

Государственное бюджетное общеобразовательное учреждение
Ненецкого автономного округа «Средняя школа с.Ома»

СОГЛАСОВАНО

Заместитель директора школы по

УВР

Вольхина Вольхина Н.Г.
«22» августа 2024г.

УТВЕРЖДАЮ

Директор школы

Болтовская Болтовская Л.В.
Приказ № *101* от «23» 08 2024г.
«СШ с.Ома»



Дополнительная общеразвивающая программа
«IT лаборатория. Программирование на языке Python»

Учитель: *Салмина Лариса Николаевна*, первая квалификационная категория

РАССМОТРЕНО на заседании

Методического объединения учителей естественно-
математического цикла

Протокол № *1* от «22» 08 2024г.

Пояснительная записка

Задача построения в стране новой инновационной экономики и достижения технологического уровня, запланированного Концепцией долгосрочного социально-экономического развития Российской Федерации до 2020 года и долгосрочным прогнозом научно-технологического развития Российской Федерации до 2025 года, не может быть решена без существенных изменений системы дополнительного образования детей, создания новых общеразвивающих программ технической направленности.

Изменение взглядов на программирование как науку, его место в системе научного знания требует существенных изменений в содержании образовательного процесса. В связи с этим особую актуальность приобретают раскрытие личностных резервов учащихся и создание соответствующей образовательной среды.

Общепедагогическая направленность занятий – гармонизация индивидуальных и социальных аспектов обучения в отношении к информационным технологиям. Умение составлять алгоритмы решения и навыки программирования являются элементами информационной компетенции – одной из ключевых компетенций современной школы. Умение находить решение, составлять алгоритм решения и реализовать его с помощью языков программирования — необходимое условие подготовки современных учащихся. Особая роль отводится широко представленной в курсе системе рефлексивных заданий. Освоение рефлексии направлено на осознание учащимися того важного обстоятельства, что наряду с разрабатываемыми ими продуктами в виде программ на компьютере рождается основополагающий образовательный продукт: освоенный инструментарий. Именно этот образовательный продукт станет базой для творческого самовыражения учащихся в форме различных программ.

Никакая система задач, какой бы хорошей она ни была, никакие тренинги памяти, внимания и т. п. не дают того эффекта, который возникает в случае, если учащиеся осознают необходимость решения тех или иных задач, если у них появляется острая необходимость к преодолению интеллектуальных трудностей.

Содержание обучения, представленное в программе «Программирование на языке «Python», позволяет вести обучение в режиме актуального познания. Практическая направленность курса на создание внешних образовательных продуктов – блок-схем, алгоритмов, программ – способствует выявлению фактов, которые невозможно объяснить на основе имеющихся у учащихся знаний.

Направленность программы: техническая

Актуальность состоит в том, что активизация познавательного процесса позволяет учащимся более полно выражать свой творческий потенциал и реализовывать собственные идеи в изучаемой области знаний, создаёт предпосылки по применению информационных компетенций в других учебных курсах, а также способствует возникновению дальней мотивации, направленной на освоение профессий, связанных с разработкой программного обеспечения.

Курс служит средством внутривузовской специализации в области новых информационных технологий, что способствует созданию дополнительных условий для проявления индивидуальных образовательных интересов учащихся, их дальнейшей профессиональной ориентации.

Новизна программы.

Данная программа предполагает формирование системного инженерного мышления обучающихся, что позволяет не только овладевать широкой областью знаний и набором поликомпетенций, но и решать творческие, проектные задачи.

Цель и задачи программы Основной целью данного учебного курса является ознакомление слушателя с объектно-ориентированным языком программирования

Python, с возможностями, синтаксисом языка, технологией и методами программирования в среде Python, обучение практическим навыкам программирования на языке Python для решения типовых задач математики и информатики, а также при разработке простейших игр.

Содержание программы направлено на формирование научного мировоззрения, освоение методов научного познания мира, развитие 4 исследовательских, прикладных, конструкторских способностей обучающихся, с наклонностями в области точных наук и технического творчества (сфера деятельности «человек-машина»). Реализация поставленной цели предусматривает решение следующих **задач**:

- развитие интереса учащихся к изучению программирования; знакомство учащихся с основами программирования в среде Python;
- овладение базовыми понятиями теории алгоритмов при решении математических задач;
- формирование 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;
- формирование навыков грамотной разработки программ;
- формирование практических навыков решения прикладных задач;
- формирование практических навыков разработки игр

Отличительные особенности программы является его направленность на формирование у учащихся навыков поиска собственного решения поставленной задачи, составления алгоритма решения и его реализации с помощью средств программирования.

Педагогическая целесообразность программы выражена в подборе интерактивных и практикоориентированных форм занятий, способствующих формированию основных компетенций (информационных, коммуникативных, компетенций личного развития)

Адресат программы

Дополнительная общеразвивающая программа «Программирование на языке Python» рассчитана на детей 13–16 лет, проявляющих интерес к информационно-коммуникационным технологиям. Учащиеся объединения являются разными по возрасту и социальному статусу. Объединение комплектуется на основании заявлений законных представителей учащихся (самих учащихся с 14 лет). Группы формируются из школьников разного возраста на добровольной внеконкурсной основе.

Программа составлена с учётом индивидуальных и возрастных особенностей учащихся. Психолого-педагогические особенности учащихся определяют и методы индивидуальной работы педагога с каждым из них, темпы прохождения образовательного маршрута.

Возрастные и психологические особенности учащихся. Содержание программы учитывает возрастные и психологические особенности детей 13–16 лет, которые определяют выбор форм проведения занятий с обучающимися. Дети этого возраста отличаются внутренней уравновешенностью, стремлением к активной практической деятельности, поэтому основной формой проведения занятий выбраны практические занятия. Ребят также увлекает совместная, коллективная деятельность, так как резко возрастает значение коллектива, общественного мнения, отношений со сверстниками, оценки поступков и действий ребёнка со стороны не только старших, но и сверстников. Ребёнок стремится завоевать в их глазах авторитет, занять достойное место в коллективе. Поэтому в программу

включены практические занятия соревновательного характера, которые позволяют каждому проявить себя и найти своё место в детском коллективе. Также следует отметить, что дети данной возрастной группы характеризуются такими психическими процессами, как изменение структуры личности и возникновение интереса к ней, развитие абстрактных форм мышления, становление более осознанного и целенаправленного характера деятельности, проявление стремления к самостоятельности и независимости, формирование самооценки. Эти процессы позволяют положить начало созданию начального профессионального самоопределения обучающихся.

