

Муниципальное общеобразовательное учреждение
средняя общеобразовательная школа №1
пгт. Новокручининский

«РАССМОТРЕНО» Руководитель МО  / Конева М.В. Протокол № <u>1</u> от « <u>29</u> » <u>августа</u> 2022 г.	«СОГЛАСОВАНО» Заместитель директора по УВР  / Кучеренко Н.Е. Протокол № <u>1</u> от « <u>31</u> » <u>августа</u> 2022 г.	«УТВЕРЖДЕНО» Директор МОУ СОШ №1  / Маниковская В.В. Приказ № <u>105</u> от « <u>05</u> » <u>сентября</u> 2022 г.
--	--	---

Дополнительная общеразвивающая программа

Технической направленности

Основы программирования на PYTHON

с использованием
оборудования центра «Точка роста»

Составила:

Конева Марина Владимировна

учитель математики, информатики

2022 год

Дополнительная общеобразовательная (общеразвивающая) программа **«Основы программирования на PYTHON»** имеет **техническую** направленность.

Рабочая программа курса внеурочной деятельности «Основы программирования на Python» (далее — курс) для учащихся 7—11 классов составлена на основе:

- требований Федерального государственного образовательного стандарта основного общего образования к результатам освоения основной программы основного общего образования (Приказ Министерства просвещения Российской Федерации от 31.05.2021 № 287 «Об утверждении федерального государственного образовательного стандарта основного общего образования»),
- с учётом Примерной программы воспитания (протокол Федерального учебно-методического объединения по общему образованию № 3/22 от 23.06.2022)
- Примерной основной образовательной программы основного общего образования (протокол Федерального учебно-методического объединения по общему образованию № 1/22 от 18.03.2022).

В современную жизнь человека всё шире внедряются компьютеры и информационные технологии. Поэтому всё большее значение приобретает компьютерная грамотность.

Курс внеурочной деятельности отражает и расширяет содержание четырёх тематических разделов информатики на уровне основного общего образования:

- 1) цифровая грамотность;
- 2) теоретические основы информатики;
- 3) алгоритмы и программирование;
- 4) информационные технологии.

Новизна программы

Новизна Программы заключается в том, что Python даёт более широкие возможности в области программирования, чем Pascal, который входит в школьный курс информатики. На языке Python можно легко и быстро создавать простые компьютерные игры, трёхмерные модели и программировать роботов. Этот язык быстрее и легче усваивается, чем Pascal. Многие мировые компании такие, как Intel, Cisco, Hewlett-Packard, используют этот язык при реализации своих проектов. Крупнейшие интернет-ресурсы такие, как Google, YouTube, также разработаны с помощью языка программирования Python.

Актуальность программы

С развитием современных информационных технологий сегодня

любой учащийся под руководством опытного педагога может с лёгкостью научиться программировать.

Компьютеры и компьютерные системы – неотъемлемая часть жизни нашего общества. Научившись программировать, мы можем быть не только пользователями информационных технологий, но и активными их создателями.

Языки программирования можно сравнить с иностранными языками, овладеть ими может каждый. Учиться программировать очень интересно. Результат программирования очень часто виден сразу. Кроме того, создание компьютерных игр и обучающих программ способствует развитию логики и креативного мышления. Ещё одной значимой стороной обучения программированию является спрос на рынке труда на специалистов данного направления деятельности.

Педагогическая целесообразность Программы

Педагогическая целесообразность программы заключается в привлечении учащихся к занятиям техническим творчеством, что способствует развитию логического мышления, творческих способностей и навыков решения задач программирования. Программирование мотивирует к занятиям в различных научных областях (физики, информатики, алгебры, геометрии и др.), развивает воображение и способствует ранней профориентации подростков. Для достижения поставленных задач занятия проводятся в формате «от простого к сложному». Учащиеся вспоминают свои знания по основам алгоритмизации и программирования и на их основе, углубляя их, учатся составлять простые и сложные программы.

