

Муниципальное бюджетное общеобразовательное учреждение
города Новосибирска «Инженерный лицей Новосибирского государственного
технического университета»

Рекомендовано решением
педагогического совета МБОУ
«Инженерный лицей НГТУ»
Протокол № 1
от «24» 08 2019

Утверждаю
Директор МБОУ
«Инженерный лицей НГТУ»
Безлепкина М.А.
Приказ № 25
от «24» 08 2019



**Рабочая программа спецкурса
Основы программирования на языке Python**

название учебного предмета, курса, дисциплины (модуля)

для класса(ов) 8

Количество часов:

Всего 36

в 8 класс 36

в неделю 1

Разработчик программы Гурина Анна Анатольевна, учитель информатики первой
квалификационной категории

(Ф.И.О. разработчика программы, занимаемая должность, квалификационная категория)

г. Новосибирск
2019

Программа обсуждалась на заседании кафедры/методического объединения учителей Математики и информатики МБОУ «Инженерный лицей НГТУ»

Протокол заседания № 1 от «22 августа 2019 г.

Подольский /руководитель МО Подольский Е.В.

Пояснительная записка

Изучение основ программирования связано с целым рядом умений и навыков (организация деятельности, ее планирование). Большое значение при изучении программирования имеет развитие алгоритмического мышления школьников и формирование многих приемов умственной деятельности. Недостаточные знания различных языков программирования затрудняют решение задач по проектированию и прогнозированию. В связи с инженерной направленностью лица решение такого рода задач является актуальным.

Данный элективный курс способствует совершенствованию и развитию важнейших знаний и умений в области информатики, помогает оценить обучающемуся свои возможности при изучении информатики и более осознанно выбрать профиль дальнейшего обучения. Курс «Основы программирования на языке Python» является логическим продолжением спецкурса 7 класса «Программирование от Scratch до Android»

Цели курса:

- формирование углубленного интереса к программированию;
- создание основы для дальнейшего изучения языков программирования на углубленном уровне;
- оказание помощи в осознанном выборе модели дальнейшего профессионального образования.

Задачи курса:

- понять значение алгоритмизации как метода познания окружающего мира, принципы структурной алгоритмизации;
- развитие интереса учащихся к изучению программирования;
- овладеть базовыми понятиями теории алгоритмов при решении математических задач;
- научиться разрабатывать эффективные алгоритмы и программы для числовых рядов, прогрессий, значений многочленов, массивов, в области арифметики рациональных чисел;
- приобретение навыков работы в системе программирования Python;
- формирование самостоятельности и творческого подхода к решению задач с помощью средств современной вычислительной техники.
- формирование навыков алгоритмического и логического мышления.

Рабочая программа разработана на основе следующих нормативно-правовых документов:

- Федеральный закон от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации» в редакции от 29.07.2017 года.
- Приказ Министерства образования и науки РФ от 17 декабря 2010 г. N 1897– "Об утверждении федерального государственного образовательного стандарта основного общего образования" (с изменениями и дополнениями от: 29 декабря 2014 г., 31 декабря 2015 г.).
- Примерная основная образовательная программа основного общего образования. (Одобрена решением федерального учебно-методического объединения по общему образованию (протокол № 1/15 от 8 апреля 2015 г.). Основная образовательная программа среднего (общего) образования МБОУ – «Инженерный лицей НГТУ».
- Учебный план МБОУ «Инженерный лицей НГТУ»

Учебно-методическое и материально-техническое обеспечение образовательного процесса

- «Программируем на Python» (автор: М. Доусон; СПб.: Питер, 2014г.),
- «Python 3. Самое необходимое» (авторы: Н. Прохоренко , В. Дронов; БХВ-Петербург, 2016г.).

Технические средства

- Операционная система Windows 7
- Пакет офисных приложений MS Office 2007, OpenOffice, среда программирования Python.
- Персональные компьютеры для учащихся
- ПК учителя
- Интерактивная доска
- Сканер
- Принтер
- Сеть Интернет
- Проектор

Электронные образовательные ресурсы

Наименование образовательного ресурса	Электронный адрес
Ресурсы Единой коллекции цифровых образовательных ресурсов	http://school-collection.edu.ru/
СДО лицея	http://lyceum.nstu.ru/sdo/

Содержание учебного курса

Знакомство с языком Python

Общие сведения о языке Python. Установка Python на компьютер. Режимы работы Python. Что такое программа. Первая программа. Структура программы на языке Python. Комментарии.