Объем и срок освоения программы

Дополнительная общеразвивающая программа технической направленности рассчитана на 2 года, 68 часов. Занятия проводятся 1 раз в неделю.

Форма обучения: очная.

Особенности организации образовательного процесса

Режим занятий: Занятия проводятся 1 раз в неделю. Режим организации занятий по данной дополнительной общеобразовательной программе определяется календарным учебным графиком и соответствует нормам, утвержденным «СанПин к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей» (СанПин 2.4.43172 -14).

Планируемые результаты изучения программы

В результате изучения дисциплины получают дальнейшее развитие личностные регулятивные, коммуникативные и познавательные-универсальные учебные действия, учебная (общая и предметная) и общепользовательская ИКТ–

компетентность обучающихся, составляющая психолого-педагогическую, инструментальную основу формирования способности и готовности к освоению систематических знаний, к их самостоятельному пополнению, переносу и интеграции, способности к сотрудничеству и коммуникации, решению личностно и социально значимых проблем и воплощению решений в практику, способности к самоорганизации, саморегуляции и рефлексии. В ходе изучения данного курса в основном формируются и получают развитие следующие метапредметные результаты:

- умение самостоятельно планировать пути достижения цели, в том числе альтернативные, осознанно выбирать наиболее эффективные способы решения задач;
- умение соотносить свои действия с планируемыми результатами, осуществлять контроль всей деятельности в процессе достижения результата, корректировать свои действия в соответствии с изменяющейся ситуацией;
- умение оценивать правильность выполнения учебной задачи, собственные возможности ее решения; владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности; 5 умение организовывать сотрудничество и совместную деятельность с учителем и сверстниками;
- работать индивидуально в группе: находить общее решение и разрешать конфликты на основе согласования позиций и учета интересов; формулировать, аргументировать и отстаивать свое мнение;
- формирование и развитие компетентности в области использования информационно-коммуникационных технологий.

Вместе с тем, вносятся существенный вклад в развитие **личностных**

результатов:

- формирование ответственного отношения к обучению, готовности и способности обучающихся к саморазвитию и самообразованию на основе мотивации к обучению и познанию;
- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками, взрослыми в процессе образовательной, общественно-полезной, учебноисследовательской, творческой и других видах деятельности.

В части развития **предметных результатов** наибольшее влияние изучение курса оказывает на:

- формирование знаний, умений и навыков при решении задач информатики и программирования разных видов;
- приобретение опыта использования информационных технологий в индивидуальной и коллективной учебной и познавательной, в том числе проектной деятельности.
- формирование информационной и алгоритмической культуры;
- формирование представления о компьютере как универсальном устройстве обработки информации;
- развитие основных навыков и умений использования компьютерных устройств;
- формирование представления об основных изучаемых понятиях: информация, алгоритм, модель – и их свойства;
- развитие алгоритмического мышления, необходимого для профессиональной деятельности в современном обществе;
- развитие умений составить и записать алгоритм для конкретного исполнителя;
- формирование знаний об алгоритмических конструкциях,

- логических значениях и операциях;
- знакомство с одним из языков программирования и основными алгоритмическими структурами – линейной, условной и циклической;
- владение системой базовых знаний, отражающих вклад информатики в формирование современной научной картины мира;
- овладение понятием сложности алгоритма, знание основных алгоритмов обработки числовой и текстовой информации, алгоритмов поиска и сортировки;
- владение универсальным языком программирования высокого уровня Python, представлениями о базовых типах данных и структурах данных;
- умением использовать основные управляющие конструкции;
- владение навыками и опытом разработки программ в среде программирования Python, включая тестирование и отладку программ;
- владение элементарными навыками формализации прикладной задачи и документирования программ;
- формирование умения работать с библиотеками программ;
- получение опыта использования компьютерных средств представления и анализа данных.

В результате изучения дисциплины учащиеся **должны знать:**

- основные типы алгоритмов;
- иметь представление о структуре программы, основы программирования на языках высокого уровня;
- базовые алгоритмические конструкции;
- содержание этапов разработки программы: алгоритмизация кодирование-отладка тестирование;
- дополнительные возможности языка Python для выражения различных алгоритмических ситуаций;

- алгоритмы и программы на языке Python решения простых, сложных и нестандартных задач в математической области;
- основы разработки простых игр в системе программирования Python.

Учащиеся **должны уметь:**

- записывать алгоритмические структуры на языке программирования Python;
- использовать Python для решения задач из области математики, физики;
- строить алгоритмы методом последовательного уточнения (сверху вниз), изображать эти алгоритмы в виде блок-схем;
- использовать основные алгоритмические приемы при решении математических задач;
- решать простые, сложные и нестандартные задачи;
- создавать простые игры;
- анализировать текст чужих программ, находить в них неточности,
- оптимизировать алгоритм, создавать собственные варианты решения.

Формы аттестации: опрос, защита проекта.

Форма отслеживания и фиксации результатов: фронтальный и индивидуальный опрос, наблюдение, проектная деятельность, Практические занятия на заданную тему, отзывы детей и родителей.

Формы предъявления и демонстрации образовательных результатов: представление результатов своей работы перед другими учащимися школы, участие в межрайонных, городских конкурсах, олимпиадах.

Промежуточный контроль включает: выполнение практического задания.

Материально-техническое обеспечение:

Электронно-программное обеспечение:

- специализированные цифровые инструменты учебной деятельности (компьютерные программы);

Технические средства обучения:

- мультимедийный проектор;
- компьютеры с учебным программным обеспечением;
- демонстрационный экран;
- магнитная доска;
- интерактивная доска.

Информационное обеспечение: аудио-, видео-, интернет источники.

