Программирование мини-игры в программе Скретч

**Разделы:** [Информатика](https://urok.1sept.ru/%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)

**Классы:** 5, 6

**Ключевые слова:** [занимательная информатика](https://urok.1sept.ru/%D0%BA%D0%BB%D1%8E%D1%87%D0%B5%D0%B2%D1%8B%D0%B5-%D1%81%D0%BB%D0%BE%D0%B2%D0%B0/%D0%B7%D0%B0%D0%BD%D0%B8%D0%BC%D0%B0%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%B0%D1%8F-%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0), [программирование](https://urok.1sept.ru/%D0%BA%D0%BB%D1%8E%D1%87%D0%B5%D0%B2%D1%8B%D0%B5-%D1%81%D0%BB%D0%BE%D0%B2%D0%B0/%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5), Скретч, [Scratch](https://urok.1sept.ru/%D0%BA%D0%BB%D1%8E%D1%87%D0%B5%D0%B2%D1%8B%D0%B5-%D1%81%D0%BB%D0%BE%D0%B2%D0%B0/scratch)

На свете существуют множество языков программирования, но программирование для детей проще начинать с программ, которые предполагают создание игровой ситуации с героями или объектами, которые двигаются, меняются или взаимодействуют. К таким программам относятся ПервоЛого и ЛогоМиры, а также программа Скретч  (Scratch), бесплатно распространяемая для любой операционной системы. Сайт находится по адресу [http://scratch.mit.edu](http://scratch.mit.edu/). С января 2019 года разработчики выпустили новую версию 2.0. Она немного отличается от предыдущих версий.

В основе Scratch лежит графический язык программирования. Для того чтобы создать скрипт (программу для каждого объекта) нужно просто совместить графические блоки вместе, перетаскивая готовые блоки из левого поля в поле создания скрипта (программы).

Блоки разбиты на несколько типов:

* Движение (синий) – содержит команды перемещения объектов;
* Внешность (фиолетовый) – команды изменения внешнего вида объекта;
* Звук (малиновый) – команды управления звуком;
* Перо (темно-зеленый) – команды рисования на экране;
* События (желтый) – команды управления, контролирующие операторы;
* Управление (оранжевый) - условные операторы и операторы циклов;
* Сенсоры (голубой) – датчики, команды управления мышью, определяющие расстояние и координаты;, числа (ярко-зеленый)  — операции с числами, логические операторы, вычисления, команды сравнения;
* Операторы (ярко-зеленый) – операторы для переменных.

**Блок «Перо»** в новой версии надо добавлять из меню «дополнительные блоки».

На уроках информатики учащиеся постепенно знакомятся с командами различных блоков и их использованием. Это позволяет на одном из уроков перейти к реализации проекта по созданию мини-игры.

Во многих компьютерных играх игровое поле представляет собой площадку, разбитую на квадраты (как шахматная доска), и требуется собрать какие-либо предметы – это могут быть яблоки, монеты, звезды или что-то другое. Для краткости будем называть эту игру «Собери» ([*презентация*](https://urok.1sept.ru/articles/673699/pril1.ppt)).

Для создания игры нам необходимо проделать несколько этапов:

* Создание фона – игрового поля,
* Расстановка различных предметов (объектов) на игровом поле;
* Программирование главного героя;
* Программирование собираемых объектов;
* Программирование препятствий;
* Программирование финиша (финишной кнопки).

**1 этап. Создание игрового поля**

Для создания игрового поля мы должны войти в меню «Сцена», вкладка «Фоны» и с помощью инструментов для рисования ***«прямоугольник»*** и ***«линия»*** создать поле 8х8 клеток. (рис. 1). Такого размера вполне достаточно для создания первой игры. При желании можно добавить какой-либо текст (кнопка Т).

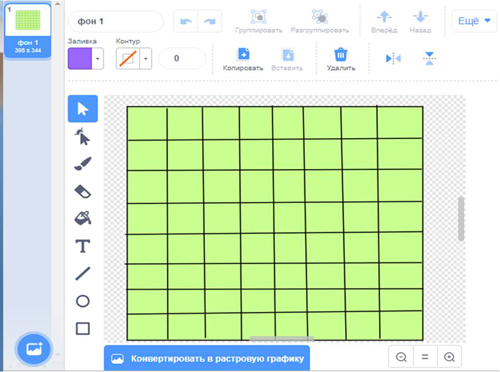


Рис. 1. Создание игрового поля.

