

ЦЕНТР ЦИФРОВОГО
ОБРАЗОВАНИЯ ДЕТЕЙ «IT-КУБ»

ТОЧКА РОСТА



**МУНИЦИПАЛЬНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ИРКУТСКОГО РАЙОННОГО МУНИЦИПАЛЬНОГО ОБРАЗОВАНИЯ
«УРИКОВСКАЯ СРЕДНЯЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ШКОЛА»**

СОГЛАСОВАНО

«13» сентября 2023г.

Заместитель директора по

УВР

Е.П. Озернова
подпись Ф.И.О.

УТВЕРЖДАЮ

Директор

Гоняковская Е.Ю.

Приказ № 650-ОД

от «13» сентября 2023г.



**Дополнительная общеобразовательная программа
«ПРОГРАММИРОВАНИЕ РОБОТОВ»**

Возраст обучающихся: 7-11 лет

Срок реализации: 1 год.

Направленность: технологическая

Авторы-составители: Суворкин Я.В., Чумаченко Н.С.

г. Иркутск, 2023

ОБЩАЯ ХАРАКТЕРИСТИКА ПРОГРАММЫ

1.1. 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 внутри профильной специализации в области новых информационных технологий, что способствует созданию дополнительных условий для проявления индивидуальных образовательных интересов учащихся, их дальнейшей профессиональной ориентации.

1.2. Направленность программы – техническая

1.3. Адресат программы:

К освоению программы допускаются обучающиеся по общеобразовательным программам в возрасте от 14 до 17 лет.

1.4. Цель, задачи и планируемые результаты освоения программы:

Цель: способствовать формированию творческой личности, обладающей информационными компетенциями, владеющей базовыми понятиями теории алгоритмов, умеющей разрабатывать эффективные алгоритмы и реализовывать их в виде модели с использованием платформы Arduino.

Основные задачи:

- познакомить с понятиями алгоритма, вычислимой функции, языка программирования;
- научить составлять и читать блок-схемы;
- сформировать навыки выполнения технологической цепочки разработки программ средствами языка программирования;
- научить отлаживать и тестировать устройства, программируемые на arduino

Планируемые результаты освоения:

В результате изучения курса должны быть достигнуты определенные результаты.

Предметные:

- умеют составлять алгоритмы для решения задач;
- умеют реализовывать алгоритмы на компьютере в виде программ, написанных на языке C++;
- владеют основными навыками программирования на языке C++;
- умеют отлаживать и тестировать программы, написанные на языке C++ на готовых устройствах;
- научить отлаживать и тестировать механизмы на платформе arduino

Личностные:

- 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;
- сформированы способности к саморазвитию и самообразованию средствами информационных технологий на основе иллюстрированной среды программирования, мотивации к обучению и познанию;
- сформированы умения работать индивидуально и в группе для решения поставленной задачи;
- сформировано целостное мировоззрение, соответствующее современному уровню развития информационных технологий;

- сформировано осознанное позитивное отношение к другому человеку, его мнению, результату его деятельности;
- обеспечено усвоение правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой;
- сформирована культура начального программирования.

Метапредметные:

- развито умение самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности;
- развито умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата;
- развито умение критически оценивать правильность решения учебно-исследовательской задачи;
- сформировано владение основами самоконтроля, способность к принятию решений;
- сформирована мотивация к профессиональному самоопределению учащихся.

1.5. Объем и срок освоения программы: объем программы 112 часов, срок освоения программы - 14 недель

1.6. Форма обучения: очная

1.7. Формы аттестации. Оценочные материалы.

- текущий контроль (тестирование) входит в количество часов по изучению разделов;

- итоговая аттестация (творческий проект).

1.8. Режим занятий – 112 часов, при очной форме обучения 2 раза в неделю, 2 часа в день (перерыв между занятиями 10 минут). Допускается использование ДОД.

1.9. Особенности организации образовательной деятельности. 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. –образовательная деятельность организована в традиционной форме: лекционно-практические занятия (возможно применение ДОД).

