

МУНИЦИПАЛЬНОЕ ОБРАЗОВАНИЕ «КИЖИНГИНСКИЙ РАЙОН»  
МУНИЦИПАЛЬНОЕ УЧРЕЖДЕНИЕ ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ  
«Кижингинская станция детского (юношеского) технического творчества»

Принята на заседании  
педагогического совета  
от «25» августа 2025 г.  
Протокол №1



УТВЕРЖДАЮ  
Директор МУДО  
«Кижингинская СД(Ю)ТТ»  
*С.С. Гармажапова*  
«25» августа 2025 г.

Рабочая программа  
дополнительной общеобразовательной общеразвивающей программы  
технической направленности по основам программирования  
**«СтартPro – 4»**

Возраст учащихся: 12-17 лет

срок реализации – 1 год

Автор-составитель:  
Самбилова Туяна Пурбуевна,  
педагог дополнительного образования

с. Кижинга  
2025 г.

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Рабочая программа стартового уровня является структурным разделом дополнительной общеобразовательной общеразвивающей программы «Основы программирования на языке Python». Программа имеет техническую направленность.

### **Актуальность программы**

Актуальность данной Программы состоит в том, что она составлена с учётом современных потребностей рынка в специалистах в области информационных технологий, особенно в области программирования.

Научившись программировать на языке Python, обучающиеся получат мощный и удобный инструмент для решения как учебных, так и прикладных задач. Вместе с тем чистота и ясность его конструкций позволит обучающимся потом с лёгкостью выучить любой другой язык программирования.

**Цель:** сформировать и развить алгоритмическое и логическое мышление у обучающихся посредством языка программирования «Python».

### **Задачи программы:**

#### ***Образовательные:***

- сформировать у детей представление об основных элементах программирования;
- познакомить с синтаксисом языка программирования Python;
- сформировать у детей навыки работы в интегрированной среде разработки на языке Python;
- способствовать приобретению навыков разработки эффективных алгоритмов и программ на основе изучения языка программирования Python у детей.

#### ***Развивающие:***

- совершенствовать аналитические навыки;
- формировать навык алгоритмического и логического мышления;
- совершенствовать навык поиска информации в сети Интернет, анализа выбранной информации на соответствие запросу, использование информации при решении задач;
- развивать умение планировать свои действия с учётом фактора времени;

#### ***Воспитательные:***

- воспитывать в детях усидчивость, аккуратность, умение доводить начатое дело до конца;
- формировать коммуникативные навыки.

### **Отличительная особенность программы:**

В основу Программы заложены принципы модульности и практической направленности. Содержание учебных модулей направлено на:

- детальное изучение алгоритмизации;
- реализацию межпредметных связей;
- организацию проектной и исследовательской деятельности обучающихся.

**Педагогическая целесообразность программы** заключается в том, что дети приобретут практические навыки, которые станут основой для

дальнейшего изучения основ программирования. Методы, применяемые в процессе обучения, такие как проблемное обучение, проектная деятельность, способствуют формированию мотивации обучающихся к углубленному изучению программирования, как одной из компьютерных наук. У детей формируется познавательный интерес, самостоятельность мышления, стремление к самопознанию.

**Адресат программы:** программа предназначена для работы с обучающимися 12-17 лет (7-11 классы общеобразовательной школы).

*Особенности комплектования групп и количественный состав*

Набор обучающихся проводится без предварительного отбора детей в соответствии с возрастом обучающихся 12-17 лет.

**Формы организации образовательного процесса и методы обучения:** при изучении тем программа предусматривает использование фронтальной, индивидуальной и групповой формы учебной работы обучающихся.

При организации занятий по курсу «Программирование на языке Python» для достижения поставленных целей и задач используются формы проведения занятий с активными методами обучения:

- занятие в форме проблемно-поисковой деятельности;
- занятие с использованием межпредметных связей;
- занятие в форме мозгового штурма;
- занятие в форме частично-поисковой деятельности.

