Муниципальное бюджетное общеобразовательное учреждение

«Дружбинская СОШ»

|  |  |  |
| --- | --- | --- |
| «Рассмотрено»На педагогическом советеПротокол №от «\_\_\_\_» \_\_\_\_\_\_2019 г. |  | «Утверждаю»Директор школы: Гасаналиев И.Г.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Приказ №\_\_\_\_\_\_\_от  «\_\_\_\_»\_\_\_\_\_2019  |

 

Программирование на языке Python

**Автор:** Рамазанов Р. М.

учитель информатики

**РАЗДЕЛ 1. КОМПЛЕКС ОСНОВНЫХ ХАРАКТЕРИСТИК ПРОГРАММЫ**

* 1. **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.

**Актуальность программы**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 внутрипрофильной специализации в области новых информационных технологий, что способствует созданию дополнительных условий для проявления индивидуальных образовательных интересов учащихся, их дальнейшей профессиональной ориентации.

**Отличительные особенности программы:**

Ключевой особенностью курса является его направленность на формирование у учащихся навыков поиска собственного решения поставленной задачи, составления алгоритма решения и его реализации с помощью средств программирования.

**Целесообразность** программы выражена в подборе интерактивных и практикоориентированных форм занятий, способствующих формированию основных компетенций (информационных, коммуникативных, компетенций личного развития и др.).

**Практическая значимость:**

В рамках предлагаемого курса «Программирования на языке Python» изучение основ программирования на языке Python− это не столько средство подготовки к будущей профессиональной деятельности, сколько формирование новых обще интеллектуальных умений и навыков: разделение задачи на этапы решения, построение алгоритма и др. Особую роль программирование служит для формирования мыслительных и психических процессов учащихся (внимание, память, логика), освоения приёмов умственных действий, самостоятельного нахождения и составления алгоритмов решения задач, умения строить модели, чётко и лаконично реализовывать этапы решения задач. Использование этих возможностей для формирования предметных и метапредметныхрезультатовучащихсяособенно важно, т.к. именно ониактивизирует процесс индивидуально-личностного становления учащихся.

**Адресат программы**

Дополнительная общеразвивающая программа «Программирование на языке Python» рассчитана на детей 12–17 лет, проявляющих интерес к нформационно-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.

**Особенности организации образовательного процесса**

* Объем программы: 68 часа.
* Срок реализации программы: 1 год.
* Режим обучения: 1 раза в неделю по 2часа.
* Форма обучения: очная.
* Особенности набора: свободный.
* Вид группы: среднешкольная.
* Состав группы: постоянный.

**Методические принципы программы**

* принцип развития: развитие индивидуальных способностей, общей культуры, навыков творческой продуктивной деятельности обучающихся;
* принцип демократизма: право каждого субъекта системы технического

творчества на выбор своей траектории развития;

* принцип дифференциации и индивидуализации образования: выявление и развитие способностей обучающихся в области технического творчества, обеспечение их развития в соответствии с потенциалом, индивидуальными возможностями и интересами;
* принцип культуросообразности: ориентация на потребности общества и личности учащихся, единство человека и социокультурной среды, адаптация детей к современным условиям жизни общества;
* принцип единства учебного и воспитательного процесса;
* принцип систематичности и последовательности: вначале используется репродуктивный и интерактивный методы усвоения знаний с постепенным введением проблемного метода обучения, метода проекта.

**Формы обучения и занятий**

Отборметодовобученияобусловленнеобходимостьюформироватьинформационнуюикоммуникативнуюкомпетентностиучащихся,реализовыватьличностно-ориентированноеобучение,направлятьихнасамостоятельноерешениеразнообразныхпроблем,развивать исследовательскиеитворческиеспособности.Решениеданныхзадачкроетсяворганизациидеятельностногоподходакобучению,впроблемномизложенииматериалапедагогом,впереходеотрепродуктивноговидаработксамостоятельным,поисково-исследовательс[кимвидамдеятельности](http://pandia.ru/text/category/vidi_deyatelmznosti/).Поэтомуосновнаяметодическаяустановкавданном курсе − обучениеучащихсянавыкамсамостоятельнойтворческойдеятельности.