Целями изучения курса внеурочной деятельности «Основы программирования на Python» являются:

формирование основ мировоззрения, соответствующего современному уровню развития науки информатики, достижениям научно-технического прогресса и общественной практики, за счёт развития представлений об информации как о важнейшем стратегическом ресурсе развития личности, государства, общества; понимание роли информационных процессов, информационных ресурсов и информационных технологий в условиях цифровой трансформации многих сфер жизни современного общества;

обеспечение условий, способствующих развитию алгоритмического мышления как необходимого условия профессиональной деятельности в современном информационном обществе, предполагающего способность обучающегося разбивать сложные задачи на более простые подзадачи; сравнивать новые задачи с задачами, решёнными ранее; определять шаги для достижения результата и т. д.;

формирование цифровых навыков, в том числе ключевых компетенций цифровой экономики, таких как базовое программирование на Python, основы работы с данными, коммуникация в современных цифровых средах, информационная безопасность; воспитание ответственного и избирательного отношения к информации;

формирование необходимых для успешной жизни в меняющемся мире универсальных учебных действий (универсальных компетентностей) на основе средств и методов информатики и информационных технологий, в том числе овладение умениями работать с различными видами информации, самостоятельно планировать и осуществлять индивидуальную и коллективную информационную деятельность, представлять и оценивать её результаты; формирование и развитие компетенций обучающихся в области использования информационно-коммуникационных технологий, в том числе знаний, умений и навыков работы с информацией, программирования, коммуникации в современных цифровых средах в условиях обеспечения информационной безопасности обучающегося;

воспитание ответственного и избирательного отношения к информации с учётом правовых и этических аспектов её распространения, стремления к продолжению образования в области информационных технологий и созидательной деятельности с применением средств информационных технологий.

Основные задачи курса внеурочной деятельности «Основы программирования на Python» — сформировать у обучающихся:

понимание принципов устройства и функционирования объектов цифрового окружения, представления об истории и тенденциях развития информатики периода цифровой трансформации современного общества;

владение базовыми нормами информационной этики и права, основами информационной безопасности;

знания, умения и навыки грамотной постановки задач, возникающих в практической деятельности, их решения с помощью информационных технологий; умения и навыки формализованного описания поставленных задач;

базовые знания об информационном моделировании, в том числе о математическом моделировании;

знание основных алгоритмических структур и умение применять его для построения алгоритмов решения задач по их математическим моделям;

умения и навыки составления простых программ по построенному алгоритму на Python;

умения и навыки эффективного использования основных типов

прикладных программ (приложений) общего назначения и информационных систем для решения с их помощью практических задач; умение грамотно интерпретировать результаты решения практических задач с помощью информационных технологий, применять полученные результаты в практической деятельности.

Отличительные особенности данной программы

Основное количество часов отводится практическому написанию программ. Каждый обучающийся реализует индивидуальный проект в результате освоения программы. Продукт, полученный в результате освоения программы, имеет прикладной характер и может быть использован по необходимости.

Место курса внеурочной деятельности «Основы программирования на Python» в учебном плане.

Программа курса предназначена для организации внеурочной деятельности за счёт направления «Дополнительное изучение учебных предметов». Программа курса внеурочной деятельности рассчитана на 105 учебных часов, по 3 ч в неделю.

Возрастная категория обучающихся по программе от 13 до 18 лет (7-11 классы).

Срок реализации программы составляет 1 год.

Формы и режим занятий – групповые – для всей группы при изучении общих практических и теоретических вопросов. Наполняемость группы до 10 человек. В ходе реализации программы применяется дифференцированный, индивидуальный подход к каждому обучающемуся. Занятия проводятся 2 раза в неделю по 1,5 часа.

Планируемые результаты освоения курса внеурочной деятельности «Основы программирования на Python»

ЛИЧНОСТНЫЕ РЕЗУЛЬТАТЫ

Патриотическое воспитание:

ценностное отношение к отечественному культурному, историческому и научному наследию;

понимание значения информатики как науки в жизни современного общества.

Духовно-нравственное воспитание:

ориентация на моральные ценности и нормы в ситуациях нравственного выбора;

готовность оценивать своё поведение и поступки, а также поведение и поступки других людей с позиции нравственных и правовых норм с учётом осознания последствий поступков;

активное неприятие асоциальных поступков, в том числе в

Интернете.