**Формы аттестации:** промежуточная аттестация за первое и второе полугодие проводится в форме защиты проекта.

Итоговые работы представляются на выставке технического творчества, что дает возможность обучающимся оценить значимость своей деятельности, услышать и проанализировать отзывы со стороны сверстников и взрослых.

**Объем программы:** 108 академических часов в год. 3 подгруппы по 3 часа в неделю.

**Срок освоения программы:** программа стартового уровня рассчитана на один год обучения.

**Режим занятий:** занятия проводятся 3 раза в неделю по 1 академическому часу (45 мин) с перерывом 5-10 минут.

**Планируемые результаты:**

**В результате освоения программы по обучающему аспекту обучающиеся должны:**

**Знать:**

- основы современных языков программирования;
- практическую направленность с ориентацией на реальные потребности, соответствующие возрасту обучающихся.

**Уметь:**

- объяснять и использовать на практике как простые так и сложные структуры данных и конструкций для работы с ними;
- искать и обрабатывать ошибки в коде;
- разбирать решение задач на подзадачи;

- писать грамотный красивый код;
- находить, оценивать, использовать информацию из различных источников, необходимую для решения профессиональных задач, в том числе на основе системного подхода;
- грамотно строить коммуникацию, исходя из целей и ситуации.

**Результатом усвоения обучающимися программы по развивающему и воспитательному аспектам являются (Soft Skills):**

- устойчивый интерес обучающихся к занятиям;
- развитие познавательных интересов обучающихся;
- навыки ведения проекта, проявление компетенции в вопросах, связанных с темой проекта, выбор наиболее эффективных решений задач в зависимости от конкретных условий;
- умение ориентироваться в информационном пространстве, продуктивно использовать техническую литературу для поиска сложных решений;
- работа в команде: работа в общем ритме, эффективное распределение задач и др.;
- проявление технического мышления, познавательной деятельности, творческой инициативы, самостоятельности;
- способность творчески решать технические задачи;
- готовность и способность применения теоретических знаний по физике, информатике для решения задач в реальном мире;
- способность правильно организовывать рабочее место и время для достижения поставленных целей.

## УЧЕБНЫЙ ПЛАН

N п/п	Название модуля, раздела, темы	Количество часов			Формы аттестации / контроля
		Всего	Теория	Практика	
	Введение	1	1	0	
1.	Введение в программирование	24	10	14	
2.	Базовые конструкции в Python	30	11	19	к/р
3.	Решение прикладных задач в Python	45	20	25	
4.	Подготовка проектных работ	4		4	
	Промежуточная аттестация	4		4	Защита проекта
	Итого	108	46	62	

## УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

№	Раздел	Количество часов			Формы аттестации/контроля	месяц	неделя
		Всего	Теор	Прак			
	<b>Введение в образовательную программу, техника безопасности</b>	<b>1</b>	<b>1</b>	<b>0</b>			
	Введение в образовательную программу, техника безопасности	1	1	0			
<b>1</b>	<b>Введение в программирование</b>	<b>24</b>	<b>10</b>	<b>14</b>			
1.1	Знакомство с системой Яндекс.Контест	2	1	1			
1.2	Условный оператор. Переменные и арифметика	2	1	1			
1.3	Знакомство с циклами	6	3	3			
1.4	Отладчик	4	2	2			
1.5	Погружение в условия. True, False, Break, Continue.	4	2	2			
1.6	Знакомство со списками	3	1	2			
1.7	Решение задач по теме модуля «Введение в программирование»	3	1	2			
<b>2</b>	<b>Базовые конструкции в Python</b>	<b>30</b>	<b>11</b>	<b>19</b>	<i>к/р</i>		
2.1	Дополнительные возможности цикла for. Кортежи. Сортировки	3	1	2			
2.2	Знакомство со срезами и диапазонами	3	1	2			
2.3	Списочные выражения. Методы split и join	3	1	2			
2.4	Другие методы списков и строк	3	1	2			
2.5	Функции	4	2	2			
2.6	Области видимости переменных	4	2	2			
2.7	Введение в компьютерную графику. Модуль TKInter	4	2	2			
2.8	Решение задач	3	1	2	<i>к/р</i>		
<b>3</b>	<b>Подготовка проектных работ</b>	<b>2</b>	<b>-</b>	<b>2</b>			