Типы данных. Преобразование типов. Переменные. Оператор присваивания. Имена переменных и ключевые слова. Выражения. Операции. Порядок выполнения операций. Математические функции. Композиция. Ввод и вывод. Ввод данных с клавиатуры. Вывод данных на экран. Пример скрипта, использующего ввод и вывод данных. Задачи на элементарные действия с числами. Решение задач на элементарные действия с числами

Условные предложения

Логический тип данных. Логические выражения и операторы. Сложные условные выражения (логические операции and, or, not). Условный оператор. Альтернативное выполнение. Примеры решения задач с условным оператором. Множественное ветвление. Реализация ветвления в языке Python.

Циклы

Понятие цикла. Тело цикла. Условия выполнения тела цикла. Оператор цикла с условием. Оператор цикла while. Бесконечные циклы. Альтернативная ветка цикла while. Обновление переменной. Краткая форма записи обновления. Примеры использования циклов.

Оператор цикла с параметром for. Операторы управления циклом. Пример задачи с использованием цикла for. Вложенные циклы. Циклы в циклах. Случайные числа. Функция randrange. Функция random. Примеры решения задач с циклом.

Функции

Создание функций. Параметры и аргументы. Локальные и глобальные переменные.

Поток выполнения. Функции, возвращающие результат. Анонимные функции, инструкция lambda. Примеры решения задач с использованием функций.

Рекурсивные функции. Вычисление факториала. Числа Фибоначчи.

Строки - последовательности символов

Составной тип данных - строка. Доступ по индексу. Длина строки и отрицательные индексы. Преобразование типов. Применение цикла для обхода строки.

Срезы строк. Строки нельзя изменить. Сравнение строк. Оператор in. Модуль string. Операторы для всех типов последовательностей (строки, списки, кортежи). Примеры решения задач со строками.

Сложные типы данных

Списки. Тип список (list). Индексы. Обход списка. Проверка вхождения в список. Добавление в список. Суммирование или изменение списка. Операторы для списков. Срезы списков. Удаление списка. Клонирование списков. Списочные параметры. Функция range. Списки: примеры решения задач.

Матрицы. Вложенные списки. Матрицы. Строки и списки. Генераторы списков в Python.

Кортежи. Присваивание кортежей. Кортежи как возвращаемые значения

Введение в словари. Тип словарь (dict). Словарные операции. Словарные методы.

Множества в языке Python. Множества. Множественный тип данных. Описание множеств.

Операции, допустимые над множествами: объединение, пересечение, разность, включение. Оператор определения принадлежности элемента множеству.

Стиль программирования и отладка программ

Стиль программирования. Отладка программ.

Зачет по курсу «Программирование на языке Python»

Тематическое планирование

Название раздела (модуля), темы, количество часов	Характеристика видов деятельности учащихся
Знакомство с языком Python (2 часа)	знакомство с языком Python; изучить структуру программы на Python, · режимы работы с Python. выполнить установку программы; выполнить простейшую программу в интерактивной среде; написать комментарии в программе.
Переменные и выражения (6 часов)	изучить операторы ввода-вывода. работать со справочной системой; решать задачи на элементарные действия с числами; пользоваться интерфейсом среды программирования Python; использовать команды редактора; организовывать ввод и вывод данных; записывать арифметические выражения.
Условные предложения (5 часов)	изучить назначение условного оператора; изучить способы записи условного оператора; изучить логический тип данных; изучить логические операторы or, and, not; использовать условный оператор; создавать сложные условия с помощью логических операторов.; решать задачи по теме "Условные операторы"; составлять программы с ветвлением
Циклы (6 часов)	изучить циклы с условием и их виды;