Гражданское воспитание:

представление о социальных нормах и правилах межличностных отношений в коллективе, в том числе в социальных сообществах;

соблюдение правил безопасности, в том числе навыков безопасного поведения в интернет-среде;

ориентация на совместную деятельность при выполнении учебных и познавательных задач, создании учебных проектов;

стремление оценивать своё поведение и поступки своих товарищей с позиции нравственных и правовых норм с учётом осознания последствий поступков.

Ценность научного познания:

наличие представлений об информации, информационных процессах и информационных технологиях, соответствующих современному уровню развития науки и общественной практики;

интерес к обучению и познанию;

любопытность;

стремление к самообразованию;

овладение начальными навыками исследовательской деятельности, установка на осмысление опыта, наблюдений, поступков и стремление совершенствовать пути достижения индивидуального и коллективного благополучия;

наличие базовых навыков самостоятельной работы с учебными текстами, справочной литературой, разнообразными средствами информационных технологий, а также умения самостоятельно определять цели своего обучения, ставить и формулировать для себя новые задачи в учёбе и познавательной деятельности, развивать мотивы и интересы своей познавательной деятельности.

Формирование культуры здоровья:

установка на здоровый образ жизни, в том числе и за счёт освоения и соблюдения требований безопасной эксплуатации средств ИКТ.

Трудовое воспитание:

интерес к практическому изучению профессий и труда в сферах деятельности, связанных с информатикой, программированием и информационными технологиями, основанными на достижениях науки информатики и научно-технического прогресса.

Экологическое воспитание:

наличие представлений о глобальном характере экологических проблем и путей их решения, в том числе с учётом возможностей ИКТ.

Адаптация обучающегося к изменяющимся условиям социальной

среды:

освоение обучающимися социального опыта, основных социальных ролей, соответствующих ведущей деятельности возраста, норм и правил общественного поведения, форм социальной жизни в группах и сообществах, в том числе в виртуальном пространстве.

МЕТАПРЕДМЕТНЫЕ РЕЗУЛЬТАТЫ

Универсальные познавательные действия:

Базовые логические действия:

умение определять понятия, создавать обобщения, устанавливать аналогии, классифицировать, самостоятельно выбирать основания и критерии для классификации, устанавливать причинно-следственные связи, строить логические рассуждения, делать умозаключения (индуктивные, дедуктивные и по аналогии) и выводы;

умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебных и познавательных задач;

самостоятельно выбирать способ решения учебной задачи (сравнивать несколько вариантов решения, выбирать наиболее подходящий с учётом самостоятельно выделенных критериев).

Базовые исследовательские действия:

формулировать вопросы, фиксирующие разрыв между реальным и желательным состоянием ситуации, объекта, и самостоятельно устанавливать искомое и данное;

оценивать применимость и достоверность информации, полученной в ходе исследования;

прогнозировать возможное дальнейшее развитие процессов, событий и их последствия в аналогичных или сходных ситуациях, а также выдвигать предположения об их развитии в новых условиях и контекстах.

Работа с информацией:

выявлять дефицит информации, данных, необходимых для решения поставленной задачи;

применять основные методы и инструменты при поиске и отборе информации из источников с учётом предложенной учебной задачи и заданных критериев;

выбирать, анализировать, систематизировать и интерпретировать информацию различных видов и форм представления;

выбирать оптимальную форму представления информации и иллюстрировать решаемые задачи несложными схемами, диаграммами, иными графическими объектами и их комбинациями;

оценивать достоверность информации по критериям, предложенным учителем или сформулированным самостоятельно;

запоминать и систематизировать информацию.

Универсальные коммуникативные действия

Общение:

сопоставлять свои суждения с суждениями других участников диалога, обнаруживать различие и сходство позиций;

публично представлять результаты выполненного опыта (исследования, проекта);

выбирать формат выступления с учётом задач презентации и особенностей аудитории и в соответствии с ним составлять устные и письменные тексты с использованием иллюстративных материалов.