<b>4</b>	<b>Промежуточная аттестация</b>	<b>1</b>	<b>-</b>	<b>1</b>	<b>Защита проекта</b>		
	<b>За 1-е полугодие</b>	<b>55</b>	<b>26</b>	<b>29</b>			
<b>5</b>	<b>Решение прикладных задач в Python</b>	<b>53</b>	<b>22</b>	<b>27</b>	<i>к/р</i>		
5.1	Программирование и базовые конструкции	4	2	2			
5.2	Функции. Продолжение	8	4	4			
5.3	Словари и множества	8	4	4			
5.4	Обзор стандартной библиотеки Python	5	2	3			
5.5	Дополнительные библиотеки Python	10	5	5			
5.6	Введение в ООП	8	4	4			
5.7	Решение задач	6	1	5	<i>к/р</i>		
<b>6</b>	<b>Подготовка проектных работ</b>	<b>2</b>	<b>-</b>	<b>2</b>			
	<b>Промежуточная аттестация</b>	<b>2</b>	<b>-</b>	<b>2</b>	<b>Защита проекта</b>		
	<b>За 2-е полугодие</b>	<b>53</b>	<b>22</b>	<b>31</b>			
	<b>Итого:</b>	<b>108</b>	<b>48</b>	<b>60</b>			

## СОДЕРЖАНИЕ ПРОГРАММЫ

### **Введение в образовательную программу, техника безопасности.**

**Теория:** правила организации рабочего места. Правила безопасной работы.

**Практика:** Знакомство друг с другом. Заполнение журнала безопасности.

### **Модуль 1. Введение в программирование.**

#### **Блок тем 1.1. Знакомство с системой Яндекс.Контест.**

##### **Тема 1.1.1. Сервис Яндекс.Контест.**

**Теория.** Основные понятия программирования: исполнитель, система команд, алгоритм, программа, среда разработки, интерпретатор, код программы и редактор кода. Интегрированные среды, исполнение кода.

**Практика.** Знакомство с виртуальной средой взаимодействия: регистрация, организация личного кабинета, поиски выкладывание материалов. Знакомство с системой автоматизированной проверки задач. Простейшие программы с выводом на экран.

#### **Блок тем 1.2. Условный оператор. Переменные и арифметика.**

##### **Тема 1.2.1. Ввод и вывод информации.**

**Теория.** Алфавит и словарь языка Python. Объявление переменных. Типы данных. Арифметические операции. Операции с присваиванием. Линейные программы. Условный оператор.

**Практика.** Обзор типов данных. Вычисление математических выражений с помощью стандартных арифметических функций. Решение задач по теме.

