



ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ СРЕДНЯЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ШКОЛА № 475  
ВЫБОРГСКОГО РАЙОНА САНКТ-ПЕТЕРБУРГА

Принята на заседании  
педагогического совета  
ГБОУ СОШ № 475 Выборгского района  
Санкт-Петербурга  
Протокол № 7 от 30.08.2023 г.

Утверждена директором  
ГБОУ СОШ № 475  
Выборгского района Санкт-Петербурга  
Овечкин А.В  
Приказ № 09/ба от 30.08.2023 г.

**РАБОЧАЯ ПРОГРАММА**  
**Сетевая программа**  
**«программирования на языке Python»**

Возраст: 11-17 лет  
Срок реализации 5 месяцев

Санкт-Петербург  
2023

## **Основные характеристики программы**

### **Пояснительная записка**

Программа позволяет обучающимся познакомиться с профессиональным языком программирования Python и получить опыт решения тематических задач. Изучая базовый синтаксис Python, а впоследствии тематические модули и библиотеки, обучающиеся начнут формировать портфолио разработчика и развивать Soft skills — навыки, необходимые для работы любого специалиста будущего: умение непрерывно учиться новому, использование передовых технологий, работа в команде, гибкость и способность к адаптации, внимательность, самостоятельность инициативность, способность к рефлексии и высокий уровень критического мышления. Изучение базовой части курса проходит на образовательной платформе «Алгоритмика», которая предоставляет доступ к теоретическим материалам, задачам с автоматической проверкой и инструкциям для выполнения проектных заданий.

Проектная часть курса изучается в профессиональной среде разработки программ Visual Studio Code от Microsoft. В этой среде учащиеся реализуют такие проекты как интерактивные платформенно-независимые игры и десктопные приложения.

Для контроля образовательных результатов учащихся используются задания на платформе с автопроверкой (все задания на тему базовых понятий проверяются системой автоматически) Они направлены на формирование навыков написания программ на языке Python и умения читать код, вносить в него изменения

(исправления) и дополнять.

Кроме расширенной проверки усвоения предусмотрены ручное ревью кода (выборочная оценка кода учеников на соблюдение стандартов его написания) и презентации проектов, на которых возможна взаимооценка и проверка.

Курс поможет тем, кто интересуется техническими науками и хочет освоить мощный язык программирования, но пока только поверхностно знаком с основами программирования.

Учащиеся изучат в ускоренном темпе модули по основам языка, включая синтаксис, переменные, условные операторы, циклы и функции, а также освоят объектно-ориентированное программирование. Далее они перейдут к изучению проектных модулей, в рамках которых реализуют два проекта: создание десктопного приложения и разработку 2D игры.

### **Описание планируемых результатов обучения**

В результате освоения программы ученик: изучит основы работы алгоритмов и их применение

- отработает навыки применения основных логических

операций, такие как AND, OR, NOT в условных выражениях;

- изучит основные конструкции управления потоком данных, такие как if-else, while и for;
- научится работать с встроенными модулями Python, например, модулем random для работы с случайными числами или math для математических функций;
- научится использовать в программах основные структуры данных Python: списки, словари, кортежи, множества;
- научится использовать и создавать собственные классы в Python для создания структурированного кода;
- научится планировать этапы собственного проекта;
- изучит основы создания 2D-игр с помощью библиотеки Pygame и способ сборки 2D-игры в файл .exe;
- изучит основы создания оконных приложений инструментов для их создания с помощью библиотеки PyQt5;
- научится готовить презентацию и устную речь для представления собственного проекта;

### **Актуальность программы**

Программа курса раскрывает область цифровых компетенций, связанную с программированием и созданием IT-продуктов, и предполагает знакомство с конструкциями языка (синтаксис, основные алгоритмические конструкции, функции и модули, объектно-ориентированное программирование), а также работу с библиотеками, выходящими за рамки стандарта языка, — Pygame и PyQt5. Кроме того, ученики научатся работе в рамках профессионального подхода с помощью системы контроля версий Git и используемого разработчиками по всему миру сервиса GitHub.