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.

1.10. Форма итоговой аттестации – защита проекта

1. УЧЕБНЫЙ ПЛАН

2.1.1. Учебный план по очной форме обучения

№	Наименование разделов, дисциплин (модулей)	всего часов	Аудиторная нагрузка			Промежуточная аттестация
			теоретические занятия	практич. занятия	СРС	форма
I	Раздел 1. Введение в Arduino	12	4	3	5	Опрос
1.1.	Что такое Arduino?	4	2	1	1	

1.2.	Среда разработки Arduino.	4	1	1	2	
1.3.	Структура программы.	4	1	1	2	
II	Раздел 2. Программирование Arduino	37	9	10	18	Тестирование, решение практических задач
2.1.	Переменные и константы.	4	1	1	2	
2.2.	Математические операции.	4	1	1	2	
2.3.	Логические и поразрядные операции.	4	1	1	2	
2.4.	Операторы сравнения.	4	1	1	2	
2.5.	Функции.	4	1	1	2	
2.6.	Циклы.	4	1	1	2	
2.7.	Операторы языка Си.	4	1	1	2	
2.8.	Условные операторы.	4	1	1	2	
2.9.	Операторы цикла.	4	1	1	2	
2.10.	Программирование Arduino.	1		1		
III	Раздел 3. Подключение	49	12	13	24	Тестирование, решение практических задач
3.1.	Мигание светодиода.	4	1	1	2	
3.2.	Цифровые выводы.	4	1	1	2	
3.3.	Цифровые выводы с ШИМ сигналом.	4	1	1	2	
3.4.	Общение с компьютером с помощью Serial Monitor.	4	1	1	2	
3.5.	Подключение кнопки. Обработка нажатия.	4	1	1	2	
3.6.	Управление светодиодами.	4	1	1	2	
3.7.	Управление сервоприводами.	4	1	1	2	
3.8.	Создание матричной клавиатуры с помощью кнопок.	4	1	1	2	
3.9.	Создание LED-куба.	4	1	1	2	
3.10.	Использование шаговых двигателей.	4	1	1	2	
3.11.	Ультразвуковой датчик. Определение расстояния.	4	1	1	2	
3.12.	Использование двигателя постоянного тока.	4	1	1	2	
3.13.	Подключение.	1		1		
IV	Раздел 4. Итоговый проект	12	1	2	9	Опрос
4.1.	Обсуждение итогового проекта. Ответы на вопросы.	2		2		
4.2.	Создание итогового	10	1		9	

	проекта.					
Итоговая аттестация		2		2		Защита проекта
	Итого:	112	26	30	56	

1 По темам разделов II, III на практических занятиях допускается деление на подгруппы (не менее 3 человек в подгруппе)

2. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

3.1. Для реализации дополнительной общеразвивающей программы предусмотрена очная форма обучения, очная форма обучения с применением ДОД.

3.2. Срок освоения ДОП

3.2.1. Срок освоения ДОП при очной форме обучения составляет 34 недели, в том числе:

Обучение по разделам (дисциплинам)	12 нед
Промежуточная аттестация	1 нед
Итоговая аттестация	1 нед
Итого	14 нед.

3.3. Календарные сроки реализации ДОП устанавливаются на основании плана-графика.

3.3.1. Календарные сроки реализации ДОП при очной форме обучения

№	Наименование разделов, дисциплин, модулей и тем	всего часов в/в т.ч. СР	1 месяц				2 месяц				3 месяц				4 месяц						
I	Раздел 1. Введение в Arduino	12																			
1.1.	Что такое Arduino?	4			4																
1.2.	Среда разработки Arduino.	4			4																
1.3.	Структура программы.	4				4															
II	Раздел 2. Программирование Arduino	37																			
2.1.	Переменные и константы.	4				4															
2.2.	Математические операции.	4					4														
2.3.	Логические и поразрядные операции.	4					4														
2.4.	Операторы сравнения.	4						4													
2.5.	Функции.	4						4													
2.6.	Циклы.	4							4												