Кадровое обеспечение. Реализация программы обеспечивается педагогическими кадрами, имеющими среднее профессиональное образование или высшее образование, соответствующее направленности дополнительной общеобразовательной программы. Требования к педагогам дополнительного образования и преподавателям:

- среднее профессиональное образование – программы подготовки специалистов среднего звена или высшее образование – бакалавриат, направленность (профиль) которого, как правило, соответствует направленности дополнительной общеобразовательной программы;
- дополнительное профессиональное образование – профессиональная переподготовка, направленность (профиль) которой соответствует направленности дополнительной общеобразовательной программы; При отсутствии педагогического образования – дополнительное профессиональное педагогическое образование;

Дополнительная профессиональная программа может быть освоена после трудоустройства. Рекомендуется обучение по дополнительным профессиональным

программам по профилю педагогической деятельности не реже чем один раз в три года.

Учебный план. Модуль 1

№ п/п	Название раздела	Всего	Кол-во часов		Формы аттестации и контроля
			Теория	Практика	
Тема 1. Знакомство с языком Python					
	Общие сведения о языке	2	1	1	Опрос
	Практическая работа. Установка программы Python	2	1	1	Практическое задание
	Режимы работы	2	1	1	Опрос
	Практическая работа 1.2. Режимы работы с Python	2	1	1	Практическое задание
Тема 2. Переменные и выражения					
	Урок 3. Переменные	2	1	1	Практическое задание
	Практическая работа 2.1. Работа со справочной системой	2	1	1	Практическое задание
	Практическая работа 2.2. Переменные	2	1	1	Практическое задание
	Урок 4. Выражения	2	1	1	Практическое задание

	Практическая работа 2.3. Выражения	2	1	1	Практическое задание
	Урок 5. Ввод и вывод	2	1	1	Практическое задание
	Урок 6. Задачи на элементарные действия с числами	2	1	1	Практическое задание
	Практическая работа 2.5. Задачи на элементарные действия с числами	2	1	1	Практическое задание
Тема 3. Условные выражения					
	Урок 7. Логические выражения и операторы	2	1	1	Практическое задание
	Практическая работа 3.1. Логические выражения	2	1	1	Практическое задание
	Урок 8. Условный оператор	2	1	1	Практическое задание
	Практическая работа 3.2. "Условный оператор"	2	1	1	Практическое задание
	Урок 9. Множественное ветвление	2	1	1	Практическое задание
	Практическая работа 3.3. Множественное ветвление	2	1	1	Практическое задание

	Урок 10. Реализация ветвления в языке Python	2	1	1	Практическое задание
	Практическая работа 3.4. "Условные операторы"	2	1	1	Практическое задание
	Итого	40	20	20	

Учебный план. Модуль 2.

№	Название раздела	Всего	Кол-во часов		Формы аттестации и контроля
			Теория	Практика	
п/п					
Тема 4. Циклы					
2.	Урок 12. Оператор цикла с условием	2	1	1	Практическое задание
3.	Практическая работа 4.1. "Числа Фибоначчи"	2	1	1	Практическое задание
	Урок 13. Оператор цикла for	2	1	1	Практическое задание
	Практическая работа 4.2. Решение задачи с циклом for	2	1	1	Практическое задание
	Урок 14. Вложенные циклы	2	1	1	Практическое задание
	Практическая работа 4.3. Реализация циклических алгоритмов	2	1	1	Практическое задание
	Урок 15. Случайные числа	2	1	1	Практическое задание
	Практическая работа 4.4. Случайные числа	2	1	1	Практическое задание

	Урок 16. Примеры решения задач с циклом	2	1	1	Практическое задание
	Практическая работа 4.5. Решение задач с циклом.	2	1	1	Практическое задание
	Урок 17. Творческая работа № 1. "Циклы"	2	1	1	Практическое задание
Тема 5. Функции					
	Урок 18. Создание функций	2	1	1	Практическое задание
	Урок 19. Локальные переменные	2	1	1	Практическое задание
	Урок 20. Примеры решения задач с использованием функций	2	1	1	Практическое задание
Итого		28	14	14	

Содержание учебной программы. Модуль 1.

1. Знакомство с языком Python .

Теория (4 часа). Техника безопасности на занятии. Понятие «алгоритм», «исполнитель», «язык программирования», «программа», «интерпретатор». История языка программирования Python и его возможности. Виды окон в VSCode: окно программы и окно консоли.

Практика (4 часа). Установка программы. Знакомство с интерфейсом. Сохранение и запуск python-программ в среде разработки VSCode. Установка модулей.

2. Переменные и выражения.

Теория (8 часов). Понятие «модуль». Модуль turtle. Импортирование модуля. Создание холста. Перемещение черепашки. Понятие «переменная». Правила именования переменных в языке Python. Оператор присваивания. Функция input(). Арифметические операции с помощью математических операторов +, -, *, /. Порядок выполнения операций. Понятие «выражение», «типы данных». Функции int() и str(). Понятие «строка». Создание строк. Переменные внутри строк. Операции со строками. Понятие «список». Создание списков. Добавление/удаление элементов в/из список/списка. Операции со списками. Правила синтаксиса Python: правило начала, правило порядка, правило регистра. Понятие функции. Функция print().

Практика (8 часов). Проект «Символьная графика». Создание определённого рисунка с помощью символов. Отработка функции print(). Проект «Аватар». В данном проекте отрабатывается функция input(), с помощью которой становится возможным ввести свои данные в программу и отобразить их. Проект «Сумматор». При написании данной программы отрабатываются математические операторы и функции int() и str(). Решение задач на отработку операций со строками и списками. Проект «Я рисую»: рисование изображения, состоящего из линий с помощью команд модуля на холсте. Проект «Любимые вещи»: создание списка любимых развлечений и любимых лакомств.

3. Условные выражения

Теория (8 часов). Логические операторы: and, or, not. Порядок выполнения операций. Переменные без значения – None. Понятие «условный оператор», «вложенные команды», «оператор сравнения». Конструкция if и её синтаксис.

Операторы сравнения: <, >, >=, <=, !=, ==. Структура программы. Конструкция if-else. Команды if и elif.

Практика (8 часов). Проект «Калькулятор»: создание приложения по определенным условиям. Решение задач на обработку условного оператора и операторов сравнения.

Содержание учебной программы. Модуль 2.

1. Циклы.