#### **Блок тем 1.3. Знакомство с циклами.**

##### **Тема 1.3.1. Цикл while.**

**Теория.** Конструкция цикла while.

**Практика.** Выполнение практических заданий по теме «Цикл while».

##### **Тема 1.3.2. Циклические алгоритмы.**

**Теория.** Работа цикла.

**Практика.** Решение задач по теме «Цикл while».

##### **Тема 1.3.3. Цикл for.**

**Теория.** Конструкция цикла for.

**Практика.** Выполнение практических заданий по теме «Цикл for».

##### **Тема 1.3.4. Функции range и enumerate.**

**Теория.** Ознакомление с алгоритмом работы цикла.

**Практика.** Решение задач по теме «Цикл for».

#### **Блок тем 1.4.Отладчик.**

##### **Тема 1.4.1. Интерактивное взаимодействие с отладчиком.**

**Теория.** Определение. Установка и запуск отладчика программного кода.

**Практика.** Решение задач.

##### **Тема 1.4.2. Интеграция отладчика в программу.**

**Теория.** Проверка работы отладчика над различными программами.

**Практика.** Решение задач.

**Блок тем 1.5. Погружение в условия. True, False, Break, Continue.**

**Тема 1.5.1. Принцип работы условий.**

**Теория.** Принцип работы операторов True, False, Break, Continue.

**Практика.** Решение задач.

**Тема 1.5.2. Логические операции.**

**Теория.** Выполнение и разбор логических операций в программе.

**Практика.** Решение задач на тему «Погружение в условия».

**Блок тем 1.6. Знакомство со списками.**

**Тема 1.6.1. Что такое списки?**

**Теория.** Определение понятия «списки». Объявление списка. Действия над списками, над элементами списка.

**Практика.** Решение задач по теме «Списки».

**Тема 1.6.2. Функции списков.**

**Теория.** Рассмотрение полного функционала списков.

**Практика.** Выполнение практических заданий.

**Блок тем 1.7. Решение задач по теме модуля «Введение в программирование»**

**Тема 1.7.1. Простые задачи с переменными и условиями.**

**Тема 1.7.2. Простые задачи со списками и циклами.**

**Модуль 2. Базовые конструкции языка Python.**

**Блок тем 2.1. Дополнительные возможности цикла for. Кортежи. Сортировки.**

**Тема 2.1.1. Понятие среза и диапазона.**

**Теория.** Понятие срезов и диапазонов. Равенство и совпадение объектов.

**Практика.** Решение задач по теме «Срезы и диапазоны»

**Тема 2.1.2. Различные операции со срезами и диапазонами.**

**Теория.** Структура программы. Стандартные функции и арифметические выражения. Ветвления. Логические операции. Оператор цикла с известным количеством повторов, постусловием, предусловием. Оператор безусловного перехода.

**Практика.** Выполнение практических заданий.

**Блок тем 2.3. Списочные выражения. Методы split и join**

**Тема 2.3.1. Метод split.**

**Теория.** Работа со списками. Применение метода split.

**Практика.** Решение задач на тему «Методы split».

**Тема 2.3.2. Метод join.**

**Теория.** Работа со списками. Применение метода join.

**Практика.** Решение задач на тему «Методы join».

**Блок тем 2.4. Другие методы списков и строк**

**Тема 2.4.1. Методы списков и строк**

**Теория.** Генераторы списков. Методы find и rfind. Метод replace и count.

**Практика.** Решение задач на методы списков и строк.

**Тема 2.4.2. Углубленное изучение методов списков и строк.**

*Теория.* Функции и методы строк. Разбор таблицы.

*Практика.* Практическое задание с использованием функций и сетодов строк.

**Блок тем 2.5. Функции.**

**Тема 2.5.1. Определение функции.**

*Теория.* Определение функции. Правила составления и вызова функций в программном коде.

*Практика.* Выполнение практических заданий.

**Тема 2.5.2. Функции. Локальные и глобальные переменные.**

*Теория.* Локальные и глобальные переменные.

*Практика.* Практические задания.

**Тема 2.5.2. Анонимные функции.**

*Теория.* Анонимные функции. Понятие. Действия с ними.

*Практика.* Практические задания.

**Блок тем 2.6. Области видимости переменных.**

**Тема 2.6.1. Локальная область видимости.**

*Теория.* Локальная область видимости переменных. Нелокальная область видимости переменных.

*Практика.* Решение задач на тему «Области видимости переменных».

**Тема 2.6.2. Глобальная область видимости.**

*Теория.* Глобальная область видимости переменных.

*Практика.* Решение задач на тему «Области видимости переменных».

**Тема 2.6.3. Именованые переменных.**

*Теория.* Подбор имен переменных. Использование дебаггера в коде.

*Практика.* Решение задач на тему «Области видимости переменных».

**Блок тем 2.7. Введение в компьютерную графику. Модуль TKInter.**

**Тема 2.7.1. Обзор графического интерфейса.**

*Теория.* Установка и обзор графического интерфейса.

*Практика.* Выполнение заданий в графическом интерфейсе «Модуль TKInter».

**Тема 2.7.2. Создание GUI-интерфейса.**

*Теория.* Компьютерная графика и её виды. Кроссплатформенная библиотека для разработки графического интерфейса TKInter.

*Практика.* Решение задач на тему «Модуль TKInter».

