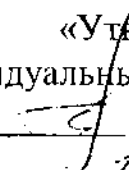


Индивидуальный предприниматель Майоров Рустем Марванович

«Утверждаю»
Индивидуальный предприниматель
 Майоров Р.М.
«19» 12 2022 г.
Приказ № 16 от 19.12.2022 г.

«Рассмотрено»
педагогическим советом
ЦРШ «Код Успеха»
Протокол № 1 от 12.12.2022 г.

Дополнительная общеобразовательная общеразвивающая программа
технической направленности «Программирование.Python.Подготовка.Уровень 1»

Возраст обучающихся: 8 - 9 лет
Срок реализации: 1 год
Форма обучения – очная
Уровень программы - начальный

Автор-составитель:
Майорова Оксана Юрьевна
Полувесова Юлия Борисовна

Оглавление

1.	Пояснительная записка	3
1.1.	Направленность	3
1.2.	Актуальность программы	3
1.3.	Педагогическая целесообразность	4
1.4.	Цель и задачи программы	4
1.5.	Планируемые результаты	5
1.6.	Отличительные особенности программы	7
1.7.	Условия реализации программы	7
1.8.	Формы организации занятий	8
2.	Календарный учебный график	8
3.	Учебный план	8
4.	Содержание программы	9
5.	Оценочные и методические материалы	11
5.1.	Формы аттестации	11
5.2.	Методическое обеспечение	12
6.	Материально-техническое обеспечение программы	14
6.1.	Перечень необходимого оборудования	14
6.2.	Учебно-методическое обеспечение	14
7.	Кадровое обеспечение	14
8.	Список литературы	14
8.1.	Литература для педагога	14
8.2.	Литература для учащихся	15
8.3.	Интернет-ресурсы	15

1 Пояснительная записка

Дополнительная общеобразовательная программа «Программирование.Python.Подготовка.Уровень 1» разработана согласно требованиям следующих нормативных документов:

- Федеральный Закон «Об образовании в Российской Федерации» от 29.12.2012 № 273-ФЗ.
- Концепция развития дополнительного образования детей (утверждена распоряжением Правительства РФ от 04.09.2014 № 1726-р).
- СП 2.4.3648-20 "Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи"
- СанПиН 1.2.3685-21 "Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания"
- Порядок организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам (утвержден приказом Министерства просвещения РФ от 9 ноября 2018 г. N 196).
- Стратегия развития воспитания в Российской Федерации на период до 2025 года (утверждена Распоряжением Правительства Российской Федерации от 29 мая 2015 г. N 996-р)

В современном мире обучение детей программированию начинается всё в более раннем возрасте. В школьном курсе изучение информатики начинается с седьмого класса, однако в наше время олимпиады по программированию проводятся с третьего класса. Дополнительная общеобразовательная программа «Программирование.Python.Подготовка» направлена на устранение данного пробела.

1.1. Направленность

Направленность дополнительной образовательной программы «Программирование.Python.Подготовка.Уровень 1» – научно-техническая. Она является прикладной, носит практико-ориентировочный характер и направлена на овладение воспитанниками основных приемов программирования. Обучение по данной программе создаст благоприятные условия для интеллектуального и духовного воспитания личности ребенка, социально - культурного и профессионального самоопределения, развития познавательной активности и творческой самореализации учащихся.

1.2. Актуальность программы

Данная дополнительная общеобразовательная программа соотносится с тенденциями развития дополнительного образования и согласно Концепции развития дополнительного образования способствует:

- созданию необходимых условий для личностного развития обучающихся, позитивной социализации и профессионального самоопределения;
- удовлетворению индивидуальных потребностей обучающихся в интеллектуальном развитии, а также в занятиях научно-техническим творчеством;
- формированию и развитию творческих способностей учащихся, выявлению, развитию и поддержке талантливых учащихся;
- обеспечению духовно-нравственного, гражданского, патриотического, трудового воспитания учащихся;

Кроме того, актуальность программы обуславливается тем, что при обычном обучении информатике, тема «программирование» изучается очень мало и поздно, это не способствует развитию интереса учащихся в области программирования, учащиеся, как правило, не готовы успешно выступать на олимпиадах по информатике, теряют интерес к предмету. Обучение по данной программе создаёт благоприятные условия для более раннего «погружения» учащихся в мир логики, математического моделирования, для интеллектуального и духовного воспитания личности ребенка, социально - культурного и профессионального самоопределения, развития познавательной активности и творческой самореализации учащихся. Она рассчитана на сотворчество и сотрудничество педагога и воспитанников. Данная программа даст возможность детям творчески мыслить, находить самостоятельные индивидуальные решения, а полученные умения и навыки применять в жизни.

