексация. Срезы. Методы строк. Строки в памяти компьютера. Таблица символов Unicode.

Практика (2а.ч.) Разработка алгоритмов с использованием строковых функций и таблицы символов Unicode.

Планируемые результаты: слушатели познакомятся с индексацией и срезами; получат навыки работы с функциями строк; получат информацию о специфике хранения строк в памяти компьютера и таблица символов Unicode.

Блок 6. Списки.

6.1 Введение в списки. Основы работы со списками. Методы списков. Вывод элементов списка. Методы строк: split, join. Списочные выражения. Сортировка списков

Теория (1а.ч.) Списки и основные принципы работы со списками. Списочные выражения. Сортировка списков.

Практика (3а.ч.) Разработка алгоритмов с использованием списков и списочных выражений.

Планируемые результаты: слушатели получают основные навыки работы со списками; познакомятся с методами списков; научатся осуществлять вывод элементов списка; научатся использовать функции строк: split, join; получат навыки работы со списочными выражениями; научатся осуществлять сортировку списков.

Блок 7. Функции.

7.1 Функции без параметров. Функции с параметрами. Локальные и глобальные переменные. Функции с возвратом значения

Практика (3а.ч.) Разработка алгоритмов с использованием переменных, функций с параметрами и без.

Планируемые результаты: слушатели познакомятся с локальными и глобальными переменными; получат основные навыки работы с функциями без параметров, функциями с параметрами; функциями с возвратом значения.

Блок 8. Работа над мини – проектом.

Практика (3а.ч.) Разработка мини-приложения по выбранной теме.

Планируемые результаты: слушатели создадут свое мини – приложение по одной из заявленных тем: «Числовая угадайка»; «Магический шар»; «Генератор безопасных паролей»; «Шифр Цезаря»; «Генератор безопасных паролей»; «Угадайка слов».

4. ФОРМЫ АТТЕСТАЦИИ И КОНТРОЛЯ

Виды контроля. В образовательном процессе используются следующие методы контроля усвоения учащимися учебного материала:

Текущий контроль. Проводится с целью непрерывного отслеживания уровня усвоения материала и стимулирования учащихся не отвлекаться. Для реализации текущего контроля в процессе теоретического материала педагог обращается к учащимся с вопросами и дает короткие задания, в процессе практических занятий по итогам каждой темы обучающийся выполняет лабораторную работу.

Форма итоговой аттестации – итоговая аттестация проводится на основании совокупности выполненных промежуточных практических работ

Оценивание: зачтено/не зачтено.

5. Организационно-педагогические условия реализации программы

Методическое обеспечение программы

Методы обучения, используемые в программе: словесные (устное объяснение материала), наглядные (презентация), практические (демонстрационное выполнение исследовательской лабораторной работы); аналитические.

С целью стимулирования творческой активности учащихся будут использованы: игровые методики;

- метод проектов;
- метод погружения;
- методы сбора и обработки данных;
- исследовательский и проблемный методы;
- анализ справочных и литературных источников;
- поисковый эксперимент;
- опытная работа;
- обобщение результатов.

Виды дидактических материалов

Для обеспечения наглядности и доступности изучаемого материала будут использоваться:

- наглядные пособия смешанного типа (слайды, видеозаписи, эскизы);
- дидактические пособия (карточки с заданиями, раздаточный материал). Занятия будут проходить в форме лекций-бесед с демонстрацией преподавателем презентаций, а также в форме мастерских, на которых обучающиеся смогут реализовать на практике теоретические знания.

Организационно-педагогические ресурсы программы

Материально-техническое обеспечение учебной программы

Оборудование:

Наименование	На группу, ед.	Примечание
Персональный компьютер или ноутбук	По количеству обучающихся	ОС не ниже Windows 7, необходим доступ к сети Интернет скорость не ниже 50 Мбит/с Процессор 64-разрядный Примерно 2 ГБ свободного пространства на диске Память: 3ГБ ОЗУ, желательно выше
Проектор	1	
Экран	1	Для проектора

Кадровое обеспечение программы

Реализаторы программы:

Ремизова Ольга Игоревна, доцент кафедры инженерной кибернетики;

Широков Андрей Игоревич, доцент кафедры инженерной кибернетики.

6. Список литературы

а) Используемый при написании программы

1. Мусин Дмитрий. Самоучитель Python. Выпуск 0.2. pythonworld. 154 с.
2. Любанович Билл. Простой Python. Современный стиль программирования. - СПб.: Питер, 2016 - 480 с.
3. Программирование для «нормальных» с нуля на языке Python. В двух частях. Часть 1. Москва. Базальт СПО МКС Пресс. 2018. 180 с.
4. Брудно А. Л., Каплан Л. И. Олимпиады по программированию для школьников. — М.: Наука, 1985. — 96 с.

5. Дагене В. А., Григас Г. К., Аугутис К. Ф. 100 задач по программированию. — М.: Просвещение, 1993. — 255 с.

б) Рекомендованный обучающимся для успешного освоения программы

1. Джейсон Бриггс. Python для детей. Самоучитель по программированию. Иванов и Фабер. 2017. - 320 с.
2. Ханс-Георг Шуман. Python для детей. М. ДМК-Пресс. 2019, Майк МакГрат, 2016. 344 с.
3. Кирюхин В. М., Лапунов А. В., Окулов С. М. Задачи по информатике. Международные олимпиады. — М.: АБФ, 1996. — 272 с.

Приложение. Контрольно-измерительные материалы.

Примеры задач, решаемых на практических занятиях.

1. Вычислить $s = 3 + 3^2 + 3^3 + \dots + 3^8$.
2. Вычислить частное p и остаток q от деления двух натуральных чисел r и t , не используя операцию деления. Число r можно представить в виде $r = pt + q$. Будем последовательно вычитать t из r и подсчитывать количество вычитаний в переменной p . Значение q – результат последнего
3. Сформировать в цикле для заданных параметров такую фигуру. Задается количество звездочек в каждой строке и количество строк. Пример для 3 звездочек в строке и 4 строк.

4. Дано натуральное число. Определить, будет ли это число: нечётным, кратным 7.
5. Имеется коробка со сторонами: $A \times B \times C$. Определить, пройдет ли она в дверь с размерами $M \times K$.
6. Найти все углы треугольника со сторонами a , b и c . Предусмотреть в программе перевод радианной меры угла в градусы, минуты и секунды.
7. В компьютер вводятся координаты точек (x и y), лежащих на координатной плоскости. После ввода очередной точки программа спрашивает: «Будете вводить еще?». Если далее вводится 1 – ввод закончен, в противном случае обрабатываются данные еще одной точки. После ввода координат каждой точки выводить номер квадранта, в котором она находится. Определить количество точек, лежащих по отдельности в 2-м и 3-м квадрантах.
8. Для заданной символьной строки подсчитать сколько символов имеют код – четное число.
9. Программе задается несколько целых чисел. Для каждого из них определить, может ли он быть номером квартиры в 9-этажном доме, в котором на первом этаже 2 две квартиры, а на всех остальных – по четыре. Алгоритм вычисления номера квартиры оформить функцией.
10. Программе задается несколько вещественных чисел и два числа (a и b). Определить, смогут ли задаваемые числа a и b являться длинами треугольника.