Теория (11 часов). Понятие «цикл», «цикл со счётчиком». Конструкция for и её синтаксис. Понятие «цикл с предусловием». Конструкция while и её синтаксис. Заикливание и выход из цикла с помощью команды break. Виды циклов и их конструкции. Вложенные циклы. Циклы в циклах. Случайные числа. Функция randrange. Функция random. Примеры решения задач с циклом

Практика (11 часов). Проект «Таблица умножения»: создание приложения по определенным условиям. Проект «Бомба взорвалась!». Написание программы по определенным условиям. Решение задач на применение циклов for и while. Решение задач на обработку понятия «функция», её строение и синтаксис.

2. Функции

Теория (3 часа). 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. Функции: abs, bool, dir, eval, exec, float, int, len, max, min, range, sum.

Практика (3 часа). Решение задач на обработку функций.

Методическое обеспечение программы

Методическое обеспечение программы

1. Учебно-наглядные пособия:

- схемы, бланки заданий;
- схемы программы;
- мультимедиаобъекты по темам курса;

2. Оборудование:

- компьютер;

Электронно-программное обеспечение:

- специализированные цифровые инструменты учебной деятельности (компьютерные программы);

Технические средства обучения:

- мультимедийный проектор;
- компьютер с выходом в сеть Интернет;
- демонстрационный экран;
- магнитная доска;
- **Форма организации учебного занятия:**
- лекция,
- учебная игра,
- тематические задания по группам,
- индивидуальная работа;
- проект.

Материально-техническое обеспечение

1. Кабинет информатики;
2. Ноутбук для каждого ученика;
3. Проектор;
4. Экран для проектора;

Алгоритм учебного занятия

Учебное занятие любого типа как модель можно представить в виде последовательности следующих этапов: организационного, проверочного, подготовительного, основного, контрольного, рефлексивного (самоанализ), итогового, информационного. Каждый этап отличается от другого сменой видов деятельности, содержанием и конкретной задачей. Основанием для выделения этапов может служить процесс усвоения знаний, который строится как смена видов деятельности учащихся: восприятие – осмысление – запоминание – применение – обобщение – систематизация.

1 этап: организационный.

Задача: подготовка детей к работе на занятии.

Содержание этапа: организация начала занятия, создание психологического настроя на учебную деятельность и активизация внимания.

2 этап: проверочный.

Задача: установление правильности и осознанности выполнения домашнего задания (если было), выявление пробелов и их коррекция.

Содержание этапа: проверка усвоения знаний предыдущего занятия.

3 этап: подготовительный (подготовка к новому содержанию).

Задача: обеспечение мотивации и принятие детьми цели учебно-познавательной деятельности.

Содержание этапа: сообщение темы, цели занятия и мотивация деятельности детей (к примеру, эвристический вопрос, познавательная задача, проблемное задание детям).

4 этап: основной.

В качестве основного этапа могут выступать следующие:

1) Усвоение новых знаний и способов действий.

Задача: обеспечение восприятия, осмысления и первичного запоминания связей и отношений в объекте изучения. Целесообразно при усвоении новых знаний использовать задания и вопросы, которые активизируют познавательную деятельность детей.

2) Первичная проверка понимания.

Задача: установление правильности и осознанности усвоения нового учебного материала, выявление неверных представлений и их коррекция. Применяют пробные практические задания, которые сочетаются с объяснением соответствующих правил или обоснованием

3) Закрепление знаний и способов действий.

Задача: обеспечение усвоения новых знаний и способов действий. Применяют тренировочные упражнения, задания, которые выполняются самостоятельно детьми.

4) Обобщение и систематизация знаний.

Задача: формирование целостного представления знаний по теме. Распространенными способами работы являются беседа и практические задания.

5 этап: контрольный.

Задача: выявление качества и уровня овладения знаниями, их коррекция. Используются виды устного и письменного опроса, вопросы и задания различного уровня сложности (репродуктивного, творческого, поисково-исследовательского).

6 этап: Итоговый.

Перечень оборудования для работ:

- компьютер
- интернет.

Для успешной реализации программы применяются следующие **дидактические материалы**:

- схемы

- Эскизы.

Список литературы

Литература для педагога

1. Федеральный закон № 273-ФЗ от 29.12.2012 «Об образовании в Российской Федерации».
2. СанПиН 2.4.4.3172-14 «Санитарно-эпидемиологические требования к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей» (Главный государственный санитарный врач РФ, Постановление от 4 июля 2014 года №41).
3. Бондаренко А.М. Проектная деятельность – запуск механизма развития личности ребёнка // Эксперимент и инновации в школе, 2011. – №3.
4. Гура В.В. Теоретические основы педагогического проектирования личностно-ориентированных образовательных ресурсов и сред. – Ростов н/Д: Изд-во ЮФУ, 2007.
5. Новожилова М.М. Как корректно провести учебное исследование: от замысла к открытию / М. М. Новожилова, С.Г. Воровщиков, И.В. Таврель; науч. ред. Т.И. Шамова. – 3-е изд. – М.: 5 за знания, 2008.
6. Страхова И.А. Проектная деятельность как один из способов формирования универсальных учебных действий // Методист. – 2012. – № 4.
7. Федеральный портал «Российское образование». Каталог образовательных Интернет-ресурсов [Электронный ресурс]. – режим доступа: www.edu.ru.

Литература для учащихся

1. Абрамов С.А, Гнездилова Г.Г., Капустина Е.Н., М.И. Селюн. Задачи по программированию. – М.: Наука, 1998.
2. Аллен И. Голуб. С и С++. Правила программирования. – М.: БИНОМ, 1996.

3. Долинский М.С. Решение сложных и олимпиадных задач по программированию - Учебное пособие - М.: – 2006.
4. Майк МакГрат «Программирование на Python для начинающих» Эксмо, 2015.
5. Россум Г., Дж. Дрейк Ф.Л., Откидач Д.С. Язык программирования Python. 2001.
6. Сэнд У., Сэнд К. «Hello World! Занимательное программирование на языке Python» - М.: – 2016.
7. Федоров Д. Ю. Основы программирования на примере языка Python. //Учебное пособие. – Санкт-Петербург: 2016.
8. Любанович Б. Простой Python. Современный стиль программирования / Б. Любанович. - СПб.: Питер, 2016.
9. Саммерфильд М. Python на практике / М. Саммерфильд, пер. А.А. Слинкин – М.: ДМК-Пресс, 2014.