1.3. Педагогическая целесообразность

Педагогическая целесообразность заключается в том, что сделанный в представленной программе акцент на практическую работу позволяет повысить результативность обучения информатике и ИКТ при изучении данного дополнительного курса до начала обучения информатики в школьном курсе, расширить мировоззрение учащихся, повысить предметные и межпредметные ЗУН и УУН, подготовки учащихся успешно освоить учебный материал и участвовать в олимпиадах, осознанного выбора профиля дальнейшего обучения и будущей профессии.

1.4. Цель и задачи программы

Цель программы: развитие творческих способностей учащихся, овладение теоретическими и практическими навыками решения алгоритмических задач программными средствами.

Задачи:

образовательные:

обучить основным видам вычислительных процессов и способам их реализации с помощью синтаксических конструкций языка программирования Python;

обучить приемам работы в основных интегрированных средах программирования (IDLE, Wing);

сформировать навыки тестирования, поиска и исправления ошибок в программном коде;

воспитательные:

воспитывать культуру написания «чистого», понятного для других программистов, кода, соблюдения рекомендаций по выбору имен для идентификаторов;

воспитать культуру проектной деятельности, в том числе умение планировать, работать в коллективе; чувство ответственности за результаты своего труда, используемого другими людьми;

формировать установку на позитивную социальную деятельность в информационном обществе, недопустимость действий, нарушающих правовые и этические нормы работы с информацией;

содействовать повышению привлекательности науки, научно-технического творчества для подрастающего поколения;

формировать у подрастающего поколения ответственное отношение к своему здоровью и потребности в здоровом образе жизни, соблюдении правил гигиены;

развивающие:

быстро ориентироваться в динамично развивающемся и обновляющемся информационном пространстве; получать, использовать и создавать разнообразную информацию; принимать обоснованные решения и решать проблемы на основе полученных знаний, умений и навыков;

овладение умениями строить математические объекты, в том числе логические формулы и программы на формальном языке, удовлетворяющие заданному описанию; использовать общепользовательские инструменты и настраивать их для нужд пользователя; применять алгоритмы и приёмы программирования;

развитие алгоритмического мышления, способностей к формализации, элементов системного мышления;

приобретение опыта создания, редактирования, оформления, сохранения, передачи информационных объектов различного типа с помощью современных программных средств; построение компьютерных моделей, коллективной реализации информационных проектов, преодоление трудностей в процессе интеллектуального проектирования, информационной деятельности в различных сферах, востребованных на рынке труда; решения сложных задач и олимпиадных задач программирования.

1.5. Планируемые результаты

Планируемые результаты обучения по дополнительной общеобразовательной программе «Программирование.Python.Подготовка»:

Предметные:

- умение анализировать поставленные задачи и создавать соответствующие информационные модели;
- умение разрабатывать алгоритмы решения поставленных задач и выбирать оптимальные структуры данных;
- умение применять основные конструкции языка Python для реализации алгоритмов решения поставленных задач;
- умение тестировать и выполнять отладку программного кода, применять инструментальные средства отладки.

Личностные:

- чувство патриотизма, уважения к своему народу, гордости за свой край;
- ответственное отношение к труду, качеству своей деятельности;
- овладение основами саморазвития и самовоспитания в соответствии с общечеловеческими ценностями и идеалами гражданского общества; готовность и способность к самостоятельной, творческой и ответственной деятельности;
- умение общаться и сотрудничать со сверстниками, детьми старшего и младшего возраста, взрослыми в процессе образовательной, общественно полезной, творческой и других видов деятельности;
- навыки сотрудничества с взрослыми и сверстниками в разных социальных ситуациях, умение не создавать конфликтов и находить выходы из спорных ситуаций;
- сформированность познавательных интересов и творческая активность в данной области.