### **Условия формирования групп**

**Условия набора в коллектив:** принимаются все желающие в возрасте от 11 до 17 лет без конкурсного отбора и минимального набора знаний. Базовые знания не требуются. Противопоказания по состоянию здоровья отсутствуют. Решение о наборе принимается в соответствии с локальными актами образовательной организации.

**Количество учащихся в группе:** оптимальный состав учебной группы до 14 человек.

**Рекомендации по набору детей в группу.** Рекомендовано базовые навыки владения компьютером. Умение пользоваться офисными пакетами и средствами поиска информации.

**Условия формирования групп:** ученики в возрасте от 11 до 17 лет включительно. На первый год обучения принимаются все желающие,

формируются разновозрастные группы. допускается дополнительный набор учащихся на основе собеседования в соответствии с локальными актами образовательной организации.

**Режим занятий.** Форма проведения занятий: очная, 2 раза в неделю по 2 академических часа с перерывом 10-15 минут. Один академический час составляет: для детей – 45 минут.

#### **Объем и срок реализации программы**

Продолжительность освоения программы составляет 144 часов.

Программа рассчитана на 1 год.

#### **Формат обучения**

Очная форма без применения дистанционных образовательных технологий.

**Цель программы** - Предоставить ученикам возможность для развития и закрепления навыков программирования на языке Python, основываясь на углубленном понимании его базового синтаксиса и дальнейшей реализации собственных проектов.

## Рабочая программа с описанием каждого модуля

Модуль (описание)	Тема	Содержание	Вид учебных занятий	Объем в ак.ч.
<b>Модуль 1. Основы языка, управляющие конструкции</b>  Изучение основ языка Python: базовые функции, типы данных, правила оформления кода, условные операторы, циклы	<b>Тема 1.</b> Введение в язык Python	Изучение понятий: «язык программирования», «алгоритм», «программа». Знакомство с синтаксисом языка. Функция ввода input(). Функция вывода print(). Необходимость вывода данных.	теоретические занятия	1
		Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программы, осуществляющей ввод-вывод данных, ввод команд с клавиатуры.	практические занятия	1
		Изучение теоретического материала на платформе. Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.	самостоятельная работа	1
	<b>Тема 2.</b> Переменные	Изучение понятий: данные, типы данных, переменная, числовой тип данных, оператор присваивания, приоритет операций.	теоретические занятия	1
		Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих обработку переменных числового типа. Изменение типов данных с помощью функций int() и str()	практические занятия	1

		Изучение теории для следующего занятия.	самостоятельная работа	1
	<b>Тема 3.</b> Строки	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих обработку строковых переменных (сумма строк, повтор строки n-раз, вычисление длины строки, получение символа строки по номеру, замена подстроки на другую подстроку, разбиение строки по разделителю).	практические занятия	2
		Изучение теории для следующего занятия.	самостоятельная работа	1
	<b>Тема 4.</b> Вложенные конструкции	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, содержащих вложенные конструкции.	практические занятия	2
		Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.	самостоятельная работа	1
	<b>Тема 5.</b> Практическое занятие по темам: "Переменные", "Строки", "Вложенные конструкции"	Закрепление пройденного материала. Выполнение автопроверяемых заданий на платформе по пройденной теме.	практические занятия	2
		Изучение теории для следующего занятия.	самостоятельная работа	1
	<b>Тема 6.</b>	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, содержащих	практические занятия	2

	Условный оператор	условный оператор.		
		Изучение теории для следующего занятия.	самостоятельная работа	1
	<b>Тема 7.</b> Вложенный условный оператор	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, содержащих вложенный условный оператор.	практические занятия	2
		Изучение теоретического материала на платформе. Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.	самостоятельная работа	1
	<b>Тема 8.</b> Практическое занятие по темам: "Условный оператор", "Вложенный условный оператор"	Закрепление пройденного материала. Выполнение автопроверяемых заданий на платформе по пройденной теме.	практические занятия	2
		Изучение теории для следующего занятия.	самостоятельная работа	1
	<b>Тема 9–10.</b> Циклы	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, содержащих циклы различных типов.	практические занятия	4
		Изучение теории для следующего занятия.	самостоятельная работа	2
	<b>Тема 11.</b>	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ,	практические занятия	2