Совместная деятельность (сотрудничество):

понимать и использовать преимущества командной и индивидуальной работы при решении конкретной проблемы, в том числе при создании информационного продукта;

принимать цель совместной информационной деятельности по сбору, обработке, передаче, формализации информации; коллективно строить действия по её достижению: распределять роли, договариваться, обсуждать процесс и результат совместной работы;

выполнять свою часть работы с информацией или информационным продуктом, достигая качественного результата по своему направлению и координируя свои действия с другими членами команды;

оценивать качество своего вклада в общий информационный продукт по критериям, самостоятельно сформулированным участниками взаимодействия;

сравнивать результаты с исходной задачей и вклад каждого члена команды в достижение результатов, разделять сферу ответственности и проявлять готовность к предоставлению отчёта перед группой.

Универсальные регулятивные действия

Самоорганизация:

выявлять в жизненных и учебных ситуациях проблемы, требующие решения;

составлять алгоритм решения задачи (или его часть), выбирать способ решения учебной задачи с учётом имеющихся ресурсов и собственных возможностей, аргументировать выбор варианта решения задачи;

составлять план действий (план реализации намеченного алгоритма решения), корректировать предложенный алгоритм с учётом получения новых знаний об изучаемом объекте.

Самоконтроль (рефлексия):

владеть способами самоконтроля, самомотивации и рефлексии;

учитывать контекст и предвидеть трудности, которые могут возникнуть при решении учебной задачи, адаптировать решение к меняющимся обстоятельствам;

вносить коррективы в деятельность на основе новых обстоятельств, изменившихся ситуаций, установленных ошибок, возникших трудностей;

оценивать соответствие результата цели и условиям.

Эмоциональный интеллект:

ставить себя на место другого человека, понимать мотивы и намерения другого.

Принятие себя и других:

осознавать невозможность контролировать всё вокруг даже в условиях открытого доступа к любым объёмам информации;

осознанно относиться к другому человеку, его мнению.

ПРЕДМЕТНЫЕ РЕЗУЛЬТАТЫ

К концу обучения обучающийся научится:

соблюдать требования безопасности при работе на компьютере;

объяснять, что такое алгоритм, язык программирования, программа;

использовать переменные различных типов при написании программ на Python;

использовать оператор присваивания при написании программ на Python;

искать ошибки в программном коде на Python и исправлять их;

дописывать программный код на Python;

писать программный код на Python;

использовать ветвления и циклы при написании программ на Python;

анализировать блок-схемы и программы на Python;

объяснять, что такое логическое выражение;

вычислять значение логического выражения;

записывать логическое выражение на Python;

понимать структуру адресов веб-ресурсов;

искать информацию в Интернете;

писать программы на Python для рисования различных геометрических фигур, используя модуль Turtle;

понимать различия локальных и глобальных переменных;

решать задачи с использованием глобальных переменных на Python;

писать свои функции на Python;

разбивать задачи на подзадачи;

использовать списки и словари при написании программ на Python;

разрабатывать веб-страницы, содержащие рисунки, списки и гиперссылки.

Формы определения результативности обучения

Результаты освоения программы отслеживаются по итогам опросов, выполнения практических заданий.

Формы аттестации и оценочные материалы

В процессе реализации программы предусмотрены следующие виды контроля:

- входной контроль проводится с целью определения уровня знаний учащихся;
- промежуточный контроль проводится регулярно на занятиях с целью определения степени усвоения материала в форме опроса, решения задач и практических заданий;
- итоговый контроль – защита проекта.

Формы подведения итогов реализации программы

Результаты обучения по программе выявляются по итогам проведения олимпиад, соревнований по программированию, защиты проекта.

СОДЕРЖАНИЕ КУРСА

1. История языков программирования. Компиляция и интерпретация.
2. Знакомство с Python и средами программирования.
3. Типы данных в программировании. Определение переменной.
4. Ввод данных с клавиатуры.
5. Логические выражения.
6. Условный оператор. Инструкция if.
7. Множественное ветвление.
8. Цикл While и For.
9. Строки как последовательности символов.
10. Списки — изменяемые последовательности. Массивы.
1. Основные задачи обработки массивов: поиск, сортировка, реверс.
11. Введение в словари.
12. Функции в программировании. Параметры и аргументы функций.
2. Локальные и глобальные переменные. Процедуры.
13. Файлы. Чтение текстового файла. Запись в файл.
14. Алгоритм Евклида (нахождение наибольшего общего делителя)
15. Вычисление факториала на языке программирования Python
16. Двоичный (бинарный) поиск элемента в массиве
17. Замена элементов в списке
18. Перевод чисел из десятичной системы счисления в двоичную