Теория преподносится в форме беседы, эвристической беседы, презентации, обзора и т.п.

 Практические занятия проходят в форме тестирования, практикума, совместной продуктивной деятельности, моделирования и защиты проектов, проблемном изложении материала, с помощью которого дети сами решают познавательные задачи.

* 1. **Цель и задачи программы**

**Цель:** Способствовать формированию творческой личности, обладающей информационными компетенциями,владеющей базовыми понятиями теории алгоритмов, умеющейразрабатывать эффективные алгоритмы и реализовывать их в виде программы, написанной на языке программирования Python.

**Задачи**

1. **Образовательные:**
* познакомить с понятиями алгоритма, вычислимой функции, языка программирования;
* научить составлять и читать блок-схемы;
* сформировать навыки выполнения технологической цепочки разработки программ средствами языка программирования Python;
* объяснить основные конструкции языка программирования Python, позволяющие работать с простыми и составными типами данных (строками, списками, кортежами, словарями, множествами);
* научить применять функции при написании программ на языке программирования Python;
* научить отлаживать и тестировать программы, делать выводы о работе этих программ.
1. **Воспитательные:**
* воспитать уважительное отношение к преподавателям и сверстникам, культуру поведения во время занятий и совместной продуктивной деятельности;
* сформировать культуру занятий, направленную на воспитание личностных и социальных качеств;
1. **Развивающие:**
* развить познавательные процессы (внимание, восприятие, логическое мышление, память),
* развить креативность,
* развить способности к самореализации.
	1. **Содержание программы**

**Учебный план**

|  |  |  |  |
| --- | --- | --- | --- |
| **№****п/п** | **Наименованиетем** | **Количествочасов** | **Формы аттестации/ контроля** |
| **Всего** | **Теория** | **Практика** |  |
| **Инструктаж по технике безопасности в компьютерном классе** | **1** | **1** |  | Опрос |
| **1.** | **Раздел1. Знакомствосязыком Python** | **2** | **1** | **1** | Тестирование,решение практических задач |
| **2.** | **Раздел 2. Переменныеи выражения** | **5** | **1** | **4** | Тестирование,решение практических задач |
| 2.1 | Занятие3. ПеременныеПрактическаяработа: Работа со справочнойсистемой | 1 | 0 | 1 |
| 2.2 | Практическаяработа: Переменные | 1 |  | 1 |
| 2.3 | Занятие4. ВыраженияПрактическаяработа: Выражения | 1 | 0 | 1 |
| 2.4 | Занятие5. Ввод и вывод | 1 | 1 |  |
| 2.5 | Занятие6. Задачи на элементарныедействия с числамиПрактическаяработа2.5. Задачи на элементарные действия с числами | 1 | 0 | 1 |
| **3** | **Раздел 3. Условные предложения** | **14** | **3** | **11** | Тестирование,Решение практических задач |
| 3.1 | Занятие7. Логические выражения и операторы. Практическая работа: Логические выражения | 3 | 1 | 2 |
| 3.2 | Занятие8. Условный операторПрактическая работа: «Условный оператор» | 4 | 1 | 3 |
| 3.3 | Занятие9. Множественное ветвлениеПрактическаяработа: Множественное ветвление | 4 | 1 | 3 |
| 3.4 | Занятие10. Реализация ветвления в языке Python.Практическаяработа: «Условные операторы» | 3 |  | 3 |
| **4** | **Раздел 4. Циклы** | **23** | **6** | **17** | Тестирование,решение практических задач,творческая работа |
| 4.1. | Занятие12. Операторцикла сусловиемПрактическаяработа"Числа Фибоначчи" | 2 | 1 | 1 |
| 4.2. | Занятие13. Операторцикла forПрактическаяработаРешение задачи сциклом for. | 3 | 1 | 2 |
| 4.3. | Занятие14. ВложенныециклыПрактическаяработа:Реализацияциклических алгоритмов | 2 |  | 2 |
| 4.4. | Занятие15. СлучайныечислаПрактическаяработа:Случайныечисла | 3 | 1 | 2 |
| 4.5. | Занятие16. Примерырешениязадач сцикломПрактическаяработа:Решение задач сциклом. | 4 | 1 | 3 |
| 4.6 | Самостоятельная работа №2 "Составление программ сциклом" | 5 | 1 | 4 |
| 4.7 | Тест №4. Циклы | 1 |  | 1 |
| 4.8 | Занятие17. Творческая работа№1. "Циклы" | 3 | 1 | 2 |
| **5** | **Раздел 5. Функции** | **10** | **4** | **6** | Тестирование,решение практических задач |
| 5.1. | Занятие18. Создание функцийПрактическаяработаСоздание функций | 2 | 1 | 1 |
| 5.2. | Занятие19. ЛокальныепеременныеПрактическаяработаЛокальныепеременные | 2 | 1 | 1 |
| 5.3. | Занятие20. Примерырешениязадач c использованиемфункцийПрактическаяработаРешение задач сиспользованием функций | 2 | 1 | 1 |
| 5.4 | Самостоятельнаяработа №3 по теме"Функции" | 2 |  | 2 |
| 5.5 | Занятие21. РекурсивныефункцииПрактическаяработа5.4. Рекурсивныефункции | 2 | 1 | 1 |
| **6** | **Раздел 6. Строки - последовательности символов** | **5** | **1** | **4** | Решение практических задач |
| 6.1. | Занятие 22. СтрокиПрактическая работа: Строки | 2 | 1 | 1 |
| 6.2. | Занятие23. Срезы строк | 2 |  | 2 |
| 6.3 | Занятие24. Примерырешениязадач сострокамиПрактическаяработа:Решение задач состроками. | 1 |  | 1 |
| **7** | **Раздел 7.Сложные типы данных** | **8** | **2** | **6** | Тестирование,Решение практических задач |
| 7.1. | Занятие25. СпискиПрактическая работа:Списки | 2 | 1 | 1 |
| 7.2 | Занятие26.Срезы списков | 2 |  | 2 |
| 7.6 | Занятие29. Кортежи | 4 | 1 | 3 |
|  | **ВСЕГО** | **68** | **16** | **52** |  |

