

ДЕПАРТАМЕНТ ОБРАЗОВАНИЯ ГОРОДА МОСКВЫ

ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ГОРОДА МОСКВЫ  
«ШКОЛА С УГЛУБЛЕННЫМ ИЗУЧЕНИЕМ ИНОСТРАННЫХ ЯЗЫКОВ  
№ 1288 ИМЕНИ ГЕРОЯ СОВЕТСКОГО СОЮЗА Н.В. ТРОЯН»  
(ГБОУ ШКОЛА № 1288)

Хорошевское шоссе, д.3, Москва, 123007


Телефон: (495) 941-29-22, (495) 945-72-95, факс: (495) 941-29-22

E-mail: 1288@edu.mos.ru

ОКПО 14174287, ОГРН 1127747146361, ИНН/КПП 7714890087/771401001

«Рассмотрено»

Методист по  
дополнительному  
образованию

 /Одаренко Л.В./

«Согласовано»

Заместитель директора  
ГБОУ Школа № 1288

 /Лебедева С.Е./

«Утверждаю»

Директор  
ГБОУ Школа № 1288

 /Мартынова Е.В./



**РАБОЧАЯ ПРОГРАММА**

Направленность: Техническая

Название: «Юный программист»

Уровень: ознакомительный

Срок реализации: 1 учебный год

Количество часов обучения: 54 часа

Возраст учащихся: 9-11 лет (3-4 класс)

Программу составил (а)  
педагог дополнительного образования  
А.Д. Головин

город Москва  
2016 год

## **Пояснительная записка**

Подготовка школьников к изучению программирования во многих странах мира начинается с различных программных сред, одна из которых – программная среда SCRATCH, разработанная MIT (<http://scratch.mit.edu/>)

Развитие логики и алгоритмического мышления при использовании фундаментальных понятий программирования соответствует плану мероприятий по реализации Концепции математического образования в РФ (Приказ Минобрнауки России № 265 от 3 апреля 2014 г.).

Данный курс посвящён знакомству учащихся 3–4-х классов с основными алгоритмическими конструкциями, развитию логических способностей школьников на базе среды программирования SCRATCH.

Курс разработан на основе авторской программы «Пропедевтика программирования со SCRATCH» Т.Е. Сорокиной в соответствии с: требованиями Федерального государственного образовательного стандарта основного общего образования (ФГОС ООО); требованиями к результатам освоения основной образовательной программы (личностным, метапредметным, предметным); основными подходами к развитию и формированию универсальных учебных действий (УУД) для основного общего образования. В ней соблюдается преемственность с федеральным государственным образовательным стандартом начального общего образования; учитываются возрастные и психологические особенности школьников, обучающихся на ступени основного общего образования, учитываются межпредметные связи.

В программе предложен авторский подход в части структурирования учебного материала, определения последовательности его изучения, путей формирования системы знаний, умений и способов деятельности, развития, воспитания и социализации учащихся.

### **Планируемые результаты изучения**

Планируемые результаты освоения обучающимися основной образовательной программы основного общего образования уточняют и конкретизируют общее понимание личностных, метапредметных и предметных результатов как с позиции организации их достижения в образовательном процессе, так и с позиции оценки достижения этих результатов.

Планируемые результаты сформулированы к каждому разделу учебной программы.

Планируемые результаты, характеризующие систему учебных действий в отношении опорного учебного материала, размещены в рубрике «ученик научится ...». Они показывают, какой уровень освоения опорного учебного материала ожидается от ученика. Эти результаты потенциально достигаемы большинством учащихся и выносятся на итоговую оценку как задания базового уровня (исполнительская компетентность) или задания повышенного уровня (зона ближайшего развития).

Планируемые результаты, характеризующие систему учебных действий в отношении знаний, умений, навыков, расширяющих и углубляющих опорную систему, размещены в рубрике «ученик получит возможность научиться ...». Эти результаты достигаются отдельными мотивированными и способными учащимися; они не отрабатываются со всеми группами учащихся в повседневной практике, но могут включаться в материалы итогового контроля.

*Ученик научится:*

- самостоятельно устанавливать программную среду на домашний компьютер;
- изменять некоторые стандартные установки пользовательского интерфейса (например, язык отображения информации);
- использовать различные способы отладки программ, включая пошаговую отладку;
- уверенно использовать инструменты встроенного графического редактора, включая работу с фрагментами изображения и создание градиентов;
- упрощать программы за счёт использования циклических команд и применять их;

*Ученик получит возможность:*

- создавать программы и игры с использованием интерактивных технологий;
- планировать и создавать анимации по определенному сюжету;

- создавать игры, используя интерактивные возможности программной среды Scratch;
- подходить творчески к построению моделей различных объектов и систем.