Литература для родителей

1. Баркан А. Практическая психология для родителей или Как научиться понимать своего ребенка. М. 2000
2. Валеев Р. Дело по душе и жизненное самоопределение школьника // Воспитание школьников. – 2000. – № 6. 13
3. Макаренко А.С. Книга для родителей // Соч.: В 7 т. – М., АПН РСФСР, 1957.- Т IV .
4. Моргун, Д. В. Дополнительное образование детей в вопросах и ответах / Д.В. Моргун, Л.М. Орлова. - М.: ЭкоПресс, 2016.

Календарный учебный график. Модуль 1.

№ п/п	Месяц	Число	Время проведения занятия	Форма занятия	Кол-во часов	Тема занятия	Место проведения	Форма контроля
Тема 1. Знакомство с языком Python								
			2 половина дня		2	Общие сведения о языке	к. 313	опрос
			2 половина дня		2	Практическая работа. Установка программы Python	к. 313	Практическое занятие
			2 половина дня		2	Режимы работы	к. 313	Практическое занятие
			2 половина дня		2	Практическая работа 1.2. Режимы работы с Python	к. 313	Практическое занятие
Тема 2. Переменные и выражения								
			2 половина дня		2	Урок 3. Переменные	к. 313	Практическое занятие
			2 половина дня		2	Практическая работа	к. 313	Практическое

						2.1. Работа со справочной системой		занятие
			2 половина дня		2	Практическая работа 2.2. Переменные	к. 313	Практическое занятие
			2 половина дня		2	Урок 4. Выражения	к. 313	Практическое занятие
			2 половина дня		2	Практическая работа 2.3. Выражения	к. 313	Практическое занятие
			2 половина дня		2	Урок 5. Ввод и вывод	к. 313	Практическое занятие
			2 половина дня		2	Урок 6. Задачи на элементарные действия с числами	к. 313	Практическое занятие
			2 половина дня		2	Практическая работа 2.5. Задачи на элементарные действия с числами	к. 313	Практическое занятие
Тема 3. Условные выражения								
			2 половина дня		2	Урок 7. Логические	к. 313	Практическое

						выражения и операторы		занятие
			2 половина дня		2	Практическая работа 3.1. Логические выражения	к. 313	Практическое занятие
			2 половина дня		2	Урок 8. Условный оператор	к. 313	Практическое занятие
			2 половина дня		2	Практическая работа 3.2. "Условный оператор"	к. 313	Практическое занятие
			2 половина дня		2	Урок 9. Множественное ветвление	к. 313	Практическое занятие
			2 половина дня		2	Практическая работа 3.3. Множественное ветвление	к. 313	Практическое занятие
			2 половина дня		2	Урок 10. Реализация ветвления в языке Python	к. 313	Практическое занятие
			2 половина дня		2	Практическая работа	к. 313	Практическое

						3.4. "Условные операторы"		занятие
--	--	--	--	--	--	---------------------------	--	---------

Календарный учебный график. Модуль 2.

№ п/п	Месяц	Число	Время проведения занятия	Форма занятия	Кол-во часов	Тема занятия	Место проведения	Форма контроля
	Тема 4. Циклы							
			2 половина дня		2	Урок 12. Оператор цикла с условием	к. 313	опрос
			2 половина дня		2	Практическая работа 4.1. "Числа Фибоначчи"	к. 313	Практическое занятие
			2 половина дня		2	Урок 13. Оператор цикла for	к. 313	Практическое занятие
			2 половина дня		2	Практическая работа 4.2. Решение задачи с	к. 313	Практическое занятие

						циклом for		
			2 половина дня		2	Урок 14. Вложенные циклы	к. 313	Практическое занятие
			2 половина дня		2	Практическая работа 4.3. Реализация циклических алгоритмов	к. 313	Практическое занятие
			2 половина дня		2	Урок 15. Случайные числа	к. 313	Практическое занятие
			2 половина дня		2	Практическая работа 4.4. Случайные числа	к. 313	Практическое занятие
			2 половина дня		2	Урок 16. Примеры решения задач с циклом	к. 313	Практическое занятие
			2 половина дня		2	Практическая работа 4.5. Решение задач с циклом.	к. 313	Практическое занятие
			2 половина дня		2	Урок 17. Творческая работа № 1. "Циклы"	к. 313	Практическое занятие

Тема 4. Функции

			2 половина дня		2	Урок 18. Создание функций Практическая работа 5.1. Создание функций	к. 313	Практическое занятие
			2 половина дня		2	Урок 19. Локальные переменные Практическая работа 5.2. Локальные переменные	к. 313	Практическое занятие
			2 половина дня		2	Урок 20. Примеры решения задач с использованием функций	к. 313	Практическое занятие

Входящий контроль

Низкий уровень: 0-70% выполненных заданий;

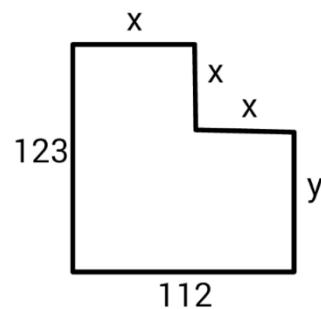
Средний уровень: 70-85% правильно выполненных заданий;

Высокий уровень: 85-100% правильно выполненных заданий.

Для учащихся – 14-17 лет

Фамилия, Имя	
---------------------	--

1. Из прямоугольника вырезали квадрат со стороной x , получилась фигура как на рисунке. Чему равна сумма цифр y ?



2. В алфавите племени мумба-юмба 32 буквы. Любое слово в языке этого племени состоит из пяти букв и должно одинаково читаться справа налево и слева направо, при этом первые две буквы слова обязательно различаются, а третья совпадает с пятой.

Каково максимальное количество слов в этом языке?

3. Катя наклеила на рулет тонкие поперечные кольца трёх разных цветов. Если разрезать по серым кольцам, получится 25 кусков рулета, если по малиновым — 47 кусков, а если по зеленым — 31 кусок.

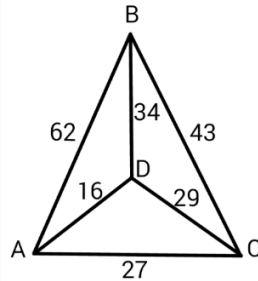
Сколько кусков рулета получится, если разрезать по кольцам всех трёх цветов?