**Содержание учебного плана**

**Вводное занятие (1 часа).**

Теория: Знакомство с планом работы объединения, инструктаж по ТБ. Практика: Опрос. Тренинг на командообразование.

### Раздел 1. Знакомствосязыком Python (1часов)

Теория: ОбщиесведенияоязыкеPython.УстановкаPythonнакомпьютер.РежимыработыPython.Чтотакоепрограмма.Перваяпрограмма.СтруктурапрограммынаязыкеPython.Комментарии.

Практическаяработа 1.1. Установка программыPythonПрактическаяработа 1.2. Режимыработы сPythonТест №1. Знакомствосязыком Python

*Учащиеся должны знать/ понимать:*

* понятие программы;
* структура программы на Python;
* режимы работы с Python.

*Учащиеся должныуметь:*

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

### Раздел 2. Переменные и выражения (5 часов)

Теория: Типыданных. Преобразование типов. Переменные. Оператор присваивания.Именапеременныхиключевыеслова.

Выражения.Операции. Порядок выполненияопераций.МаРазделтическиефункции.Композиция.

Вводивывод.Вводданныхсклавиатуры.Выводданныхнаэкран.Примерскрипта,использующеговводивыводданных.Задачинаэлементарныедействиясчислами.Решение задач на элементарныедействия счислами.

Практическаяработа 2.1. Работа сосправочнойсистемойПрактическаяработа 2.2. Переменные

Практическаяработа 2.3. Выражения

Практическаяработа 2.5.Задачи на элементарныедействия с числамиТест №2.Выражения иоперации.

*Учащиеся должны знать/ понимать:*

* общую структуру программы;
* типы данных;
* целые, вещественные типы данных и операции над ними;
* оператор присваивания;

### Раздел 3. Условные предложения (14 часов)

Теория: Логическийтипданных.Логическиевыраженияиоператоры.Сложныеусловныевыражения(логическиеоперацииand,or,not).Условныйоператор.Альтернативноевыполнение.Примерырешениязадачсусловнымоператором.Множественноеветвление.Реализацияветвления вязыкеPython.