	Вложенные управляющие конструкции	осуществляющих обработку данных с помощью вложенных алгоритмических конструкций.			
		Подготовка к аттестации по модулю.	самостоятельная работа	2	
	Прохождение аттестации по модулю.			2	
				<b>Объем в ак.ч.</b>	<b>Объем в %</b>
<b>ИТОГО:</b>			теоретические занятия	2	6%
			практические занятия	20	56%
			самостоятельная работа	12	33%
			аттестация	2	
			<b>Всего:</b>	<b>36</b>	

Модуль (описание)	Тема	Содержание	Вид учебных занятий	Объем в ак.ч.
Модуль 2. Функции, модуль Turtle  Изучение правил создания собственных функций и модулей на	Тема 1. Практикум по решению задач	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, содержащих материалы всего модуля.	практические занятия	2
		Изучение теории для следующего занятия.	самостоятельная работа	1
	Тема 2–3.	Изучение понятий: «функция», «локальная и глобальная функция», «оператор def», «параметры функции»,	теоретические занятия	1



<p>языке Python. Изучение модуля Turtle для создания графических объектов</p>	<p>Функции</p>	<p>«значение функции». Изучение синтаксиса создания и вызова функции. Использование одной функции внутри другой.</p>		
		<p>Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих обработку данных с помощью функций и содержащих функцию внутри другой функции.</p>	<p>практические занятия</p>	<p>3</p>
		<p>Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.</p>	<p>самостоятельная работа</p>	<p>2</p>
	<p><b>Тема 4.</b> Практическое занятие по теме "Функции"</p>	<p>Закрепление пройденного материала. Выполнение автопроверяемых заданий на платформе по пройденной теме.</p>	<p>практические занятия</p>	<p>2</p>
		<p>Изучение теории для следующего занятия.</p>	<p>самостоятельная работа</p>	<p>1</p>
	<p><b>Тема 5–6.</b> Модули</p>	<p>Изучение понятий: «модуль», «встроенные модули стандартной библиотеки». Использование модулей random и time. Синтаксис создания и подключения модуля.</p>	<p>теоретические занятия</p>	<p>1</p>
		<p>Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих обработку данных с помощью модулей, встроенных в стандартную библиотеку, и модулей, не входящих в стандартную библиотеку.</p>	<p>практические занятия</p>	<p>3</p>

		Изучение теоретического материала на платформе. Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.	самостоятельная работа	2
	<b>Тема 7.</b> Практическое занятие по теме "Модули"	Закрепление пройденного материала. Выполнение автопроверяемых заданий на платформе по пройденной теме.	практические занятия	2
		Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме. Изучение теории для следующего занятия.	самостоятельная работа	1
	<b>Тема 8.</b> Хакатон	Выполнение заданий на платформе. Создание консольного приложения на заданную тему, используя инструменты изученные за модуль.	практические занятия	2
		Изучение теории для следующего занятия.	самостоятельная работа	1
	<b>Тема 9.</b> Turtle. Линейные алгоритмы	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих рисование геометрических фигур с использованием условного оператора.	практические занятия	2
		Изучение теории для следующего занятия.	самостоятельная работа	1
	<b>Тема 10.</b> Turtle. Циклы	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих рисование многоугольников.	практические занятия	2
			самостоятельная работа	1

	<b>Тема 11.</b> Turtle. Вложенные структуры данных	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих рисование геометрических фигур с использованием условного оператора.	практические занятия	2	
		Изучение теоретического материала на платформе. Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.	самостоятельная работа	2	
	Прохождение аттестации по модулю.			2	
				<b>Объем в ак.ч.</b>	<b>Объем в %</b>
<b>ИТОГО:</b>			теоретические занятия	2	6%
			практические занятия	20	56%
			самостоятельная работа	12	33%
			аттестация	2	
			<b>Всего:</b>	<b>36</b>	