.																			
2.7	Операторы языка Си.	4							4										
2.8	Условные операторы.	4							4										
2.9	Операторы цикла.	4							4										
2.1 0.	Программирование Arduino.	1							1										
III	Раздел 3. Подключение	49																	
3.1	Мигание светодиода.	4								4									
3.2	Цифровые выводы.	4								4									
3.3	Цифровые выводы с ШИМ сигналом.	4									4								
3.4	Общение с компьютером с помощью Serial Monitor.	4									4								
3.5	Подключение кнопки. Обработка нажатия.	4										4							
3.6	Управление светодиодами.	4										4							
3.7	Управление сервоприводами.	4											4						
3.8	Создание матричной клавиатуры с помощью кнопок.	4												4					
3.9	Создание LED-куба.	4													4				
3.1 0.	Использование шаговых двигателей.	4													4				
3.1 1.	Ультразвуковой датчик. Определение расстояния.	4														4			
3.1 2.	Использование двигателя постоянного тока.	4														4			
3.1 3.	Подключение.	1															1		
IV	Раздел 4. Итоговый проект	12																	
4.1	Обсуждение итогового проекта. Ответы на вопросы.	2																2	
4.2	Создание итогового проекта.	10																5	5
	Итоговая аттестация	2																	2
	Итого	112																	

3. СОДЕРЖАНИЕ ДОПОЛНИТЕЛЬНОЙ ОБЩЕРАЗВИВАЮЩЕЙ ПРОГРАММЫ

Наименование, содержание раздела, дисциплины (практические, теоретические занятия)	Всего часов
Раздел 1. Введение в Arduino	12
Тема 1.1. Инструктаж по технике безопасности в компьютерном классе	1
Теоретическое занятие 1.1. Инструктаж по технике безопасности в компьютерном классе.	1
Тема 1.2. Что такое Arduino?	3
Теоретическое занятие 1.2. Основные понятия, связанные с Arduino. Микроконтроллеры. Цифровые порты. Макетная плата. Напряжение.	1
Практическое занятие 1.2. Питание микроконтроллера Arduino Uno.	1
Самостоятельная работа 1.2. Конспект по пройденному материалу.	1
Тема 1.3. Среда разработки Arduino	4
Теоретическое занятие 1.3. Изучение среды разработки Arduino IDE.	1
Практическое занятие 1.3. Изучение интерфейса программы.	1
Самостоятельная работа 1.3.1. Конспект по пройденному материалу.	1
Самостоятельная работа 1.3.2. Поиск нужной вкладки в интерфейсе программы.	1
Тема 1.4. Структура программы	4
Теоретическое занятие 1.4. Стартовая структура программы. Процедуры.	1
Практическое занятие 1.4. Написание простого скетча на Arduino.	1
Самостоятельная работа 1.4.1. Конспект по пройденному материалу.	1
Самостоятельная работа 1.4.2. Написание простого скетча на Arduino.	1
Раздел 2. Программирование Arduino	37
Тема 2.1. Переменные и константы	4
Теоретическое занятие 2.1. Создание переменных. Типы данных. Простейшие математические действия с числами.	1
Практическое занятие 2.1. Создание переменных и их использование.	1
Самостоятельная работа 2.1. Решение задач на тему «Переменные и константы».	2
Тема 2.2. Математические операции	4
Теоретическое занятие 2.2. Базовые математические операции. Типы данных для математических операций.	1
Практическое занятие 2.2. Решение математических задач.	1
Самостоятельная работа 2.2. Перевод решений математических задач в код.	2
Тема 2.3. Логические и поразрядные операции	4
Теоретическое занятие 2.3. Логическое И, ИЛИ, НЕ. Память, для совершения процедур.	1
Практическое занятие 2.3. Ответы на вопросы.	1
Самостоятельная работа 2.3. Написание конспекта. Решение задач.	2
Тема 2.4. Операторы сравнения	4
Теоретическое занятие 2.4. Сравнение значений. True, False.	1
Практическое занятие 2.4. Решение модели задачи из реальной жизни.	1
Самостоятельная работа 2.4.1. Конспект по пройденным темам.	1
Самостоятельная работа 2.4.2. Сравнение двух переменных.	1
Тема 2.5. Функции	4