Практическаяработа 3.1. Логические выраженияПрактическаяработа 3.2. "Условныйоператор"Практическаяработа 3.3. Множественное ветвлениеПрактическаяработа 3.4. "Условныеоператоры"

Самостоятельнаяработа №1. Решение задач потеме"Условныеоператоры".Зачетнаяработа№1. "Составление программс ветвлением".

Тест №3."Условныеоператоры".

*Учащиеся должны знать/ понимать:*

* назначение условного оператора;
* способ записи условного оператора;
* логический тип данных;
* логические операторы or, and, not;
* Учащиеся должны уметь:
* использовать условный оператор;
* создавать сложные условия с помощью логических операторов.

### Раздел 4. Циклы (23 часов)

Теория: Понятиецикла.Телоцикла.Условиявыполнениятелацикла.Операторцикласусловием.Операторциклаwhile.Бесконечныециклы.Альтернативнаяветкациклаwhile.Обновлениепеременной.Краткаяформазаписиобновления.Примерыиспользованияциклов.

Операторцикласпараметромfor.Операторыуправленияциклом.Примерзадачисиспользованиемциклаfor.Вложенныециклы.Циклывциклах.Случайныечисла.Функцияrandrange.Функцияrandom.Примерырешениязадач с циклом.

Практическаяработа 4.1. "Числа Фибоначчи"Практическаяработа 4.2. Решение задачи сцикломfor.

Практическаяработа 4.3. РеализацияциклическихалгоритмовПрактическаяработа 4.4. Случайныечисла

Практическаяработа 4.5. Решение задач с циклом.Самостоятельнаяработа №2"Составлениепрограмм сциклом"

Тест №4. Циклы

Творческая работа№ 1."Циклы"

*Учащиеся должны знать/ понимать:*

* циклы с условием и их виды;
* правила записи циклов условием;
* назначение и особенности использования цикла с параметром;
* формат записи цикла с параметром;
* примеры использования циклов различных типов.

*Учащиеся должныуметь:*

* определять вид цикла, наиболее удобный для решения поставленной задачи;
* использовать цикл с условием;
* определять целесообразность применения и использовать цикл с параметром для решения поставленной задачи.

### Раздел 5. Функции (10 часов)

Теория: Создание функций.Параметры и аргументы. Локальныеи глобальныепеременные.

Поток выполнения. Функции,возвращающие результат. Анонимныефункции,инструкцияlambda.Примеры решения задачc использованием функций.

Рекурсивныефункции.Вычисление факториала.Числа Фибоначчи.Практическаяработа 5.1. Создание функций

Практическаяработа 5.2. Локальныепеременные

Практическаяработа 5.3. Решение задач с использованием функцийПрактическаяработа 5.4. Рекурсивныефункции

Самостоятельнаяработа №3 по теме"Функции"Тест №5. Функции

*Учащиеся должны знать/ понимать:*

* понятие функции;
* способы описания функции;
* принципы структурного программирования;
* понятие локальных переменных подпрограмм;
* понятие формальных и фактических параметров подпрограмм;
* способ передачи параметров.

*Учащиеся должныуметь:*

* создавать и использовать функции;
* использовать механизм параметров для передачи значений.

### Раздел 6. Строки- последовательностисимволов (5 часов)

Теория: Составнойтипданных - строка.Доступ по индексу. Длина строки и отрицательныеиндексы.Преобразование типов. Применение цикладля обхода строки.

Срезыстрок.Строкинельзяизменить.Сравнениестрок.Операторin.Модульstring.Операторыдлявсехтиповпоследовательностей(строки,списки,кортежи).Примерырешениязадач состроками.

Практическаяработа 6.1. Строки

Практическаяработа 6.2. Решение задач состроками.

*Учащиеся должны знать/ понимать:*

* назначение строкового типа данных;
* операторы для работы со строками;
* процедуры и функции для работы со строками;
* операции со строками.

*Учащиеся должныуметь:*

* описывать строки;
* соединять строки;
* находить длину строки;
* вырезать часть строки;
* находить подстроку в строке;
* находить количество слов в строке.