**2 этап. Расстановка игровых объектов**

На втором этапе мы расставляем объекты разных видов (рис. 2 и 3). Для этого используем добавление спрайтов из библиотеки. Объекты в нашей игре будут двух видов – те, которые по условию надо собрать (например, яблоки), и те, которые будут являться препятствиями. Количество объектов можно задавать самостоятельно. Рекомендую размер объектов уменьшить до 50.

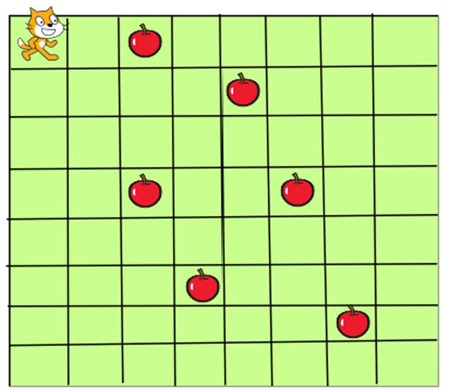


Рис. 2. Добавление объектов на игровое поле.

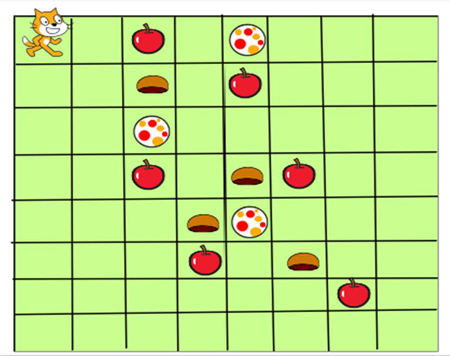


Рис. 3. Добавление объектов-препятствий на игровое поле.

**3 этап. Программирование «героя»**

В качестве героя можно оставить исходного спрайта-котенка под именем «Спрайт1». При желании можно выбрать (назначить) «героем» любого другого. Имя спрайта также можно изменить.

В начале игры наш герой должен встать в исходную точку, которую мы определяем с помощью координат. Поэтому команда должна выглядеть так:

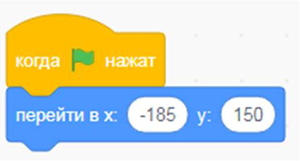


Рис. 4. Задание начальной координаты для спрайта.

При необходимости можно задать и начальное направление – например, «повернуться в направлении 90».

Далее для того, чтобы наш спрайт ходил сразу по клеткам, мы должны запрограммировать кнопки управления его передвижением (стрелки вверх, влево, вправо, вниз), которые удобно расположены на клавиатуре.

Команды для кнопок должны выглядеть следующим образом:

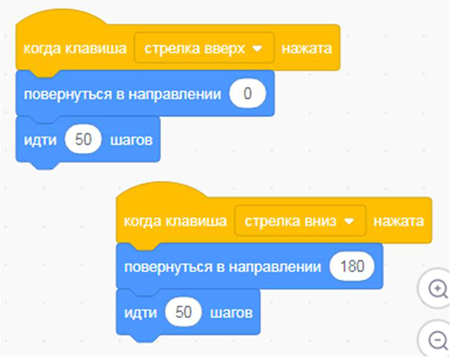


Рис. 5. Программирование управляющих кнопок (стрéлок).

Таких команд должно быть четыре. Направление задается следующим образом: вверх – **0**; вправо – **90**; вниз – **180**; влево –  **-90**. Длина шага определяется размером клеток на игровом поле. При размере поля 8х8 клеток шаг получается – примерно 50. Далее можно проверить, как ходит герой и подредактировать длину шага.

**Этап 4. Программирование собираемых объектов**

Для того, чтобы любой объект реагировал на прохождение «героя» и мог менять свою форму или исчезать, необходимо, чтобы он совершал небольшое движение. Это можно задать двумя способами:

1) небольшое движение «фишки» (например, «идти 1 шаг», «ждать 1 секунду», «идти -1 шаг» либо

2) «изменить размер на 5%», «ждать 1 секунду», «изменить размер на -5%».

Реакция на прохождение героя задается через ветвление «Если – то» с условием «касается Спрайт1» - команда «Спрятаться». Все команды запускаются по команде «Когда флажок нажат» и через цикл «Повторять всегда», так как мы не знаем, в какой момент игры наш герой коснется именно этой фишки.