Теоретическое занятие 2.5. Собственные функции. Возврат значений. Аргументы функций.	1
Практическое занятие 2.5. Написание функции.	1
Самостоятельная работа 2.5.1. Написание функции без входного аргумента.	1
Самостоятельная работа 2.5.2. Написание функции с входным аргументом.	1
Тема 2.6. Циклы	4
Теоретическое занятие 2.6. Циклы for и while. Команды continue, break.	1
Практическое занятие 2.6. Использование цикла для поиска нужного значения.	1
Самостоятельная работа 2.6.1. Использование цикла for.	1
Самостоятельная работа 2.6.2. Использование цикла while.	1
Тема 2.7. Операторы языка Си	4
Теоретическое занятие 2.7. Безусловный переход goto. Switch case. Do while.	1
Практическое занятие 2.7. Использование операторов языка Си.	1
Самостоятельная работа 2.7. Использование операторов языка Си.	2
Тема 2.8. Условные операторы	4
Теоретическое занятие 2.8. Переход к коду по условию. Исключение четных чисел. Условие if, else if, else.	1
Практическое занятие 2.8. Решение задач с условиями.	1
Самостоятельная работа 2.8.1. Использование вложенных условий.	1
Самостоятельная работа 2.8.2. Использование каскадных условий.	1
Тема 2.9. Операторы цикла	4
Теоретическое занятие 2.9. Счетчик событий. Счетчик количества. Условие завершения.	1
Практическое занятие 2.9. Использование операторов цикла.	1
Самостоятельная работа 2.9. Использование операторов цикла.	2
Тема 2.10. Программирование Arduino	1
Практическое занятие 2.10. Тестирование по пройденным темам.	1
Раздел 3. Подключение	49
Тема 3.1. Мигание светодиода	4
Теоретическое занятие 3.1. Подключение платы к компьютеру. Использование встроенного светодиода. Включение и выключение светодиода.	1
Практическое занятие 3.1. Включение светодиода.	1
Самостоятельная работа 3.1. Конспект необходимых команд. Включение-выключение светодиода.	2
Тема 3.2. Цифровые выводы	4
Теоретическое занятие 3.2. Вход и выход. Функции pinMode, digitalWrite, digitalRead. Резисторы.	1
Практическое занятие 3.2. Подключение проводов к плате Arduino.	1
Самостоятельная работа 3.2. Конспект необходимых команд. Подключение проводов к Arduino.	2
Тема 3.3. Цифровые выводы с ШИМ сигналом	4
Теоретическое занятие 3.3. Значение ШИМ-сигнала. Использование цифровых портов с ШИМ-сигналом. Символ тильды. Функция analogWrite.	1
Практическое занятие 3.3. Использование цифрового вывода с ШИМ-сигналом.	1