19. Решето Эратосфена - алгоритм определения простых чисел
20. Сортировка выбором (поиск минимума и перестановка)
21. Сортировка методом пузырька
22. Сумма и произведение цифр числа
23. Тестирование простоты числа методом перебора делителей
24. Числа Фибоначчи (вычисление с помощью цикла while и рекурсии)

УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

№ п/п	Тема	Виды учебной деятельности	Количество часов
1.	История языков программирования. Язык Python. Простейшие программы. Реализация вычислений и ветвлений.	знать место языка Python среди языков программирования высокого уровня, знать особенности структуры программы, представленной на языке Python, ° иметь представление о модулях, входящих в состав среды Python, знать возможности и ограничения использования готовых модулей, иметь представление о величине, ее характеристиках, знать что такое операция, операнд и их характеристики, знать принципиальные отличия величин структурированных и неструктурированных, иметь представление о логических выражениях и входящих в них операндах, операциях и функциях, уметь записывать примеры арифметических и логических выражений всех атрибутов, которые могут в них входить, знать основные операторы языка Python, их синтаксис, иметь представление о процессе исполнения каждого из операторов, знать принципиальные отличия между формальными, локальными и глобальными переменными	18
2.	Реализация циклических, вспомогательных алгоритмов. Рекурсия.	знать основные операторы языка Python, их синтаксис, иметь представление о процессе исполнения каждого из операторов, уметь разрабатывать программы обработки числовой и символьной информации, уметь разрабатывать программы (линейные, разветвляющиеся и с циклами), иметь представление о значении полноценных процедур и функций для структурно-ориентированного языка высокого уровня, знать правила описания процедур и функций в Python и построение вызова процедуры, знать область действия описаний в процедурах, иметь представление о рекурсии, знать ее реализацию на Python, владеть основными приемами формирования процедуры и функции	21
3.	Множества. Словари	знать основные операторы языка Python, их синтаксис, иметь представление о процессе исполнения каждого из операторов, уметь разрабатывать программы обработки числовой и символьной информации, уметь разрабатывать программы (линейные, разветвляющиеся и с циклами)	16
4.	Символьные строки. Обработка символьных строк.	знать основные операторы языка Python, их синтаксис, иметь представление о процессе исполнения каждого из операторов, уметь разрабатывать программы обработки числовой и символьной информации, уметь разрабатывать	21

		программы (линейные, разветвляющиеся и с циклами)	
5.	Матрицы. Ввод, вывод, обработка матриц. Чтение и запись текстовых файлов.	знать основные операторы языка Python, их синтаксис, иметь представление о процессе исполнения каждого из операторов, уметь разрабатывать программы обработки числовой и символьной информации, уметь разрабатывать программы (линейные, разветвляющиеся и с циклами), определять в программе тип «массив», «матрица», знать свойства данных типа «массив», «матрица», уметь воспроизводить алгоритмы сортировки массивов и матриц, поиска в упорядоченном массиве, распространять эти алгоритмы на сортировку и поиск в нечисловых массивах, уметь читать и записывать текстовые файлы в заданном формате	27
6.	Итоговое занятие	решать олимпиадные задачи в среде Python.	2
Итого:			105

Календарно-тематическое планирование

№ п/п	Название раздела, темы	Всего часов	Дата проведения	
			План.	Факт.
I	История языков программирования. Язык Python. Простейшие программы. Реализация вычислений и ветвлений	18		
1.	История языков программирования	1		
2.	Язык Python	1		
3.	Оператор ввода – input	1		
4.	Оператор вывода – print	1		
5.	Типы данных	1		
6.	Тип данных int (целочисленный)	1		
7.	Операции над типом int (целочисленное деление, остаток от деления)	1		
8.	Решение задач	1		
9.	Тип данных float (вещественный)	1		
10.	Операции над типом float	1		
11.	Решение задач	1		
12.	Ветвление. Полное ветвление	1		
13.	Ветвление. Неполное ветвление	1		
14.	Знакомство с конструкцией if...elif...else	1		
15.	Логические операнды	1		
16.	Решение задач с помощью and, or	1		
17.	Вложенное ветвление	1		
18.	Решение задач	1		
II	Реализация циклических, вспомогательных алгоритмов. Рекурсия	21		
19.	Знакомство с понятием «цикл»	1		
20.	Цикл for, итерация цикла	1		
21.	Решение задач с помощью for	1		
22.	Выход из цикла for с помощью break	1		
23.	Конструкция for...else	1		
24.	Решение задач	1		