Метапредметные:

- умение определять и формулировать цель деятельности на занятии, проговаривать последовательность действий с помощью педагога;
- умение планировать, контролировать и оценивать свои действия в соответствии с поставленной задачей и условиями её реализации, определять наиболее эффективные способы достижения результата;
- навыки самостоятельного выполнения практических работ по предложенному плану с опорой блок-схемы;
- способность самостоятельно оценивать и принимать решения, определяющие стратегию поведения, с учетом гражданских и нравственных ценностей;
- умение понимать причины успеха/неуспеха учебной деятельности и способности конструктивно действовать даже в ситуациях неуспеха.

В результате обучения выпускники должны знать:

- Основные типы данных языка программирования Python;
- Основные конструкции языка программирования Python;
- Обозначения, применяемые для оформления блок-схем алгоритмов;

- Порядок применения инструментальных средств для разработки программ на языке Python;
 - Порядок применения инструментальных средств для отладки программ.
- должны уметь:**
- Выполнять блок-схемы алгоритмов;
 - Составлять программный код, реализующий тот или иной алгоритм;
 - Определять необходимые для решения задачи данные структуры данных;
 - Оформлять программный код с соответствием с общепринятыми правилами;
 - Осуществлять поиск исходных данных, приводящих к неверному результату, определять причину неверного результата и принимать решение по оптимальному исправлению ошибок в программном коде;
 - Работать в команде над решением задачи;
 - Проявлять чувство коллективизма, взаимопомощи, взаимопонимания;
 - Уважать труд своих товарищей;
 - Стремиться к совершенствованию своих навыков.

1.6. Отличительные особенности программы

Отличительные особенности данной дополнительной программы от существующих образовательных программ в том, что она содержит дополнительный изучаемый материал, значительно расширяет возможности формирования универсальных учебных и предметных навыков. В программе большое внимание уделяется практическим заданиям. Проверка заданий осуществляется комбинированным способом. На первом этапе автоматически с помощью тестирующей системы. На этом этапе отсекается большая часть неверных решений. Второй этап контроля – проверка отправленного кода преподавателем. На этом этапе проверяется, выполнены ли ограничения по используемым конструкциям языка, оптимальность решения и правильность оформления кода. Допущенные ошибки разбираются на занятии. Такой подход приучает детей к самостоятельной работе, освобождает преподавателя от рутины проверки заведомо неверных решений, тем самым повышая эффективность учебного процесса.

Все учебные материалы представлены в формате презентаций. Их удобно использовать с помощью мультимедийного проектора. Обучаемые имеют к ним постоянный доступ в сети Интернет и могут использовать их в качестве опорного конспекта.

1.7. Условия реализации программы

В освоении настоящей учебной программы участвуют дети в возрасте 8 - 9 лет. В группу зачисляются все желающие возраста 8-9 лет.

Форма обучения – очная.

Формирование групп целесообразно осуществлять со схожим уровнем навыков решения олимпиадных задач по математике. Оценка указанного уровня возможна на основании рекомендаций педагогов, ведущих математику или на основе собеседования.

1.8. Формы организации занятий

Основной формой организации деятельности детей на занятии является групповое занятие. При изучении теоретических занятий педагог объясняет учебный материал с использованием слайдов и магнитно-маркерной доски.

Закрепление изученного материала осуществляется самостоятельной работой с помощью педагога по решению задач на компьютере с использованием специализированного сайта test.kod-u.ru.

Периодически проводятся олимпиады, предполагающие самостоятельную работу по решению задач на компьютере без помощи педагога.

После олимпиад, как правило, педагогом осуществляется разбор задач. Для разбора могут привлекаться обучаемые, успешно решившие те или иные задачи в ходе олимпиады.

2 Календарный учебный график

Режим организации занятий по данной дополнительной общеобразовательной программе определяется календарным учебным графиком и соответствует нормам, утвержденным СП 2.4.3648-20 "Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи", СанПиН 1.2.3685-21 "Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания"

Начало занятий первого года обучения – 1 сентября.