**Тема 2.7.3. Виджеты.**

*Теория.* Обзор дополнительного контента в графическом интерфейсе.

*Практика.* Выполнение практических заданий по теме «Виджеты».

**Блок тем 2.8. Решение задач.**

*Практика.* Решение задач на темы модулей «Введение в программирование» и «Базовые конструкции в Python».

**Модуль 3. Подготовка проектных работ.**

*Практика.* Выбор проекта, разработка игры, разработка сайта и т.д.

**Промежуточная аттестация:** к/р

**II полугодие: Модуль 4. Решение прикладных задач в Python.**

## **Блок тем 4.1. Программирование и базовые конструкции.**

### **Тема 4.1.1. Условные операторы. Операторы цикла**

*Теория.* Условные операторы. Операторы цикла. Повторение пройденного материала по модулям 1,2.

*Практика.* Решение задач.

### **Тема 4.1.2. Строки, списки, множества и функции.**

*Теория.* Строки, списки, множества и функции. Повторение пройденного материала по модулям 1,2.

*Практика.* Решение задач.

## **Блок тем 4.2. Функции. Продолжение.**

### **Тема 4.2.1. Синтаксис объявления функции.**

*Теория.* Понятие функции. Именные функции, инструкция def. Аргументы функции.

*Практика.* Решения задач на тему «Функции и функциональная парадигма программирования».

### **Тема 4.2.2. Вызов функции.**

*Теория.* Определение и вызов простых функций.

*Практика.* Выполнение практических заданий на тему «Вызов функции».

### **Тема 4.2.3. Функция с переменным числом аргументов.**

*Теория.* Определение и разбор функций с переменным числом аргументов.

*Практика.* Выполнение практических заданий на тему 4.2.3.

### **Тема 4.2.4. Синтаксис анонимных функций.**

*Теория.* Определение анонимных функций, разбор синтаксиса.

*Практика.* Выполнение практических заданий на тему «Синтаксис анонимных функций».

### **Тема 4.2.5. Возврат значений. Рекурсия.**

*Теория.* Определение функции return, использование.

*Практика.* Выполнение практических заданий на тему 4.2.5.

### **Тема 4.2.6. Пустая функция.**

*Теория.* Определение. Разбор синтаксиса.

*Практика.* Выполнение практических заданий на тему «Пустая функция».

### **Тема 4.2.7. Основные встроенные функции.**

*Теория.* Разбор основных встроенных функций. Определения.

*Практика.* Выполнение практических заданий по теме 4.2.7.

### **Тема 4.2.8. Функциональная парадигма программирования.**

*Теория.* Функциональная парадигма программирования. Основы.

*Практика.* Выполнение практических заданий на тему 4.2.8.

## **Блок тем 4.3. Словари и множества**

### **Тема 4.3.1. Создание и ознакомление со словарями.**

*Теория.* Словари (dict) и работа с ними. Методы словарей.

*Практика.* Решение задач на тему «Словари и множества».

### **Тема 4.3.2. Объединение словарей.**

*Теория.* Слияние двух и более словарей.

*Практика.* Решение задач.

### **Тема 4.3.3. Операции с элементами словарей.**

*Теория.* Создание словаря с различными элементами словарей.  
Операции с элементами словарей.

*Практика.* Решение задач.

### **Тема 4.3.4. Методы множества.**

*Теория.* Множества (set и frozenset).

*Практика.* Решение задач.

### **Тема 4.3.5. Сравнение.**

*Теория.* Сравнение работы со словарями и множествами.

*Практика.* Решение задач.

### **Тема 4.3.6. Операции с элементами множеств.**

*Теория.* Выполнение операций с элементами множеств.

*Практика.* Решение задач.

## **Блок тем 4.4. Обзор стандартной библиотеки Python.**

### **Тема 4.4.1. Стандартная библиотека Python.**

*Теория.* Работа с модулями: создание, подключение инструкциями import и from. Модуль os, sys. Модуль contextlib. Модуль abc. Модуль datetime. Модуль collections.