## ***Раздел 1. Знакомство с программной средой Scratch***

*Ученик научится:*

- выбирать и запускать программную среду Scratch;
- работать с основными элементами пользовательского интерфейса программной среды;
- изменять размер и перемещать окно программы, выбирать необходимый режим окна;
- вводить имя файла с помощью клавиатуры;
- выбирать необходимый файл из нужной папки;
- создавать, копировать, переименовывать, перемещать, копировать и удалять файлы;
- соблюдать требования техники безопасности при работе в компьютерном классе.

*Ученик получит возможность:*

- определять технические устройства для ввода и вывода информации;
- понимать иерархическую организацию библиотек исполнителей программной среды;
- выделять путь к элементам библиотеки.

## ***Раздел 2. Компьютерная графика***

*Ученик научится:*

- использовать простейшие растровые и векторные редакторы для создания и редактирования изображений;

- изменять центр изображения костюма исполнителя;
- редактировать изображения встроенной библиотеки исполнителей;
- создавать сложные графические объекты путем копирования и модификации простых объектов и их фрагментов;
- использовать возможности работы с цветом.

*Ученик получит возможность:*

- выделять фрагменты изображения для дальнейшей работы с ними;
- планировать работу по созданию сложных изображений путём копирования и масштабирования простых;
- различать верхний и нижний цвета изображения;
- придумывать и создавать различные градиенты для заливки замкнутой области;
- планировать создание симметричных объектов с использованием возможности отражения фрагмента изображения.

### ***Раздел 3. Алгоритмы и исполнители***

*Ученик научится:*

- составлять и отлаживать программный код;
- использовать конструкции программной среды для создания линейных, разветвлённых и циклических алгоритмов;
- организовывать параллельные вычисления;
- моделировать последовательность событий программы, передачу управления от одних исполнителей другим.

*Ученик получит возможность:*

- придумывать задачи для исполнителей программной среды;
- выделять ситуации, для описания которых можно использовать линейный алгоритм, алгоритм с ветвлениями, повторениями;

- определять эффективный способ решения поставленной задачи;
- находить параллельности в выполняемых действиях и программировать их с помощью нескольких исполнителей;
- планировать последовательность событий для заданного проекта.

#### ***Раздел 4. Проектная деятельность и моделирование процессов и систем***

*Ученик научится:*

- использовать возможности программной среды Scratch для создания мультимедийных проектов;
- создавать имитационные модели, интерактивные проекты и игры средствами программной среды.

*Ученик получит возможность:*

- создавать план появления событий для отражения определенной темы;
- выбирать иллюстративный материал из встроенной библиотеки;
- выбирать метод анимации для конкретной задачи;
- планировать последовательность событий для создания эффекта анимации по выбранному сценарию.

# Содержание учебного курса «Юный программист»

## *Раздел 1. Знакомство с программной средой Scratch*

Свободное программное обеспечение. Авторы программной среды Scratch. Параметры для скачивания и установки программной среды на домашний компьютер. Основные элементы пользовательского интерфейса программной среды Scratch. Внешний вид рабочего окна. Блочная структура систематизации информации. Функциональные блоки. Блоки команд, состояний, программ, запуска, действий и исполнителей. Установка русского языка для Scratch.

Создание и сохранение документа. Понятия «спрайт», «сцена», «скрипт». Очистка экрана. Основной персонаж как исполнитель программ. Система команд исполнителя (СКИ). Блочная структура программы. Непосредственное управление исполнителем.

Библиотеки исполнителей. Сцена и разнообразие сцен, исходя из библиотеки сцен. Систематизация данных библиотек исполнителей. Иерархия в организации хранения костюмов персонажа и фонов для сцен. Импорт костюма, импорт фона.

## *Раздел 2. Компьютерная графика*

Компьютерная графика. Векторные и растровые графические редакторы. Встроенный растровый графический редактор. Основные инструменты графического редактора – кисточка, ластик, заливка (цветом или градиентом), рисование линий, прямоугольников, квадратов, эллипсов и окружностей, выбор фрагмента изображения и отражение его по горизонтали или вертикали, использование инструмента «Печать» для копирования выделенной области изображения, работа с текстом. Масштаб фрагмента изображения. Палитра цветов, установка цвета переднего плана и фона, выбор цвета изображения с помощью инструмента «Пипетка». Изменение центра костюма. Изменение размера костюма.