Самостоятельная работа 3.3. Использование цифрового вывода с ШИМ-сигналом.	2
Тема 3.4. Общение с компьютером с помощью Serial Monitor	4
Теоретическое занятие 3.4. Интерфейс UART, USART. Вывод сообщений. Вывод сигнала из Arduino. Вывод текста в консоль.	1
Практическое занятие 3.4. Использование Serial Monitor.	1
Самостоятельная работа 3.4.1. Передача информации из Serial Monitor в Arduino.	1
Самостоятельная работа 3.4.2. Отправление информации из Arduino в Serial Monitor.	1
Тема 3.5. Подключение кнопки. Обработка нажатия	4
Теоретическое занятие 3.5. Кнопка. Схема кнопки. Подключение кнопки. Скрытые недостатки кнопок.	1
Практическое занятие 3.5. Подключение кнопки и передача сигнала светодиоду.	1
Самостоятельная работа 3.5.1. Конспект по пройденной теме.	1
Самостоятельная работа 3.5.2. Подключение кнопки. Получение сигнала из кнопки.	1
Тема 3.6. Управление светодиодами	4
Теоретическое занятие 3.6. Подключение светодиодов. Схема светодиода. Резисторы.	1
Практическое занятие 3.6. Подключение нескольких светодиодов.	1
Самостоятельная работа 3.6.1. Конспект по пройденной теме.	1
Самостоятельная работа 3.6.2. Подключение нескольких светодиодов и управление ими.	1
Тема 3.7. Управление сервоприводами	4
Теоретическое занятие 3.7. Сервопривод. Схема сервопривода. Электромотор, редуктор, потенциометр. Схема подключения.	1
Практическое занятие 3.7. Подключение и управление сервоприводом.	1
Самостоятельная работа 3.7.1. Подключение сервопривода.	1
Самостоятельная работа 3.7.2. Поворот сервопривода.	1
Тема 3.8. Создание матричной клавиатуры из кнопок	4
Теоретическое занятие 3.8. Схема матричной клавиатуры. Устранениедребезгов у кнопки. Сборка клавиатуры.	1
Практическое занятие 3.8. Подключение матричной клавиатуры 2x2. Обработка нажатий каждой клавиши.	1
Самостоятельная работа 3.8. Подключение матричной клавиатуры 3x3. Обработка нажатий каждой клавиши.	2
Тема 3.9. Создание LED-куба	4
Теоретическое занятие 3.9. Схема LED-куба. Настройка режимов включения светодиодов.	1
Практическое занятие 3.9. Подключение LED-куба 2x2. Использование нескольких режимов включения светодиодов.	1
Самостоятельная работа 3.9. Подключение LED-куба 3x3. Использование нескольких режимов включения светодиодов.	2
Тема 3.10. Использование шаговых двигателей	4
Теоретическое занятие 3.10. Шаговый двигатель. Метод перемещения катушки. Угол для поворота. Библиотека Stepper.	1

Практическое занятие 3.10. Разбор схемы подключения шагового двигателя.	1
Самостоятельная работа 3.10.1. Конспект по пройденным темам.	1
Самостоятельная работа 3.10.2. Изучение основных функций библиотеки Stepper.	1
Тема 3.11. Ультразвуковой датчик. Определение расстояния	4
Теоретическое занятие 3.11. Ультразвуковой датчик. Диапазон дальности. Перевод счисления в сантиметры. Библиотека NewPing. Схема подключения.	1
Практическое занятие 3.11. Подключение ультразвукового датчика к Arduino.	1
Самостоятельная работа 3.11.1. Подключение ультразвукового датчика.	1
Самостоятельная работа 3.11.2. Определение расстояния до объекта.	1
Тема 3.12. Использование двигателя постоянного тока	4
Теоретическое занятие 3.12. Схема подключения двигателя. Транзисторы.	1
Практическое занятие 3.12. Разбор работы двигателя постоянного тока.	1
Самостоятельная работа 3.12.1. Конспект по пройденной теме.	1
Самостоятельная работа 3.12.2. Подключение двигателя используя транзисторы.	1
Тема 3.13. Подключение	1
Практическое занятие 3.13. Тестирование по пройденным темам.	1
Раздел 4. Итоговый проект	12
Тема 4.1. Обсуждение итогового проекта. Ответы на вопросы	2
Практическое занятие 4.1.1. Обсуждения условий защиты проекта.	1
Практическое занятие 4.2.1. Примерная демонстрация защиты проекта.	1
Тема 4.2. Создание итогового проекта	10
Теоретическое занятие 4.2. Что можно применять в проекте?	1
Самостоятельная работа 4.2. Создание проекта.	9
Итоговая аттестация (презентация проекта)	2
Итого	112

5. ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИЕ УСЛОВИЯ

5.1. Материально-техническое обеспечение

Технические средства обучения:

Реализация программы по адресу: Рябикова, 63;

- мебель, оборудование и расходные материалы (Приложение 1)

5.1. Информационное обеспечение обучения

Перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы:

Основные источники:

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.