*Практика.* Решение задач на тему «Стандартная библиотека Python»

### **Тема 4.4.2. Использование стандартной библиотеки Python.**

*Теория.* Использование. Обработка текстов. Полезные константы и функции модуля string.

*Практика.* Решение задач на тему «Использование стандартной библиотеки Python»

### **Тема 4.4.3. Форматы данных стандартной библиотеки Python.**

*Теория.* CSV. Определение. Работа с различными файловыми архивами. Конфигурационные файлы.

*Практика.* Выполнение практических заданий.

## **Блок тем 4.5. Дополнительные библиотеки Python**

### **Тема 4.5.1. Нейросетевая библиотека.**

*Теория.* Keras. Нейросетевая библиотека. Определение. Применение.

*Практика.* Выполнение практических заданий.

### **Тема 4.5.2. Техническая вычислительная библиотека.**

*Теория.* NumPy. Определение. Разбор. Применение.

*Практика.* Выполнение практических заданий.

### **Тема 4.5.3. Библиотека обработки изображений**

*Теория.* Pillow. Определение. Обзор. Применение.

*Практика.* Выполнение практических заданий.

### **Тема 4.5.4. Библиотека разработки игр**

*Теория.* PYGLET. Определение. Разбор. Применение.

*Практика.* Выполнение практических заданий.

### **Тема 4.5.5. Библиотека HTTP.**

*Теория.* Requests. Определение. Разбор. Применение.

**Практика.** Выполнение практических заданий.

**Тема 4.5.6. Библиотека машинного обучения.**

**Теория.** TensorFlow. Определение. Разбор. Применение.

**Практика.** Выполнение практических заданий.

**Блок тем 4.6. Введение в ООП.**

**Тема 4.6.1. Преимущества и недостатки ООП.**

**Теория.** Объектно-ориентированное программирование.

**Практика.** Решение задач на тему «Введение в ООП».

**Тема 4.6.2. ООП. Класс, объекты и методы.**

**Теория.** Определение класса. Создание класса. Определение объекта. Создание объектов из класса. Атрибуты класса. Методы. Виды.

**Практика.** Решение задач на тему «ООП. Класс, объекты и методы».

**Тема 4.6.3. ООП. Полиморфизм и наследование.**

**Теория.** Определение полиморфизма. Перегрузка метода. Переопределение метода. Определение наследования. Множественное наследование.

**Практика.** Решение задач по теме «ООП. Полиморфизм и наследование».

**Тема 4.6.4. ООП. Инкапсуляция.**

**Теория.** Определение. Скрытие данных. Определение атрибутов.

**Практика.** Решение задач по теме «ООП. Инкапсуляция».

**Тема 4.6.5. Общее подведение итога по ООП.**

**Теория.** Повторение изученного материала по объектно-ориентированному программированию.

**Практика.** Решение задач.

**Блок тем 4.7. Решение задач.**

Решение задач по теме модуля 4.

**Модуль 5. Подготовка проектных работ.**

**Практика.** Последние подготовки перед защитой.

**Промежуточная аттестация:** защита проекта.

### **Ожидаемые результаты**

В результате изучения курса получают дальнейшее развитие *личностные, регулятивные, коммуникативные и познавательные универсальные учебные действия, учебная (общая и предметная) и общепользовательская ИКТ-компетентность обучающихся.*

В основном формируются и получают развитие *метапредметные результаты*, такие как:

- умение самостоятельно планировать пути достижения целей, осознанно выбирать наиболее эффективные способы решения учебных и познавательных задач;

- умение соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности в процессе достижения результата;

- умение оценивать правильность выполнения учебной задачи, собственные возможности ее решения;

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

- владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности;

- умение организовывать учебное сотрудничество совместную деятельность с учителем и сверстниками;

- формирование и развитие компетентности в области использования информационно коммуникационных технологий (далее ИКТ-компетентности).

Вместе с тем вносится существенный вклад в развитие *личностных результатов*, таких как:

- формирование ответственного отношения к учению;

- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками, детьми старшего и младшего возраста, взрослыми в процессе образовательной, творческой и других видов деятельности.