Основные способы изменения внешнего вида исполнителя:

1) использование встроенной библиотеки исполнителей путём импорта её элемента;

2) редактирование выбранного элемента с помощью инструментов встроенного растрового графического редактора;

3) создание собственных изображений в других программах и импортирование их в программную среду Scratch.

### ***Раздел 3. Алгоритмы и исполнители***

Алгоритм. Понятие алгоритма как формального описания последовательности действий исполнителя, приводящих от исходных данных к конечному результату. Схематическая запись алгоритма. Использование геометрических фигур для схематической записи алгоритма.

#### *Линейные алгоритмы*

Основные признаки линейного алгоритма. Схематическое описание линейного алгоритма. Геометрические примитивы, используемые для описания линейного алгоритма.

Программное управление исполнителем. Создание программ для перемещения исполнителя по экранному полю. Понятие «поворот исполнителя в определённое направление». Угол поворота  $90^\circ$ . Поворот исполнителя на  $90^\circ$  по часовой или против часовой стрелки.

Создание программ для рисования линий. Изменение цвета и толщины рисуемой линии. Особенности пунктирной линии. Программа для исполнителя, чтобы он оставлял пунктирную линию при перемещении по экранному полю.

Прямоугольник, квадрат – основные черты. Написание программ для движения исполнителя вдоль сторон квадрата, прямоугольника. Редактирование программы рисования квадрата для получения квадрата другой с другим размером стороны.

Прерывание программы.

#### *Циклические алгоритмы*

Организация циклического алгоритма при необходимости многократного повторения команд. Особенности использования цикла в программе. Оптимизация программы путём сокращения количества команд при переходе от линейных алгоритмов к циклическим.

Схематическая запись циклического алгоритма.



Типы циклических алгоритмов. Основные конструкции программной среды, используемые для написания программ исполнителям с применением циклов.

Конечный цикл. Оптимизация программы для исполнителя, рисующего линии, квадраты, прямоугольники при использовании цикла. Программа исполнителя для рисования нескольких однотипных геометрических фигур, например, нескольких квадратов из одной вершины, но с различным значением стороны.

Конструкции программной среды «спрятаться/показаться». Выполнение программы исполнителем, не показанным в поле выполнения программы.

Написание и отладка программ с применением конструкции «цикл в цикле».

Бесконечный цикл. Повторяющаяся смена внешности исполнителя для имитации движения персонажа. Использование бесконечного цикла для создания анимации.

Получение различных эффектов воспроизведения программы за счёт изменения костюмов исполнителей Scratch.

### *Параллелизм в программной среде*

Использование нескольких исполнителей. Копирование программ исполнителей. Выполнение одинаковых программ разными исполнителями при использовании различных начальных условий. Параллельное выполнение однотипных действий. Принцип суперкомпьютерных технологий. Таймер для вычисления времени выполнения программы.

Интерактивность программ. Возможность организации диалога между исполнителями. Операторы для слияния текстовых выражений.

Взаимодействие исполнителей. Сенсоры касания объектов или цвета. Использование сенсоров при взаимодействии исполнителей. Задержка выполнения программы.

Слои изображения. Выполнение команд исполнителями в разных слоях изображения.

### *Ветвление в алгоритмах*

Использование ветвления при написании программ. Короткая форма. Полная форма условного оператора. Конструкции ветвления для моделирования ситуации.

Цикл «пока». Повторение команд исполнителя при выполнении определенного условия.

Последовательное выполнение фрагментов программы разными исполнителями

Типы исполнителей программной среды Scratch. Системы команд исполнителей. Различные системы команд для разных типов исполнителей.

Управление событиями. Передача сообщений исполнителям для выполнения определенной последовательности команд.

Передача управления между различными типами исполнителей.

#### ***Раздел 4. Проектная деятельность и моделирование процессов и систем***

Мультимедийный проект. Описание сюжетных событий. Анимация. Создание эффекта анимации с помощью последовательной смены изображений. Имитационные модели. Интерактивные проекты. Игры.

## Учебно-тематический план

№	Название темы	Количество часов		
		общее	теория	практика
1	Знакомство с программной средой Scratch	2	1	1
2	Компьютерная графика	5	2	3
3	Алгоритмы и исполнители	14	10	9
4	Проектная деятельность и моделирование процессов и систем	13	6	7
5	Резерв	1	1	
	Итого: для 1 группы	38	19	19