Примечания

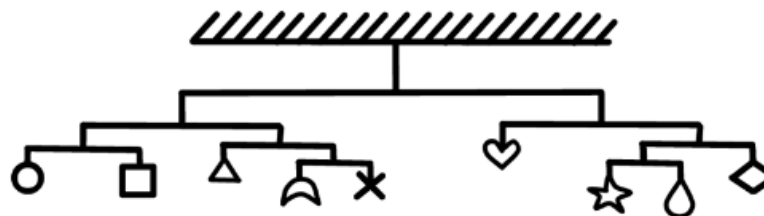
Учтите, что наклеить кольцо одного цвета на кольцо другого нельзя.

4. На рисунке показано расположение городов А, В, С и D и расстояния между ними. Турист выходит из города В и собирается посетить остальные города, побывав в каждом по разу.

Какова наименьшая возможная длина маршрута, если он хочет закончить свой путь в том же городе?



5. Фигурки, общей массой 432 грамма, при помощи невесомых нитей и планок собрали в конструкцию, изображённую на рисунке. Оказалось, что все её части находятся в равновесии. Сколько весит ромб?



6. Люди переезжают в города, за год численность людей удваивается.

Если люди заселят весь город за 12 лет, то сколько лет понадобится, чтобы занять лишь половину города?

7. Сравните пары слов. Сколько среди них полностью идентичных?

O/Sanmarco

O/Samnarco

Ф.Wagonerrte

Ф.Wagonertre

A.S.Schmetterling

A.S.Schnetterling

N.V.Murfreesboroque

N.V.Munfreesboroque

P.S.Splendoursec

P.S.Sqlendoursec

Семь человек выясняли, какой сегодня день недели.

Первый сказал: «Послезавтра – воскресенье».

Второй: «Вчера был понедельник».

Третий: «Завтра будет суббота».

Четвертый: «Завтра будет среда».

Пятый: «Вчера был четверг».

Шестой: «Позавчера было воскресенье».

Седьмой: «Позавчера была среда».

Какой сегодня день недели, если трое ошибаются?

8. Вам предложены несколько высказываний и следствие из них (выделено жирным).

Согласны ли Вы с этим следствием?

1. Все клёны — растения.

2. Некоторые растения быстро желтеют.

Значит, некоторые клёны быстро желтеют.

 Да Нет

9. Гусеница прогрызает яблоко диаметром 6 сантиметров насквозь за 16 секунд, вылезая снаружи полностью.

Известно, что середину яблока она начинает грызть уже через 6 секунд после начала пути.

Какова длина гусеницы в сантиметрах?

Для учащихся 11-13 лет

Фамилия, Имя.	
---------------	--

1. Назовите два числа, у которых количество цифр равно количеству букв, составляющих название каждого из этих чисел.

--	--

2. Собака была привязана к десятиметровой веревке, а прошла двести метров. Как ей это удалось?

--

3. Есть дорога, по которой может проехать только одна машина. По дороге едут две машины: одна с горы, другая под гору. Как им разъехаться?

--

4. Как с помощью только одной палочки образовать на столе треугольник?

--

5. В каком месяце болтливая девочка говорит меньше всего?

--

6. Что становится на треть больше, если его поставить вверх ногами?

--

7. Представьте себе, что вы кондуктор. Поезд везет сто вагонов, в каждом вагоне 10 купе, в каждом купе 4 пассажира. Сколько лет кондуктору?

8. Юра разрезал огромную пиццу на 10 кусков. Затем он взял один из кусков и разрезал его еще на 10. После этого из имеющихся кусков он выбрал два и разрезал каждый из них на 10.

Вопрос: Сколько кусков пиццы получилось у Юры.

9. В компьютерной игре нужно победить монстра. Изначально у Юры было только 9 выстрелов. Но за каждое попадание он получал дополнительно еще 3 выстрела.

Вопрос: Сколько раз Юра попал в цель, если всего он выстрелил 30 раз, израсходовав все выстрелы?

Итоговый контроль
Диагностическая карта

№/ п	Имя	Входящая Диагностика	Промежуточ- ная диагностика	Итоговая диагностика
1				
2				
3+				
	Итого в %	Н С В		Н С В

Результативность отслеживается с помощью карты наблюдений, анализа участия детей в совместной продуктивной деятельности, разработки и защите творческих проектов. Усвоение программы возможно по 3-м уровням: низкий (Н), средний (С), высокий (В).

Низкий уровень

Учащиеся должны знать/ понимать:

- технику безопасного поведения во время занятий;
- правила поведения в общественных местах,
- понятие программы Python; общую структуру программы;

- основные типы данных;
- оператор присваивания;
- назначение условного оператора;
- способ записи условного оператора;
- логический тип данных;
- логические операторы or, and, not;
- основные циклы с условием;
- основные правила записи циклов условием;
- формат записи цикла с параметром;
- понятие функции;
- основные принципы структурного программирования;
- понятие локальных переменных подпрограмм;
- способ передачи параметров.
- назначение строкового типа данных;
- операторы для работы со строками;
- операции со строками;
- способ описания списка;
- способ описания кортежа;
- способ описания словаря;
- основные операции, выполняемые со списками, кортежами и словарями;
- понятие множества;
- способы описания множества;
- операторы работы с множествами.

- что такое стиль программирования;
- правила именования объектов;
- основные рекомендации при написании программ.

Учащиеся должны уметь:

- понимать учебную задачу, сохранять ее содержание в процессе ее выполнения под руководством педагога;
- работать в паре, малой группе;
- выполнить установку программы под руководством педагога;
- выполнить простейшую программу в интерактивной среде;
- написать комментарии в программе под руководством педагога;
- решать задачи на элементарные действия с числами;
- использовать условный оператор;
- определять вид цикла, наиболее удобный для решения поставленной задачи;
- использовать цикл с условием под руководством педагога;
- определять целесообразность применения и использования цикла с параметром для решения поставленной задачи по наводящим вопросам педагога;
- создавать и использовать основные функции;
- описывать и соединять строки;
- находить подстроку в строке с помощью педагога;
- находить количество слов в строке;
- вводить и выводить элементы списка под руководством педагога;
- приводить примеры использования вложенных списков (матриц) по наводящим вопросам педагога;
- описывать множества под руководством педагога;
- определять принадлежность элемента множеству по наводящим вопросам;

- определять вид ошибок и находить ошибки в программе под руководством педагога;
- составлять элементарные алгоритмы для решения задач;
- реализовывать алгоритмы на компьютере в виде программ, написанных на языке Python под руководством педагога.