Модуль (описание)	Тема	Содержание	Вид учебных занятий	Объем в ак.ч.
<b>Модуль 3. Объектно-ориентированное программирование</b>	<b>Тема 1.</b> Проект «Городская»	Выполнение заданий на платформе по планированию и реализации проекта. Презентация реализованного проекта.	практические занятия	2

Изучение основ объектно-ориентированного программирования. Реализация проектов в стиле ООП	среда»	Кастомизация реализованного проекта. Повторение материала, изученного в рамках модуля.	самостоятельная работа	1
	<b>Тема 2.</b> ООП. Объекты и методы	Изучение понятий: «объекты», «поля и методы объектов». Синтаксис создания объектов.	теоретические занятия	1
		Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих обработку данных с помощью передачи объекта в функцию.	практические занятия	1
		Изучение теоретического материала на платформе. Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.	самостоятельная работа	1
	<b>Тема 3.</b> Практическое занятие по доработке проекта "Черепашьи бега"	Закрепление пройденного материала. Выполнение автопроверяемых заданий на платформе по пройденной теме.	практические занятия	2
		Выполнение автопроверяемых заданий на платформе по пройденной теме.	самостоятельная работа	1
	<b>Тема 4.</b> ООП. События	Изучение типов событий в программе. Обработка взаимодействия пользователя и программы. Обработка событий мыши.	теоретические занятия	1
		Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих	практические занятия	1

		обработку событий.		
		Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода.	самостоятельная работа	1
	<b>Тема 5.</b> Практическое занятие по доработке проекта "Поймай черепашку"	Закрепление пройденного материала. Выполнение автопроверяемых заданий на платформе по пройденной теме.	практические занятия	2
		Изучение теории для следующего занятия.	самостоятельная работа	1
	<b>Тема 6-7</b> ООП. Проект Simple Paint 2035	Выполнение заданий на платформе по планированию и реализации проекта. Презентация реализованного проекта.	практические занятия	2
		Кастомизация реализованного проекта. Изучение теории для следующего занятия.	самостоятельная работа	2
	<b>Тема 8.</b> Классы	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, осуществляющих обработку классов.	практические занятия	2
		Изучение теории для следующего занятия.	самостоятельная работа	1
	<b>Тема 9–10.</b> Наследование	Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ, использующих наследование.	практические занятия	4

		Изучение теоретического материала на платформе. Решение задач, которые включают в себя тестовые задания и задание на написание кода.	самостоятельная работа	2	
	<b>Тема 11.</b> Практическое занятие по темам "Классы", "Наследование"	Закрепление пройденного материала. Выполнение автопроверяемых заданий на платформе по пройденной теме.	практические занятия	2	
		Изучение теории для следующего занятия.	самостоятельная работа	1	
	Прохождение аттестации по модулю.				2
				<b>Объем в ак.ч.</b>	<b>Объем в %</b>
<b>ИТОГО:</b>			теоретические занятия	2	6%
			практические занятия	20	56%
			самостоятельная работа	12	33%
			аттестация	2	
			<b>Всего:</b>	<b>36</b>	

Модуль (описание)	Тема	Содержание	Вид учебных занятий	Объем в ак.ч.
	<b>Тема 1–2.</b>	Библиотека Rугame. Спрайт. Игровой цикл: обработка событий, обновление игры, прорисовка, подсчет времени.	теоретические занятия	1