Окончание занятий первого года обучения – 31 мая.

Всего учебных недель (продолжительность учебного года) – 37 недель.

Количество учебных дней: – 74 дня.

Объем учебных часов:– 148 часа.

Режим работы: 2 раза в неделю по 2 часа.

Продолжительность учебного времени: 2 урока по 25 мин

Перерыв между уроками – 10 мин.

3 Учебный план

№ п/п	Название раздела, темы	Количество часов			Формы аттестации, контроля
		Всего	Теория	Практика	
1.	Введение. Основные понятия	64	9	55	Текущий контроль

2.	Вычисления	20	1	19	Текущий контроль
3.	Условная инструкция	20	4	16	Текущий контроль
4.	Цикл while	22	4	18	Текущий контроль
5.	Цикл for	22	3	19	Текущий контроль
	Итого	148			

4 Содержание программы

Тема 1. Введение. Основные понятия

Цели и задачи:

Показать обучаемым место изучаемого материала в общей системе информационных технологий. Дать необходимые сведения для начала работы. Изучить особенности работы с тестирующей системой. Ввести основные понятия языка программирования и показать взаимосвязь между ними.

Результаты освоения:

Самостоятельное написание программ с использованием возможностей функции `print()`, `input()`. Знать основные понятия языка. Уметь строить и разбирать выражения. Понимать различие между типами данных `str`, `int`, `float`, умение использовать их для решения задач, а также переводить из одного типа данных в другой. Умение использовать оператор склеивания "+".

Краткое содержание занятий:

Применение компьютеров, алгоритмы, устройство ЭВМ, машинные команды, взаимодействие программ, понятие языка программирования, среда IDLE режим калькулятора и запуск файлов. Работа в режиме калькулятора: повторный запуск команд. Функция `print()`, возможности функции `print()`. Важность самостоятельной работы (не гуглить и работать дома). Регистрация на сайте. Поиск курса и добавление в избранное. Доступ к слайдам. Порядок сдачи задач. Вычисления, переменная (Модель «коробки»), присвоение, Множественное присваивание. Обмен значений переменных. Последовательное присвоение (цепочка). Идентификаторы. Зарезервированные слова. Взаимозаменимость переменных и значений.

Понятие функции (Пример `min`). Подпрограмма, скобки для запуска, без скобок - взятие адреса функции, понятие параметра (аргумента), возврат значения, пример `input()`. Термины оператор, операция, операнд. Выражение. Использование значений, переменных и функций в выражениях. Практика:

Ввод двух чисел и сложение. Понятие типа. Типы str, int, float. Преобразование типов. Вложенный вызов функций. (int+int, str+str, int+str). Int() для округления. Ввод нескольких значений. Комментарии в программах.

Тема 2. Вычисления

Цели и задачи:

Изучить целочисленное деление и нахождение остатка от него.

Результаты освоения:

Уметь решать задачи повышенного уровня на целочисленное деление.

Краткое содержание занятий:

Тип результата в разных операциях. Использование в циклических процессах операций // и % (неполное частное и взятие по модулю). Увеличение числа на 1. Сокращенная операция +=. Взятие по модулю от отрицательного числа.

Тема 3. Условная инструкция

Цели и задачи:

Познакомиться с видами ветвящихся процессов и изучить способы их реализации. Изучить логический тип данных.

Результаты освоения:

Умение применять для решения задач повышенного уровня условные инструкции, строить комбинированные логические выражения.

Краткое содержание занятий:

Последовательность выполнения инструкций. Ветвления. Условная инструкция. Любой тип, как логическое выражение. Использование отступов. Блок схема. Условия. Тип bool. Вычисление логических выражений. else. Вложенные условия. Каскадные условия. Пошаговая отладка программ в IDLE. Проверка вхождения подстроки в строку. Умножение строковых перемешных.

Тема 4. Цикл while

Цели и задачи:

Познакомиться с циклическими вычислительными процессами, изучить синтаксис и порядок работы инструкции while.

Результаты освоения:

Умение применять для решения задач инструкцию `while`.