### Раздел 7. Сложные типы данных (8 часов)

Теория: Списки.Типсписок(list).Индексы.Обходсписка.Проверкавхождениявсписок.Добавлениевсписок.Суммированиеилиизменениесписка.Операторыдлясписков.Срезысписков.Удалениесписка.Клонированиесписков.Списочныепараметры.Функцияrange.Списки:примерырешениязадач.

Матрицы.Вложенныесписки.Матрицы.Строки и списки.Генераторысписков в Python.Кортежи. Присваивание кортежей. Кортежикаквозвращаемые значения

Введение в словари. Тип словарь(dict). Словарныеоперации. Словарныеметоды.

Множества вязыкеPython.Множества.Множественныйтипданных. Описаниемножеств.Операции,допустимыенадмножествами:объединение,пересечение,разность,включение.Операторопределенияпринадлежностиэлемента множеству.

Практическаяработа 7.1. Списки.

Практическаяработа 7.2. Решение задач со списками.Тест №7. Списки

*Учащиеся должны знать/ понимать:*

* сложныетипы данных;
* способописаниясписка;
* способдоступа к элементам списка;
* способописаниякортежа;
* способописаниясловаря;
* операции,выполняемые сосписками,кортежамиисловарями;
* понятие множества;
* способыописаниямножества;
* операторыработы с множествами.

*Учащиеся должныуметь:*

* 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.

**1.4. Планируемые результаты освоения программы**

### Планируемыерезультатыкурса

Врамкахкурса«ПрограммированиянаязыкеPython»учащиесяовладеваютследующимизнаниями, умениями и способамидеятельности:

* умеют составлять алгоритмы для решения задач;
* умеют реализовывать алгоритмы на компьютере в виде программ, написанных на языке Python;
* владеют основными навыками программирования на языке Python;
* умеют отлаживать и тестировать программы, написанные на языке Python.

### Способы оцениванияуровня достиженийучащихся

Предметомдиагностикииконтролявкурсе«ОсновыпрограммированиянаязыкеPython»являютсявнешниеобразовательныепродуктыучащихся(созданныеблок-схемы,программы),атакжеихвнутренниеличностныекачества(освоенныеспособыдеятельности,знания,умения), которыеотносятся к целям и задачам курса.

Качествовнешнейобразовательнойпродукциижелательнооцениватьпоследующимпараметрам:

* алгоритм должен быть оптимальным по скорости выполнения и максимально простым в реализации на языке программирования;
* программа должна выполнять поставленные задачи;
* по степени «читаемости кода» (должны быть соблюдены отступы, обязательное наличие комментариев к коду программы и т. д.).
* Проверка достигаемых учащимися результатов производится в следующих формах:
* текущий рефлексивный самоанализ, контроль и самооценка учащимися выполняемых заданий;
* текущая диагностика и оценка педагогом деятельности учащихся;
* итоговая оценка деятельности и образовательной продукции ученика в соответствии с его индивидуальными и возрастными особенностями;
* Итоговый контроль проводится в конце всего курса. Он состоит из тестирования и решения практической задачи, защиты творческого проекта.

**РАЗДЕЛ 2. КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ**

**2.1. Календарный учебный график**

Работа по программе осуществляется с 01 сентября по 31мая.

**2.2. Условия реализации программы**

***Материально-техническое обеспечение:***

* столы для компьютера;
* компьютерные стулья;
* шкафы для дидактических материалов, пособий;
* специальная и научно-популярная литература для педагога и учащихся;
* канцтовары;

***Информационное обеспечение:***

* персональный компьютер (на каждого участника);
* мультимедийный проектор;
* видеоматериалыразной тематики по программе;
* оргтехника;
* выход в сеть Internet;

***Аппаратное обеспечение:***

* Процессор не ниже Core2 Duo;
* Объем оперативной памяти не ниже 4 ГбDDR3;
* Дисковое пространство на менее 128 Гб;
* Монитор диагональю на мене 19’;

***Программное обеспечение:***

* Операционная система Windows 7 Профессиональная или выше;
* Интерпретатор Python версии 3.7 и выше;
* IDE JatBrainsPyCharm;
* FoxitReaderили другойпросмоторщик PDF файлов;
* WinRAR;
* Пакет офисных программ;
* Adobe Photoshop или другой растровый графический редактор;
* Любой браузер дляинтернет серфинга.