<b>Модуль 4. Разработка игр на Pygame</b>  Изучение библиотеки Pygame для создания 2D-игр. Реализация и презентация итоговых проектов. Подведение итогов обучения на командном хакатоне	Знакомство с Pygame. События	Расположение спрайта. Обработка списков.		
		Выполнение автопроверяемых заданий на платформе по пройденной теме. Разработка программ для создания и расположения спрайтов. Разработка программ, осуществляющих обработку списков.	практические занятия	3
		Самостоятельное выполнение автопроверяемых заданий на платформе по пройденной теме, которые включают в себя тестовые задания и задание на написание кода. Изучение теории для следующего занятия.	самостоятельная работа	2
	<b>Тема 3.</b> Практическое занятие по доработке проекта "Черепашьи бега"	Закрепление пройденного материала. Выполнение автопроверяемых заданий на платформе по пройденной теме.	практические занятия	2
		Выполнение автопроверяемых заданий на платформе по пройденной теме.	самостоятельная работа	1
	<b>Тема 4–7.</b> Игра Fast Clicker	Разбор технического задания. Планирование проекта. Игровая сцена, игровой таймер, частота кадров, отображение текста, создание спрайта-карточки. Элементы интерфейса для счетчиков статистики. Условия победы и проигрыша.	теоретические занятия	1
		Разработка игры Fast Clicker. Обработка событий мыши. Работа с модулем time: отображение надписи на случайном спрайте ограниченное количество времени.	практические занятия	7

		Самостоятельная доработка проекта разработанного в рамках урока. Изучение теории для следующего занятия.	самостоятельная работа	4	
	Тема 8–11. Игра «Арканоид»	Разработка игры «Арканоид». Работа с большим количеством спрайтов. Автоматическое движение спрайта. Обработка нажатий на клавиши. Презентация проекта.	практические занятия	8	
		Самостоятельная доработка проекта, разработанного в рамках урока.	самостоятельная работа	5	
	Прохождение аттестации по модулю.			2	
				<b>Объем в ак.ч.</b>	<b>Объем в %</b>
<b>ИТОГО:</b>			теоретические занятия	2	6%
			практические занятия	20	56%
			самостоятельная работа	12	33%
			аттестация	2	
			<b>Всего:</b>	<b>36</b>	

### Календарно-тематическое планирование

№	№ модуля и тема	Тема занятия	Кол-во занятий*	Кол-во часов
1		Введение в язык Python	1	2



2	Модуль 1. Основы языка, управляющие конструкции	Переменные	1	2
3		Строки	1	2
4		Вложенные конструкции	1	2
5		Практическое занятие по темам: "Переменные", "Строки", "Вложенные конструкции"	1	2
6		Условный оператор	1	2
7		Вложенный условный оператор	1	2
8		Практическое занятие по темам: "Условный оператор", "Вложенный условный оператор"	1	2
9		Циклы	2	4
10		Вложенные управляющие конструкции	1	2
11		Аттестация	1	2
12	Модуль 2. Функции, модуль Turtle	Практикум по решению задач	1	2

13		Функции	2	4
14		Практическое занятие по теме "Функции"	1	2
15		Модули	2	4
16		Практическое занятие по теме "модули"	1	2
17		Хакатон	1	2
18		Turtle. Линейные алгоритмы	1	2
19		Turtle. Циклы	1	2
20		Turtle. Вложенные структуры данных	1	2
21	Аттестация		1	2
22	Модуль 3. Объектно-ориентированное программирование	Проект «Городская среда»	1	2
23		ООП. Объекты и методы	1	2

24		Практическое занятие по доработке проекта "Черепашьи бега"	1	2
25		ООП. События	1	2
26		Практическое занятие по доработке проекта "Поймай черепашку"	1	2
27		ООП. Проект Simple Paint	2	4
28		Классы	1	2
29		Наследование	2	4
30		Практическое занятие по темам "Классы", "Наследование"	1	2
31	Аттестация		1	2
32	Модуль 4. Разработка игр на Pygame	Знакомство с Pygame. События	2	4
33		Практическое занятие по доработке проекта "Черепашьи бега"	1	2
34		Игра Fast Clicker	4	8

35		Игра «Арканоид»	4	8
36	Аттестация		1	2

**\*Количество занятий не включает часы, отведенные на самостоятельное изучение, и часы, отведенные на прохождение аттестации**

## **Методы и формы обучения**

На протяжении всего курса используются смешанные формы и методы обучения.