Краткое содержание занятий:

Необходимость повторения действий. Структура циклов. Порядок работы цикла `while`. Заикливание и его предотвращение. Решение задач повышенного уровня на цикл `while`.

Тема 5. Цикл `for`**Цели и задачи:**

Изучить синтаксис и порядок работы инструкции `for`, изучить функцию `range`.

Результаты освоения:

Умение применять для решения задач инструкцию `for` и функцию `range`.

Краткое содержание занятий:

Понятие итерируемого объекта. Цикл `for` как перебор элементов итерируемого объекта. Функция `range` - создание итерируемого объекта из последовательности целых чисел. `Else` в циклах. Решение задач повышенного уровня на цикл `for`.

5 Оценочные и методические материалы**5.1. Формы аттестации**

Для оценки результативности обучения по дополнительной общобразовательной программе «Программирование.Python.Подготовка. Ур.1» применяется входной, текущий, промежуточный контроль.

Входной контроль

Предназначен для определения исходного уровня знаний и умений, проводится в форме собеседования..

Текущий контроль

Осуществляется автоматизировано с помощью автоматизированной системы на базе системы управления курсами Moodle, размещенной по адресу test.kod-u.ru. Задачи для решения находятся в разделе «Курсы / Программирование / Python.Подготовка».

Раздел считается освоенным, если задачи 100 % решены. Если задачи наборов решались с существенной помощью педагога, обучаемому предлагается перерешать задачи наборов.

Фиксация результатов осуществляется автоматически в разделе «Курсы / Программирование / Python.Подготовка / Управление курсом / Отчеты / Завершение элементов курса».

Промежуточный контроль

Проводится раз в три месяца в форме олимпиады по программированию. Порядок проведения описан в документе «Положение о закрытой олимпиаде по программированию ЦРШ "Код Успеха"».

Итоговый контроль

Для определения итогов освоения программы используется теоретический зачёт и результаты промежуточного контроля.

5.2. Методическое обеспечение

Дидактический материал

№ п/п	Тема	Формы занятий	Дидактический материал
1.	Введение. Основные понятия	Лекция, Практика	Введение. Основные понятия. Слайды
2.	Вычисления	Лекция, Практика	Вычисления. Слайды
3.	Условная инструкция	Лекция, Практика	Ветвления. Слайды
4.	Цикл while	Лекция, Практика	Цикл while. Слайды
5.	Цикл for	Лекция, Практика	Цикл for. Слайды

Приемы и методы организации образовательного процесса

При проведении лекционных занятий рекомендуется делать их максимально интерактивными, используя активизирующие вопросы, ставя перед обучающимися проблемы и обсуждая возможные пути их решения, анализируя их достоинства и недостатки. В ходе изложения материала избегать объяснения алгоритма приведением конкретного кода. Вместо этого

целесообразно использовать блок-схемы алгоритма. Объяснение алгоритма необходимо сопровождать примерами его работы на конкретных исходных данных.

При проведении практических занятий добиваться максимально самостоятельного решения задач. Обращать внимание на осознание необходимости самостоятельной работы. Сложность решаемых задач должна быть в зоне ближайшего развития.

При возникновении длительных затруднений давать небольшие подсказки в форме наводящих вопросов, графических схем, иллюстрирующих суть задачи. Начать помощь стоит с того, что нужно убедиться, что обучаемый правильно понял условие задачи. Для этого можно попросить пересказать условие и объяснить, как получается ответ на одном из примеров исходных данных.

Следующим шагом может быть предложить понаблюдать обучаемому над своими рассуждениями при решении задачи и словами описать алгоритм. Далее можно предложить ему составить блок-схему алгоритма. Если при этом допущены ошибки, попросить выполнить действия алгоритма, записывая при этом значения переменных на доске. В качестве исходных данных предложить такие, на которых предлагаемый учеником алгоритм дает неверный ответ. Далее совместно ищется способ исправить ошибку в блок-схеме. После этого ее предлагается закодировать.