**2.6. Список литературы**

***Нормативно-правовые акты и документы***

1. Федеральный закон «Об образовании в Российской Федерации» от 29.12.2012 № 273-ФЗ
2. Приказ Министерства образования и науки РФ от 29.08.2013 №1008 «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».
3. Письмо Министерства образования Российской Федерации от 18 июня 2003 №28-02-484/16 «Требования к содержанию и оформлению образовательных программ дополнительного образования детей»,
4. Постановление Главного государственного санитарного врача Российской Федерации от 4.07.2014 г. «Об утверждении СанПиН 2.4.4.3172-14 «Санитарно-эпидемиологические требования к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей».
5. Стратегия инновационного развития Российской Федерации на период до 2020 года. Распоряжение правительства Российской Федерации от 08.12.2011 № 2227-р.
6. Концепция развития дополнительного образования детей. Распоряжение Правительства Российской Федерации от 24 апреля 2015 года № 729-р.
7. Стратегия развития воспитания в Российской Федерации на период до 2025 года». Распоряжение Правительства Российской Федерации от 29 мая 2015 года № 996-р.

***Литература для педагога***

1. Бизли, Дэвид М. Python. Подробный справочник. – М.–СПб.: Символ-Плюс, 2010.
2. Лутц, Марк Python. Справочник. – М.: Вильямс, 2015.
3. Официальный сайт программы [Электронный ресурс] − Режим доступа: <https://docs.python.org/>, свободный.
4. Сайт, среда разработки для языка Python. [Электронный ресурс] − Режим доступа: <https://www.jetbrains.com/pycharm/?fromMenu>, свободный.

***Литература для родителей и учащихся***

1. Сайт / справочные материалы[Электронный ресурс] − Режим доступа: <https://metanit.com/python/>, свободный.
2. Сайт / интерактивный сборник задач для практики программирования [Электронный ресурс] − Режим доступа: <http://pythontutor.ru/>, свободный.
3. Сайт / Адаптивный тренажер Python[Электронный ресурс] − Режим доступа: <https://stepik.org/course/431> , свободный.
4. Сайт / среда разработки для языка Python[Электронный ресурс] − Режим доступа: <https://www.jetbrains.com/pycharm/?fromMenu>, свободный.
5. Сайтпроекта Open Book Project. Практические примеры на Python Криса Мейерса [Электронный ресурс] − Режим доступа:openbookproject.net, свободный.

**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?



|  |
| --- |
|  |

1. В алфавите племени мумба-юмба32 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?

|  |
| --- |
|  |

1. 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.

|  |
| --- |
|  |

1. На рисунке показано расположение городов A, B, C и D и расстояния между ними. Турист выходит из города B 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?



|  |
| --- |
|  |

1. 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?



|  |
| --- |
|  |

1. 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?

|  |
| --- |
|  |

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



|  |
| --- |
|  |

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?

|  |
| --- |
|  |

1. 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 |

1. 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.

|  |  |
| --- | --- |
|  |  |

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

|  |
| --- |
|  |

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

|  |
| --- |
|  |

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

|  |
| --- |
|  |

1. В каком месяце болтливая девочка говорит меньше всего?

|  |
| --- |
|  |

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

|  |
| --- |
|  |

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

|  |
| --- |
|  |

1. 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.

|  |
| --- |
|  |

1. 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?

|  |
| --- |
|  |

|  |
| --- |
| **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⋅1111111**— произведение чисел 11111 (5 единиц) и 1111111 (7 единиц)** |
| Запишите число **1.2345e3** в виде [**десятичной** дроби](https://ru.wikipedia.org/wiki/%D0%94%D0%B5%D1%81%D1%8F%D1%82%D0%B8%D1%87%D0%BD%D0%B0%D1%8F_%D0%B4%D1%80%D0%BE%D0%B1%D1%8C). |
| 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 выраженииa **and** b **or not** a **and not** bв 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 = Trueb = Falsea 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('\*')**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:Треугольникabc***где a, b и c — длины сторон треугольника***прямоугольникab***где a и b — длины сторон прямоугольника***кругr***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*×*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)** |