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

- на формирование умений формализации и структурирования информации, умения выбирать способ представления данных в соответствии с поставленной задачей с использованием соответствующих программных средств обработки данных;

- формирование навыков и умений безопасного и целесообразного поведения при работе с компьютерными программами и в Интернете, умение соблюдать нормы информационной этики и права.

- умеют составлять и записывать алгоритмы с использованием соответствующих алгоритмических конструкций;

- умеют распознавать необходимость применения той или иной алгоритмической конструкции при решении задачи;

- умеют организовывать данные для эффективной алгоритмической обработки;

- умеют разрабатывать алгоритмы и реализовывать их на языке программирования Python;

- умеют осуществлять отладку и тестирование программы

### Форма контроля за уровнем достижения учащихся

Предметом диагностики и контроля являются составленные алгоритмы и программы на языке программирования Python к предложенным задачам.

Оценка имеет различные способы выражения – устные суждения педагога, письменные качественные характеристики, систематизированные по заданным параметрам аналитические данные, в том числе и рейтинги. Оценке подлежит в первую очередь уровень достижения учеников минимально необходимых результатов, обозначенных в целях и задачах курса.

Качество знаний и умений ученика оценивается следующими характеристиками:

- знание основных алгоритмических конструкций;
- умение составить и записать алгоритм с использованием соответствующей алгоритмической конструкции;
- умение найти более эффективный способ решения задачи;
- умение тестировать программу.

В течение всего курса проводятся контрольные срезы и выставляются баллы за решенные задачи. По окончании курса **зачет** получают те учащиеся, у которых сумма баллов за учебный курс составляет не менее 60% от всей суммы баллов.

### Условия реализации программы

Аспекты	Характеристика (заполнить)
Материально-техническое обеспечение	<p><b>Программное обеспечение:</b></p> <ul style="list-style-type: none"> <li>– ОС Windows</li> <li>– Текстовый редактор Блокнот</li> <li>– Microsoft Office</li> </ul> <p><b>Презентационное оборудование:</b></p> <ul style="list-style-type: none"> <li>– Интерактивная доска – 1 шт.</li> <li>– Персональный компьютер – 10 шт.</li> </ul> <p><b>Дополнительное оборудование:</b></p> <ul style="list-style-type: none"> <li>– учительский стол – 1 шт.</li> <li>– учительский стул – 1 шт.</li> <li>– компьютерный стол -10шт.</li> <li>– кресла ученические – 10 шт.</li> </ul>
Информационное обеспечение	<ul style="list-style-type: none"> <li>- аудио</li> <li>- видео</li> <li>- фото</li> </ul>

Аспекты	Характеристика (заполнить)
	- интернет источники - курс на платформе «Яндекс-учебник» - тренажер «Гимнастика для глаз»
Кадровое обеспечение	Педагог ДО, высшая квалификация

### Литература и интернет-ресурсы

1. Ушаков Д.М., Юркова Т.А. – Python для школьников. –СПб.: Питер, 2006г. – 256с.:
2. Чернов А.Ф. – Олимпиадные задачи с решениями и подробным анализом. – Волгоград: Учитель, 2007. – 207с.:
3. Рапаков Г.Г., Ржеуцкая С.Ю. – Программирование на Python 3/0 для студентов и школьников. – СПб.:БХВ-Петербург, 2007.-352 с.:
4. Цветков А.С. – Язык программирования PASCAL Система программирования ABC Pascal. Учебное пособие для школьников, Санкт-Петербург, 2015. -46с.
5. Ускова О.Ф. – Программирование на языке Питон Задачник. Изд. Питер. 2002. -336с.
6. Потопахин В.В. Turbo Pascal. Решение сложных задач. Издательство: «БХВ-Петербург», 2006, - 12с.
7. <http://pascalabc.net/o-yazike-paskal>
8. <https://ru.wikibooks.org/wiki/PascalABC.Net>
9. <http://kpolyakov.spb.ru/school/probook/pascal.htm>
10. <http://pas1.ru/pascaltextbook>
11. <http://zedpost.ru/ispas/>