	<p>изучить правила записи циклов условием;</p> <p>изучить назначение и особенности использования цикла с параметром;</p> <p>изучить формат записи цикла с параметром;</p> <p>изучить примеры использования циклов различных типов.</p> <p>решать задачи с циклом for;</p> <p>реализовывать циклические алгоритмы</p> <p>составлять программы с циклом;</p> <p>определять вид цикла, наиболее удобный для решения поставленной задачи;</p> <p>использовать цикл с условием;</p> <p>определять целесообразность применения и использовать цикл с параметром для решения поставленной задачи.</p>
Функции (4 часа)	<p>изучить способы описания функции;</p> <p>изучить принципы структурного программирования;</p> <p>изучить понятие локальных переменных подпрограмм;</p> <p>изучить понятие формальных и фактических параметров подпрограмм;</p> <p>изучить способы передачи параметров.</p> <p>решать задачи с использованием функций;</p> <p>-решать задачи с использованием рекурсивных функций;</p> <p>создавать и использовать функции;</p> <p>использовать механизм параметров для передачи значений.</p>
Строки - последовательности символов (3 часа)	<p>изучить назначение строкового типа данных;</p> <p>изучить операторы для работы со строками;</p> <p>изучить процедуры и функции для работы со строками;</p> <p>изучить операции со строками.</p> <p>решать задачи со строками;</p> <p>описывать строки;</p> <p>соединять строки;</p> <p>находить длину строки;</p> <p>вырезать часть строки;</p> <p>находить подстроку в строке;</p> <p>находить количество слов в строке.</p>
Сложные типы данных (6 часов)	<p>изучить сложные типы данных;</p> <p>изучить способы описания списка;</p> <p>изучить способы доступа к элементам списка;</p> <p>изучить способы описания кортежа;</p> <p>изучить способы описания словаря;</p> <p>изучить операции, выполняемые со списками, кортежами и словарями;</p> <p>изучить понятие множества;</p> <p>изучить способы описания множества;</p> <p>изучить операторы работы с множествами.</p> <p>описывать списки;</p> <p>вводить элементы списка;</p>

	<p>выводить элементы списка;</p> <p>выполнять поиск элемента в списке, поиск минимума и максимума, нахождение суммы элементов списка;</p> <p>использовать вложенные списки;</p> <p>приводить примеры использования вложенных списков (матриц);</p> <p>описывать множества;</p> <p>определять принадлежность элемента множеству;</p> <p>вводить элементы множества;</p> <p>выводить элементы множества;</p> <p>решать задачи со списками.</p>
<p>Стиль программирования и отладка программ (4 часа)</p>	<p>изучить, что такое стиль программирования;</p> <p>изучить правила именования объектов;</p> <p>изучить основные рекомендации при написании программ.</p> <p>определять вид ошибок и находить ошибки в программе.</p> <p>выполнять тестирование и отладку программ.</p>

Планируемые результаты освоения обучающимися основной образовательной программы основного общего образования

Программа предполагает достижение выпускниками основной школы следующих личностных, метапредметных и предметных результатов.

В личностных результатах сформированность:

- Сформированность мировоззрения, соответствующего современному уровню развития науки и техники;
- потребность в самореализации в творческой деятельности, желании учиться;
- умение осуществлять совместную информационную деятельность, в частности, при выполнении учебных проектов;
- осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов; отношение к профессиональной деятельности как возможности участия в решении личных, общественных, государственных, общенациональных проблем.

В метапредметных результатах сформированность:

- Умение самостоятельно выделять и формулировать познавательные цели;
- умение применять методы моделирования;
- умение прогнозировать результат деятельности и его характеристики, вносить необходимые коррективы в план по ходу его выполнения;
- приобретение навыков самостоятельного создания способов решения проблем творческого и поискового характера;
- умение использовать средства ИКТ в решении когнитивных, коммуникативных и организационных задач с соблюдением требований эргономики, техники безопасности, гигиены, ресурсосбережения, правовых и этических норм, норм информационной безопасности;
- готовность и способность к самостоятельной информационно-познавательной деятельности, включая умение ориентироваться в различных источниках информации, критически оценивать и интерпретировать информацию, получаемую из различных источников;
- умение продуктивно общаться и взаимодействовать в процессе совместной деятельности, учитывать позиции других участников деятельности, эффективно разрешать конфликты.