- Лекционный формат.
- Устный опрос.
- Тестирование.
- Практические работы.
- Дискуссия.
- Презентация учениками результатов своей работы.
- Индуктивный и дедуктивный методы.
- Соревнования и турниры.

## **Педагогические технологии**

- Проектная деятельность: курс включает несколько индивидуальных мини-проектов с явной координацией со стороны учителя.
- Проблемное обучение: каждый урок построен на решении конкретной проблемы, которая может возникнуть в процессе разработки игры.
- Групповое обучение: курс включает работу в парах, во время которой один ученик даёт обратную связь по проекту другого.
- Индивидуализация обучения: задания и теоретический материал доступны ученикам в любое время.
- Модульное обучение.
- Игровые технологии.

## **Методические разработки**

Размещенные на образовательной платформе поурочные методические рекомендации к занятиям. Тематические презентации. Учебные задачи для отработки содержания изучаемой темы.

## **Материалы модуля**

Каждый из представленных на платформе модулей содержит несколько уроков. Каждый урок содержит в себе несколько блоков заданий, объединённых одной из изученных на уроке тем. Большая часть заданий проверяется автоматически. Во время урока преподаватель объясняет ученикам тему урока, подкрепляя объяснения графическими материалами и примерами кода, обсуждает тему с учениками, вовлекая их в диалог и отвечает на появившиеся у ученика вопросы. После этого ученик переходит на платформу и решает задания из соответствующего блока заданий при поддержке преподавателя. В любой момент времени (на уроке, во время практической части или самостоятельно работы, вне урока) ученик имеет доступ ко всем представленным в пройденных уроках заданиям. Также ученик имеет доступ к теоретическому материалу урока и в любой момент времени может самостоятельно повторить пройденные темы.

## Оценочные материалы

Итоговая аттестация проводится по окончании обучения по программе. В течение курса предполагаются регулярные зачеты. При этом тематические состязания также являются методом проверки, и успешное участие в них освобождает от соответствующего зачета.

### Формы контроля

Устные опросы; автопроверяемые задания; проектные задания.

### Диагностические инструменты

Показатели решаемости на платформе. Фиксация преподавателем реализованных этапов проекта на платформе. Обратная связь от учеников по каждому модулю.

Показатели и критерии оценивания

Процент решенных заданий на платформе. Реализованный проект. Примеры заданий

*Устный опрос:*

- За что отвечает оператор def? • Что такое функция?

*Автопроверяемые задания:*

- Отметь верные утверждения о функциях в Python.
- Напиши программу для печати информации об учениках курса.

*Проект:*

- Запрограммировать текстовый квест с использованием принципов объектно-ориентированного программирования.

Запрограммировать игру «Лабиринт» по предоставленному техническому заданию.

Шкала оценивания, нижнее значение: 0 Шкала оценивания, верхнее значение: 100

Шкала оценивания, минимальный проходной балл: 60

### Календарный учебный график

Занятия в объединении начинаются со 1 сентября и заканчиваются 30 июня и составляют 18 недель.

Занятия проводятся 2 раза в неделю по 2 часа. В каникулярное время занятия проводятся по обычному режиму.

Сроки проведения итоговой аттестации: Последняя неделя июня.

## **Список литературы**

1. Брайсон Пэйн. Python для детей и родителей. — ООО «Эксмо», 2017. —352 с.
2. Вордерман Кэрол, Джон Вудкок, Шон Макаманус. Программирование для детей. Иллюстрированное руководство для детей; пер. с англ. С. Ломакина. — 4-е изд. М.: Манн, Иванов и Фербер, 2018, — 224 с.
3. Джейсон Бриггс. Python для детей. Самоучитель по программированию. пер. с англ. С. Ломакина. — Манн, Иванов и Фербер, 2022, — 320 с.
4. Лапчик М.П., Семакин И.Г., Хеннер Е.К., Самылкина Н.Н. Теория и методика обучения информатике. М.: Академия, 2008. — 592 с.
5. Фридман Л.М. Как научиться решать задачи. — М.: Просвещение, 1989