Дополнительные источники:

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. Сайт проекта Open Book Project. Практические примеры на Python Криса Мейерса [Электронный ресурс] – Режим доступа: openbookproject.net , свободный.

Интернет-источники:

1. Сайт scratch.mit.edu — Официальный сайт языка программирования Scratch.
2. Сайт https://www.youtube.com/playlist?list=PL2_WTnP_CpnuiOlpGDcJWYqoSCXAGgUws — Видеокурс «Игры на Scratch».

5.2. Кадровое обеспечение: Реализация программы обеспечивается педагогическими кадрами, и имеющими, образование, соответствующее профилю/направленности программы.

5.3. Организация образовательного процесса

Каждый обучающийся имеет рабочее место с доступом к сети Интернет (при необходимости), к современным профессиональным базам данных, информационным справочным и поисковым системам, электронной библиотеке «Библиоклуб».

Программа обеспечивается учебно-методическим комплексом и материалами по всем дисциплинам, разделам (модулям).

Каждый обучающийся обеспечен не менее чем одним учебным печатным и/или электронным изданием по каждой дисциплине (модулю) (включая электронные базы периодических изданий).

Внеаудиторная работа обучающихся сопровождается методическим обеспечением и обоснованием времени, затрачиваемого на ее выполнение.

Образовательная деятельность обучающихся предусматривает следующие виды учебных занятий и учебных работ: лекции, практические занятия, выполнение проектной работы

6. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ ДОПОЛНИТЕЛЬНОЙ ОБЩЕРАЗВИВАЮЩЕЙ ПРОГРАММЫ

6.1. Текущая оценка результатов освоения программы осуществляется преподавателем в форме тестирования.

6.2. Освоение ДОП заканчивается итоговой аттестацией обучающихся. Лица, успешно освоившие ДОП и прошедшие итоговую аттестацию, получают сертификат.

6.3. В соответствии с учебным планом итоговая аттестация по программе «Основы алгоритмики и логики» осуществляется в форме проекта.

6.4. Порядок проведения итоговой аттестации:

Общий уровень достижений учащихся оценивается зачет/незачет по следующей шкале:

28 - 15 баллов: «зачет»;

15 - 0 баллов: «незачет»

Критерии оценки проекта:

1. Планирование и раскрытие плана, развитие темы. Высший балл ставится, если ученик определяет и четко описывает цели своего проекта, дает последовательное и полное описание того, как он собирается достичь этих целей, причем реализация проекта полностью соответствует предложенному им плану.

2. Сбор информации. Высший балл ставится, если персональный проект содержит достаточное количество относящейся к делу информации и ссылок на различные источники.

3. Выбор и использование методов и приемов. Высший балл ставится, если проект полностью соответствует целям и задачам, определенным автором, причем выбранные и эффективно использованные средства приводят к созданию итогового продукта высокого качества.

4. Анализ информации. Высший балл по этому критерию ставится, если проект четко отражает глубину анализа и актуальность собственного видения идей учащимся, при этом содержит по-настоящему личностный подход к теме.

5. Организация письменной работы. Высший балл ставится, если структура проекта и письменной работы (отчета) отражает логику и последовательность работы, если использованы адекватные способы представления материала (диаграммы, графики, сноски, макеты, модели и т. д.).

6. Анализ процесса и результата. Высший балл ставится, если учащийся последовательно и полно анализирует проект с точки зрения поставленных целей, демонстрирует понимание общих перспектив, относящихся к выбранному пути.

7. Личное участие. Считается в большей степени успешной такая работа, в которой наличествует собственный интерес автора, энтузиазм, активное взаимодействие с участниками и потенциальными потребителями конечного продукта и, наконец, если ребенок обнаружил собственное мнение в ходе выполнения проекта.