В предметных результатах сформированность:

- Умение интерпретировать сообщения с позиции их смысла, синтаксиса, ценности;
- умение планировать действия, необходимые для достижения заданной цели (связанной с преобразованием информации) с помощью фиксированного набора средств, способность реализовать эти действия с использованием языка программирования, оценивать результаты работы;
- приобретение опыта планирования учебного сотрудничества с учителем и сверстниками: определение цели, функции участников, способов взаимодействия;
- овладение понятием сложности алгоритма, знание основных алгоритмов обработки числовой и текстовой информации, алгоритмов поиска и сортировки;
- овладение навыками алгоритмического мышления и пониманием необходимости формального описания алгоритмов;
- умение реализовывать алгоритмы в виде программ и программных систем; владение элементарными навыками документирования программ.
- умение сопоставлять математические модели задач и их компьютерные аналоги, анализировать полученные результаты с точки зрения соответствия объекту и целям моделирования;
- умение анализировать разные способы записи алгоритмов с позиции того, что они являются информационными моделями;
- владение универсальным языком программирования высокого уровня, представлениями о базовых типах данных и структурах данных; умением использовать основные управляющие конструкции языка; умением производить отладку программ с помощью встроенного отладчика.

Планируемые результаты изучения курса

Курс	Выпускник научится	Выпускник получит возможность научиться
Основы программирование на Python	<ul style="list-style-type: none">• понимать смысл понятия «алгоритм», приводить примеры алгоритмов;• понимать правила записи и выполнения алгоритмов, содержащих алгоритмические конструкции «следование», «ветвление», «цикл»;• подбирать алгоритмическую конструкцию, соответствующую заданной ситуации;• разрабатывать план действий для решения поставленных задач;• создавать и использовать функции;	<ul style="list-style-type: none">• научиться исполнять алгоритмы, содержащие ветвления и повторения• научиться по данному алгоритму определять, для решения какой задачи он предназначен;• научиться разрабатывать в среде исполнителя короткие алгоритмы, содержащие базовые алгоритмические конструкции и вспомогательные алгоритмы;• создавать приложения на Python

Критерии оценки предметных планируемых результатов обучающихся по курсу

Оцениваются знания и умения учащихся с учетом их индивидуальных особенностей. Содержание и объем материала, подлежащего проверке, определяется программой. Основными формами проверки знаний и умений учащихся по курсу являются практические и контрольные работы.

Оценка практических и контрольных работ обучающихся

При проверке усвоения материала выявляется полнота, прочность усвоения учащимися теории и умение применять ее на практике в знакомых и незнакомых

ситуациях. Отметка зависит также от наличия и характера погрешностей, допущенных учащимися.

- *грубая ошибка* – полностью искажено смысловое значение понятия, определения;
- *погрешность* отражает неточные формулировки, свидетельствующие о нечетком представлении рассматриваемого объекта;
- *недочет* – неправильное представление об объекте, не влияющего кардинально на знания определенные программой обучения;
- *мелкие погрешности* – неточности в устной и письменной речи, не искажающие смысла ответа или решения, случайные опiski и т.п.

Ответ оценивается отметкой «5», если все задания выполнены полностью или в работе имеется 1 недочет.

Ответ оценивается отметкой «4», если в работе имеется 2 недочета или одна ошибка.

Ответ оценивается отметкой «3», если выполнено 2/3 от объема предложенных заданий;

Ответ оценивается отметкой «2», если допущены существенные ошибки, показавшие, что учащийся не владеет обязательными умениями по данной теме в полной мере (незнание основного программного материала):

Процент выполнения работы	Достигнутый уровень	Отметка
80 –100 %	Работа выполнена на высоком уровне	«5»
65– 79%	Работа выполнена на среднем уровне	«4»
50 – 64 %	Работа выполнена на базовом уровне	«3»
0 – 49%	Неудовлетворительный уровень выполнения работы	«2»

Перечень контрольных и практических работ

1. Форматированный вывод. Вывод на экран рисунок с помощью символов
2. Вычисления по формулам. Расчет треугольника и других фигур
3. Работа с разрядами целого числа
4. Составление и вычисление логических выражений
5. Полные и неполные ветвления
6. Сложные условия
7. Цикл while и целые числа
8. Цикл while и клавиатурный ввод
9. Цикл while и генератор случайных чисел
10. Рекуррентные соотношения
11. Расчеты сходящихся числовых рядов с точностью ϵ
12. Цикл for и клавиатурный ввод
13. Цикл for и генератор случайных чисел
14. Использование условной инструкции if в теле цикла while
15. Использование условной инструкции if в теле цикла for
16. Вложенные циклы
17. Вложенные циклы и целые числа
18. Работа с символами строки
19. Обработка строк с использованием оператора цикла с параметром
20. Обработка строк с использованием операторов цикла с условием
21. Функции и процедуры
22. Рекурсия