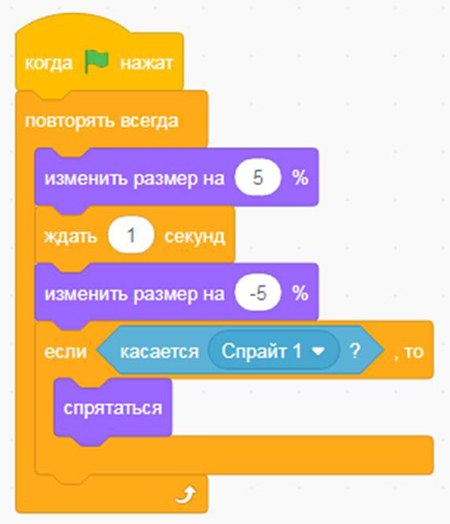


Рис. 6. Команда для собираемых объектов.

Чтобы объекты можно было «восстановить» в начале игры задаем дополнительную команду:

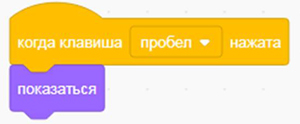


Рис.7. Команда для «восстановления» объектов.

Эти команды надо задать для каждого собираемого объекта (яблока, монеты или звезды).

**Этап 5. Программирование препятствий**

На игровом поле должны также присутствовать объекты, которые будут усложнять игру, т.е. являться препятствиями. Чтобы они могли взаимодействовать с героем, когда он попадает на соответствующую клетку, необходимо им также задать небольшое движение (аналогично собираемым объектам).

Взаимодействие с героем можно задавать по-разному: препятствие может «взрываться», «переворачиваться» (т.е. изменить свой костюм), выдавать сообщение «Ты проиграл» или «Здесь ходить нельзя» и т.д. Можно задать команду, которая будет перемещать героя в начало игры, т.е. в исходную точку. Варианты могут придумать сами дети.  В начале игры также надо вернуть этим объектам исходный костюм.

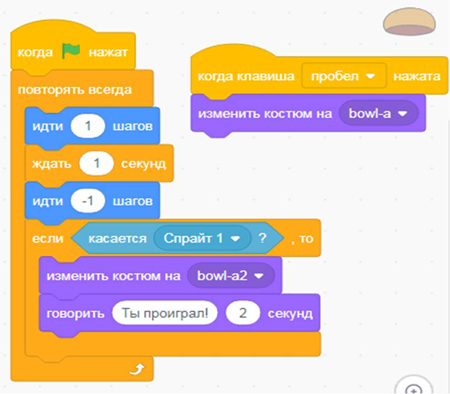




Рис. 8, 9. Программирование препятствий.

**Этап 6. Программирование кнопки финиш**

По желанию можно создать кнопку «Финиш» и поместить ее в конце игрового поля. Кнопка «Финиш» программируется аналогично предыдущим объектам. Отличие будет в тех командах, которые будут выполняться при реакции на прохождение Спрайта. Например, можно задать «Говорить «Ты победил!» или «Переходи на второй уровень» и при этом кнопка может менять цвет или мигать и т.д. Здесь дети могут использовать свою фантазию и желание.



Рис. 10. Изменение кнопки «Финиш».



Рис. 11. Программа для кнопки «Финиш».

**Заключение**

Подобную игру можно создать за 1 урок или одно занятие. Практика показывает, что подобные задания дети выполняют с удовольствием и могут проявить фантазию или использовать более сложный набор команд для создания подобной игровой ситуации. При этом ученики не замечают, как знакомятся с программированием и таким важным моментом, как **отладка программы**, которая требует усидчивости, терпения и умения видеть и исправлять ошибки в своей программе.

Scratch действительно относится к группе мультимедийных развивающих сред программирования.

Отзывы и замечания можно присылать на e-mail annasp70@mail.ru.

**СПИСОК ЛИТЕРАТУРЫ**

1. Голиков Д., Голиков А. Книга юных программистов на Scratch. Изд. Smashwords, 2013, 140 с.
2. Шапошникова С. Введение в Scratch, 2011, 41 с. [http://younglinux.info](http://younglinux.info/)
3. [https://scratch.mit.edu](https://scratch.mit.edu/)