25.	Цикл с условием	1		
26.	Решение задач на цикл с условием	1		
27.	Цикл while	1		
28.	Решение задач с помощью while	1		
29.	Выход из цикла while с помощью break	1		
30.	Понятие вложенного цикла	1		
31.	Решение задач на все типы циклов	1		
32.	Понятие функции – def	1		
33.	Описание переменных в функции, локальные и глобальные переменные	1		
34.	Решение задач на вызов функции	1		
35.	Понятие рекурсии в python	1		
36.	Понятие рекурсии в python	1		
37.	Решение типовых задач на рекурсию	1		
38.	Решение задач на замену цикла for рекурсией	1		
39.	Решение задач на замену рекурсии циклом for	1		
III	Множества. Словари	16		
40.	Понятие «Множества» в python	1		
41.	Ввод множества с клавиатуры	1		
42.	Запись в множества с помощью if	1		
43.	Проход по значениям множества с помощью for	1		
44.	Решение задач по теме «Множества»	1		
45.	Понятие «Словари» в python	1		
46.	Ввод с клавиатуры в словари	1		
47.	Понятие значения в словарях	1		
48.	Понятие значения ключа в словарях	1		
49.	Решение задач	1		
50.	Запись в множества через условия	1		
51.	Добавления элементов в множество	1		
52.	Знакомство с методами множества	1		
53.	Решение задач	1		
54.	Проход по ключам-значениям в множестве	1		

55.	Решение задач по множествам	1		
IV	Символьные строки. Обработка символьных строк	21		
56.	Ввод с клавиатуры	1		
57.	Операции над строками	1		
58.	Решение задач	1		
59.	Срезы в строках	1		
60.	Проход по элементам в строках с помощью for	1		
61.	Решение задач	1		
62.	Создание пустой строки	1		
63.	Добавление элементов в новую строку через условие	1		
64.	Вывод элементов строки в одну строку	1		
65.	Решение задач	1		
66.	Методы строк. Проверка на числа	1		
67.	Методы строк. Проверка на буквы	1		
68.	Решение задач	1		
69.	Методы строк. Проверка на заглавные символы	1		
70.	Методы строк. Поиск по значению	1		
71.	Методы строк. Метод count	1		
72.	Методы строк. Метод reversed	1		
73.	Решение задач	1		
74.	Методы строк. Перевод из строки в число	1		
75.	Методы строк. Перевод из типа int в тип str	1		
76.	Решение задач	1		
V	Матрицы. Ввод, вывод, обработка матриц. Чтение и запись текстовых файлов	27		
77.	Понятие массива в программировании	1		
78.	Понятие list (список) в python	1		
79.	Ввод списка с клавиатуры	1		
80.	Вывод списка	1		
81.	Решение задач	1		
82.	Понятие индекса и значение массива (списка)	1		

83.	Методы списка. Add	1		
84.	Методы списка. Pop	1		
85.	Решение задач	1		
86.	Методы списка. Copy	1		
87.	Методы списка. Clear	1		
88.	Решение задач	1		
89.	Методы списка. Reverse	1		
90.	Методы списка. Index	1		
91.	Решение задач	1		
92.	Понятие файла в программирование	1		
93.	Открытие файла в python	1		
94.	Запись в файл в python	1		
95.	Решение задач	1		
96.	Прохождение по элементам списка	1		
97.	Понятие «Список в списке» (двумерный массив)	1		
98.	Решение задач	1		
99.	Перевод элемента списка в тип данных int	1		
100.	Перевод из типа данных str в list	1		
101.	Перевод из типа данных list в str	1		
102.	Знакомство с методом join	1		
103.	Решение задач	1		
V	Итоговое занятие	27		
104.	Решение олимпиадных задач	1		
105.	Решение олимпиадных задач	1		