Средний уровень

Учащиеся должны знать/ понимать:

- основные положения техники безопасности на занятиях, правила поведения в общественных местах, правила дорожной безопасности;
- основные приемы взаимодействия в группе сверстников;
- понятие программы;
- структуру программы на Python;
- режимы работы с Python.
- типы данных;
- целые, вещественные типы данных и операции над ними;
- оператор присваивания;
- назначение условного оператора;
- способ записи условного оператора;
- логический тип данных;
- логические операторы or, and, not;
- циклы с условием и их виды;
- назначение и особенности использования цикла с параметром;
- формат записи цикла с параметром;
- примеры использования циклов различных типов.
- понятие функции;

- основные способы описания функции;
- принципы структурного программирования;
- понятие локальных переменных подпрограмм;
- понятие формальных и фактических параметров подпрограмм;
- способ передачи параметров.
- назначение строкового типа данных;
- операторы для работы со строками;
- процедуры и функции для работы со строками;
- операции со строками;
- сложные типы данных;
- способ описания списка;
- способ доступа к элементам списка;
- способ описания кортежа;
- способ описания словаря;
- операции, выполняемые со списками, кортежами и словарями;
- понятие множества;
- способы описания множества;
- операторы работы с множествами.
- что такое стиль программирования;
- правила именования объектов;
- основные рекомендации при написании программ;
- основные шаги работы над проектом, его презентации.

Учащиеся должны уметь:

- уважительно относиться к преподавателям и сверстникам;
- применять некоторые приемы логического (абстрактное) мышления;
- концентрировать внимание на одном или двух объектах;
- понимать причины успеха/неуспеха с помощью анализа педагога;
- выполнить установку программы;
- выполнить простейшую программу в интерактивной среде;
- написать комментарии в программе;
- решать задачи на элементарные действия с числами;
- использовать условный оператор;
- создавать сложные условия с помощью логических операторов;
- определять вид цикла, наиболее удобный для решения поставленной задачи;
- использовать цикл с условием;
- определять целесообразность применения и использования цикла с параметром для решения поставленной задачи;
- создавать и использовать функции;
- использовать механизм параметров для передачи значений;
- описывать строки;
- соединять строки;
- находить длину строки;
- вырезать часть строки;
- находить подстроку в строке;
- находить количество слов в строке;
- описывать списки;
- вводить элементы списка;

- выводить элементы списка;
- выполнять поиск элемента в списке, поиск минимума и максимума, нахождение суммы элементов списка;
- использовать вложенные списки;
- приводить примеры использования вложенных списков (матриц);
- описывать множества;
- определять принадлежность элемента множеству;
- вводить элементы множества;
- выводить элементы множества.
- определять вид ошибок и находить ошибки в программе.
- составлять алгоритмы для решения задач;
- реализовывать алгоритмы на компьютере в виде программ, написанных на языке Python;
- отлаживать и тестировать программы, написанные на языке Python;
- понимать и сохранять цели и задачи учебной деятельности в ходе занятия;
- планировать свою деятельность с помощью взрослого;
- сотрудничать с взрослыми и сверстниками в процессе выполнения учебной задачи;
- понимать причины успеха/неуспеха учебной деятельности под руководством взрослого;
- делать выводы в ходе полученных заданий; выстраивать логические цепи рассуждений под руководством педагога;
- выражать творческие идеи, разработать творческий проект на основе образца;
- конструктивно взаимодействовать в составе группы в ходе работы над проектом.

Высокий уровень

Учащиеся должны знать/ понимать:

- основные положения техники безопасности на занятиях, правила поведения в общественных местах, правила дорожной безопасности,

правила поведения во время чрезвычайных происшествий;

- приемы конструктивного взаимодействия в группе сверстников;
- понятие программы;
- структура программы на Python;
- режимы работы с Python.
- общую структуру программы;
- типы данных;
- целые, вещественные типы данных и операции над ними;
- оператор присваивания;
- назначение условного оператора;
- способ записи условного оператора;
- логический тип данных;
- логические операторы or, and, not;
- циклы с условием и их виды;
- правила записи циклов с условием;
- назначение и особенности использования цикла с параметром;
- формат записи цикла с параметром;
- примеры использования циклов различных типов.
- понятие функции;
- способы описания функции;
- принципы структурного программирования;
- понятие локальных переменных подпрограмм;
- понятие формальных и фактических параметров подпрограмм;

- способ передачи параметров.
- назначение строкового типа данных;
- операторы для работы со строками;
- процедуры и функции для работы со строками;
- операции со строками;
- сложные типы данных;
- способ описания списка;
- способ доступа к элементам списка;
- способ описания кортежа;
- способ описания словаря;
- операции, выполняемые со списками, кортежами и словарями;
- понятие множества;
- способы описания множества;
- операторы работы с множествами.
- что такое стиль программирования;
- правила именования объектов;
- основные рекомендации при написании программ;
- правила и этапы работы над проектом;
- приемы успешной презентации проекта.

Учащиеся должны уметь:

- выполнить установку программы;
- выполнить простейшую программу в интерактивной среде;
- написать комментарии в программе;

- решать задачи на элементарные действия с числами;
- использовать условный оператор;
- создавать сложные условия с помощью логических операторов;
- определять вид цикла, наиболее удобный для решения поставленной задачи;
- использовать цикл с условием;
- определять целесообразность применения и использования цикла с параметром для решения поставленной задачи;
- создавать и использовать функции;
- использовать механизм параметров для передачи значений;
- описывать строки;
- соединять строки;
- находить длину строки;
- вырезать часть строки;
- находить подстроку в строке;
- находить количество слов в строке;
- описывать списки;
- вводить элементы списка;
- выводить элементы списка;
- выполнять поиск элемента в списке, поиск минимума и максимума, нахождение суммы элементов списка;
- использовать вложенные списки;
- приводить примеры использования вложенных списков (матриц);
- описывать множества;
- определять принадлежность элемента множеству;
- вводить элементы множества;

- выводить элементы множества.
- определять вид ошибок и находить ошибки в программе.
- составлять алгоритмы для решения задач;
- реализовывать алгоритмы на компьютере в виде программ, написанных на языке Python;
- отлаживать и тестировать программы, написанные на языке Python;
- понимать учебную задачу, анализировать достижение результата;
- делать выводы в ходе полученных заданий; самостоятельно выстраивать логические цепи рассуждений;
- понимать причины успеха/неуспеха, конструктивно принимать причины неуспеха, исправлять ошибки с минимальным участием педагога;
- планировать свою деятельность, находить оригинальные способы выполнения поставленной творческой задачи;
- создать проектировочную команду и организовать ее деятельность;
- разработать творческий проект по предложенной или самостоятельно выбранной теме в составе творческой группы;
- самостоятельно подготовить презентацию и защитить проект.