С критериями оценивания проектов учащиеся знакомятся заранее. Также они сами могут предложить какие-либо дополнения в содержание критериев или даже дополнительные критерии, которые, на их взгляд, необходимо включить в критериальную шкалу. Критерии оценивания являются своего рода инструкцией при работе над проектом. Кроме того, обучающиеся, будучи осведомленными о критериях оценивания их проектной деятельности, могут улучшить отдельные параметры, предлагаемые для оценивания, тем самым получить возможность достижения наивысшего результата.

Максимальный уровень достижений учащихся по критериям:

Критерии	Максимальный уровень достижений учащихся	
1	Планирование и раскрытие плана, развитие темы	4
2	Сбор информации	4
3	Выбор и использование методов и приемов	4
4	Анализ информации	4
5	Организация письменной работы	4
7	Анализ процесса и результата	4
7	Личное участие	4
ИТОГО		28

Перечень материально-технического обеспечения

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 JetBrains PyCharm;
Foxit Reader или другой просмотрщик PDF файлов;
WinRAR;
Пакет офисных программ;
Adobe Photoshop или другой растровый графический редактор;
Любой браузер для интернет серфинга.

Необходимое оборудование:

Ноутбук HP 255 G8 R5-3500U - 13 шт
ВЕБ-камера Logitech "HD C615, черный (960-001056) 1 шт
Наушники Edifier G1SE - 12 шт
Интерактивная панель TeachTouch 6.0 75", 20 касаний
Кабель HDMI – 1 шт
МФУ Pantum M6550NW

Мебель:

Доска магнитно-маркерная 100*150 см Ecopomy
Доска магнитно-маркерная Brauberg флипчарт 700*1000мм
Стул ученический (полумягкие цветные) 12 шт
Стул учительский (кресла полумягкие черные) 1 шт
Стол ученический (цветная коемочка) 12 шт
Стол учительский 1 шт

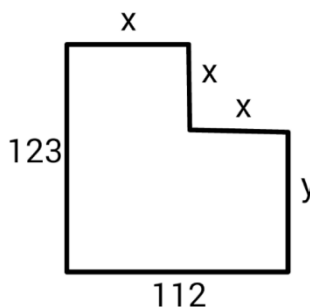
Входящий контроль

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.

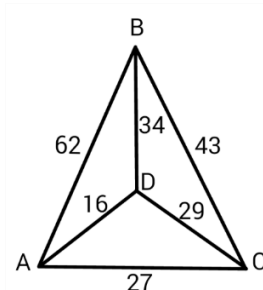
Сколько кусков рулета получится, если разрезать по кольцам всех трёх цветов?

Примечания

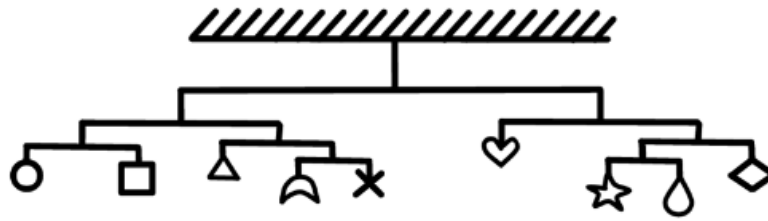
Учтите, что наклеить кольцо одного цвета на кольцо другого нельзя.

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;
- структуру 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: $11111 \cdot 11111111$ — произведение чисел 11111 (5 единиц) и 11111111 (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 $a \text{ and } b \text{ or not } a \text{ and not } b$ 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 <i>a</i> и <i>b</i> Учитывайте регистр символов при ответе. a = True b = False a and b or not a and not b
Отметьте выражения, значения которых равны True: <ul style="list-style-type: none"> • "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>i</i> после выполнения фрагмента программы? <pre>i = 0 while i <= 10: i = i + 1 if i > 7: i = i + 2</pre>
Сколько итераций цикла будет выполнено в этом фрагменте программы? <pre>i = 0 while i <= 10: i = i + 1 if i > 7: i = i + 2</pre>
Сколько всего знаков * будет выведено после исполнения фрагмента программы: <pre>i = 0 while i < 5: print('*') if i % 2 == 0:</pre>

```
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 $n=5$)