В зависимости от задачи, иногда имеет смысл сразу писать код программы. В этом случае также можно попросить подробно рассказать, как она будет работать, записывая на доске значения переменных. Так же ищется и исправляется ошибка. Составленный общими усилиями код на доске целесообразно стереть, с тем, чтобы обучаемый воспроизвел его повторно самостоятельно. Исключение могут составлять очень слабые ученики.

Если код обучаемым уже написан и в нем «глупая ошибка», часто хорошо работает метод «duck debug». Обучаемому предлагается объяснить, как работает его программа. В ходе объяснения он находит ошибку.

При возникновении ситуации, когда часть тестов проходит, а часть получает вердикт «Неправильный ответ» необходимо ознакомить, а при повторных ситуациях напоминать правила составления тестов (проверка крайних значений исходных данных и пограничных значений в условиях и циклах; тесты должны заставить вычислительный процесс пройти по всем возможным маршрутам). Если тест неочевидный и плохо подбирается по изложенным выше правилам, допускается подсказать, на каких исходных данных программа выдает неправильный ответ.

После нахождения теста, дающего ошибку, необходимо научить вдумчиво и системно ее исправлять. Объяснить необходимость избегать применения «костылей» для исправления программы.

6 Материально-техническое обеспечение программы

6.1. Перечень необходимого оборудования

Техническое оснащение:

- Кабинет, имеющий хорошее освещение.
- Компьютеры (ноутбуки) из расчета один на обучаемого, с доступом к сети Интернет.
- Стулья, рабочие столы.
- Стол и стул педагога.
- Доска магнитно-маркерная.
- Проектор

Программные средства:

- Операционная система Windows 7 (или выше)
- Интерпретатор Python 3.5 (или выше)
- IDE: IDL, Wing 101, PyCharm

6.2. Учебно-методическое обеспечение:

1. Слайды по изучаемым темам.
2. Сайты <http://test.kod-u.ru/> с подготовленными разделами для реализации данной программы.

7 Кадровое обеспечение

Согласно Профессиональному стандарту «Педагог дополнительного образования детей и взрослых» по данной программе может работать педагог дополнительного образования с уровнем образования и квалификации, соответствующим обозначениям таблицы пункта 2 Профессионального стандарта (Описание трудовых функций, входящих в профессиональный стандарт), а именно: коды А и В с уровнями квалификации 6.

8 Список литературы

8.1. Литература для педагога

1. Федеральный закон «Об образовании в Российской Федерации» №273-ФЗ от 29.12.2012 (ст.2, п.9, п. 14; ст.12, п.5; ст.33, п.2; ст.75, п.2, п.4)
2. Приказ Министерства просвещения РФ от 9 ноября 2018 г. N 196 "Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам"
3. СП 2.4.3648-20 "Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и

молодежи"

4. СанПиН 1.2.3685-21 "Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания"
5. Концепция развития дополнительного образования детей от 04.09.2014 № 1726-р (раздел IV)
6. Методические рекомендации по проектированию дополнительных общеразвивающих программ (Приложение к письму Департамента государственной политики в сфере воспитания детей и молодежи Министерства образования и науки РФ от 18.11.2015 № 09-3242)
7. Python для детей. Самоучитель по программированию / Джейсон Бриггс; пер. с англ. Станислава Ломакина; [науч. ред. Д. Абрамова]. – 2-е изд. – М.:Мани, Иванов и Фербер, 2018. -320 с.
8. Программируем с Minecraft. Создай свой мир с помощью Python / Крэйг Ричардсон; пер. с англ. Станислава Ломакина; [науч. ред. Г. Гаджиев]. –М. Манн, Иванов и Фербер, 2017. -368с.: ил.
9. Олимпиадное программирование / Антти Лааксонен; пер. с англ. А.А. Слинкин – М.:ДМК Пресс, 2018. -300 с.: ил.

8.2. Литература для учащихся

10. Python для детей. Самоучитель по программированию / Джейсон Бриггс; пер. с англ. Станислава Ломакина; [науч. ред. Д. Абрамова]. – 2-е изд. – М.:Мани, Иванов и Фербер, 2018. -320 с.

8.3. Интернет-ресурсы

1. <https://pythonworld.ru>
2. <http://pythontutor.ru>
3. <https://www.afterhoursprogramming.com/tutorial/python>