Вопросы для промежуточного контроля по усвоению материала

(Промежуточный контроль проводится в последнюю неделю декабря)

Составьте выражение для вычисления в интерпретаторе Python 3 и вставьте в поле ответа результат вычисления:

$11111 \cdot 1111111$

— произведение чисел **11111 (5 единиц)** и **1111111 (7 единиц)**

Запишите число **1.2345e3** в виде **десятичной** дроби.

Составьте и запишите выражение для вычисления:

2014.0^14(Возвестив 14 степень)

Обратите внимание на запись числа: это вещественное число.

Приведите к целому типу число 2.99

Расставьте скобки в выражении

a and b or not a and not b

в соответствии с порядком вычисления выражения (приоритетом операций).

Всего потребуется 5 пар скобок (внешние скобки входят в их число).

Найдите результат выражения для заданных значений *a* и *b*

Учитывайте регистр символов при ответе.

a = True

b = False

a and b or not a and not b

Отметьте выражения, значения которых равны True:

- "239" < "30" and 239 < 30
- "239" < "30" and 239 > 30
- "239" > "30" and 239 < 30

- "239" > "30" and 239 > 30

Укажите результат выражения:

"123" + "42"

Какое значение будет у переменной *i* после выполнения фрагмента программы?

```
i = 0  
while i <= 10:  
    i = i + 1  
    if i > 7:  
        i = i + 2
```

Сколько итераций цикла будет выполнено в этом фрагменте программы?

```
i = 0  
while i <= 10:  
    i = i + 1  
    if i > 7:  
        i = i + 2
```

Сколько всего знаков * будет выведено после исполнения фрагмента программы:

```
i = 0  
while i < 5:  
    print('*')
```

```
i if i % 2 == 0:  
    print('**')  
if i>2:  
    print('***')  
i = i + 1
```

Определите, какое значение будет иметь переменная *i* после выполнения следующего фрагмента программы:

```
i = 0  
s = 0  
while i<10:  
    i = i + 1  
    s = s + i  
    if s >15:  
        break  
    i = i + 1
```

Определите, какое значение будет иметь переменная *i* после выполнения следующего фрагмента программы:

```
i = 0  
s = 0  
while i<10:
```

```
i = i + 1
```

```
    s = s + i
```

```
if s > 15:
```

```
    continue
```

```
i = i + 1
```

Задачи для промежуточного контроля по усвоению материала

Напишите простой калькулятор, который считывает с пользовательского ввода три строки: первое число, второе число и операцию, после чего применяет операцию к введённым числам ("первое число" "операция" "второе число") и выводит результат на экран.

Поддерживаемые операции: +, -, /, *, mod, pow, div, где
mod — это взятие остатка от деления,
pow — возведение в степень,
div — целочисленное деление.

Если выполняется деление и второе число равно 0, необходимо выводить строку "Деление на 0!".

Обратите внимание, что на вход программе приходят вещественные числа.

Жители страны Малевии часто экспериментируют с планировкой комнат. Комнаты бывают треугольные, прямоугольные и круглые. Чтобы быстро вычислять жилплощадь, требуется написать программу, на вход которой подаётся тип фигуры комнаты и соответствующие параметры, которая бы выводила площадь получившейся комнаты.

Для числа π в стране Малевии используют значение 3.14.

Формат ввода, который используют Малевийцы:

Треугольник

a

b

c

где a , b и c — длины сторон треугольника

прямоугольник

a

b

где a и b — длины сторон прямоугольника

круг

r

где r — радиус окружности

Напишите программу, которая получает на вход три целых числа, по одному числу в строке, и выводит на консоль в три строки сначала максимальное, потом минимальное,

после чего оставшееся число.

На ввод могут подаваться и повторяющиеся числа.

Паша очень любит кататься на общественном транспорте, а получая билет, сразу проверяет, счастливый ли ему попался. Билет считается счастливым, если сумма первых трех цифр совпадает с суммой последних трех цифр номера билета.

Однако Паша очень плохо считает в уме, поэтому попросил вас написать программу, которая проверит равенство сумм и выведет "Счастливый", если суммы совпадают, и "Обычный", если суммы различны.

На вход программе подаётся строка из шести цифр. **(Пример:123321)**

Выводить нужно только слово "Счастливый" или "Обычный", с большой буквы.

Напишите программу, которая считывает с консоли числа (по одному в строке) до тех пор, пока сумма введённых чисел не будет равна 0 и **сразу после этого** выводит сумму квадратов всех считанных чисел.

Гарантируется, что в какой-то момент сумма введённых чисел окажется равной 0, **после этого считывание продолжать не нужно.**

В примере мы считываем числа 1, -3, 5, -6, -10, 13; в этот момент замечаем, что сумма этих чисел равна нулю и выводим сумму их квадратов, не обращая внимания на то, что остались ещё не прочитанные значения.

Напишите программу, которая выводит часть последовательности 1 2 2 3 3 3 4 4 4 4 5 5 5 5 5 ... (число повторяется столько раз, чему равно). На вход программе передаётся неотрицательное целое число n — столько элементов последовательности должна отобразить программа. На выходе ожидается последовательность чисел, записанных через пробел в одну строку.

Например, если $n = 7$, то программа должна вывести 1 2 2 3 3 3 4.

Выведите таблицу размером $n \times n$, заполненную числами от **1 до n^2** по спирали, выходящей из левого верхнего угла и закрученной по часовой стрелке, как показано в примере (**здесь $n=5$**)