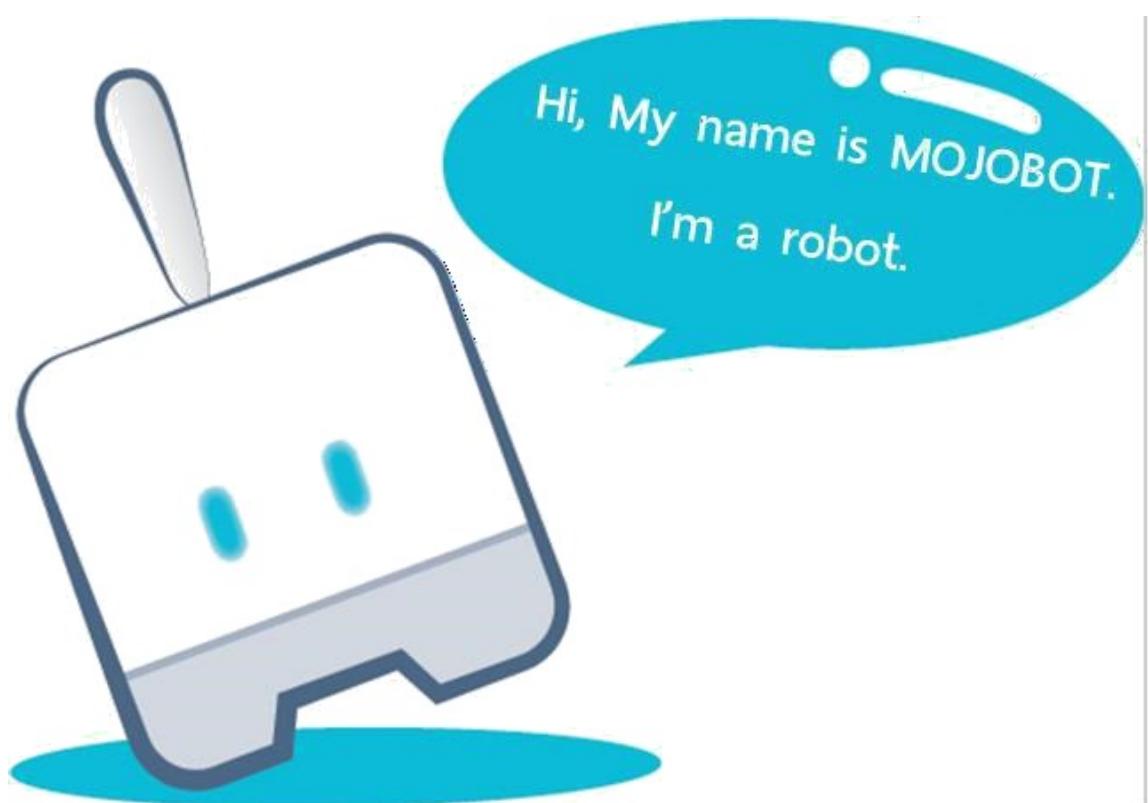




# Mojobot Level 2



# Содержание

Введение	3
Задание 1 - Добавь на консоль: расширий свой код	4
Задание 2 - Повтори и умножь	6
Задание 3 - Повторяй и создавай модели	9
Задание 4 - Больше повторений	6
Задание 5 - Застенчивый робот	13
Задание 6 - Танцующий робот	16
Задание 7 - Робот-доставщик	19
Задание 8 - Интеллектуальная лампа	22
Задание 9 - Моджобот ищет жетоны	25
Задание 10 – Робот-бык	27
Задание 11 - Моджобот уклоняется от столкновения	29
Задание 12 – Робот-уборщик	31
Задание 13 – Repeat Until («Повторять до тех пор, пока») Часть 1	33
Задание 14 – Repeat Until («Повторять до тех пор, пока») Часть 2	35
Задание 15 - Подпрограмма	36
Задание 16 – Робот-администратор	38
Задание 17 – Робот-спасатель	39
Задание 18 – Финальная миссия	41

## Введение

В Mojobot 2-го уровня основное внимание уделяется формированию понимания того, как создать программу, которая позволит Моджоботу принимать автономные решения, используя входные данные датчиков. Данные уроки подходят для любого пользователя старше 6 лет.

Обучающиеся узнают, как использовать команды управления потоком, такие как «циклы» и «if statement» в сочетании с «условными входными данными», для управления роботом и задания ему желаемого автономного поведения. В заданиях используются входные данные датчиков в виде простых логических данных, в которых все значения являются либо истинными, либо ложными. Мастерство управления потоком лежит в основе разработки алгоритмов в информатике. Другими словами, цель состоит в том, чтобы познакомить учеников с более продвинутыми и фундаментальными по сравнению с Моджоботом 1-го уровня концепциями вычислительного мышления.

По завершении приведенных здесь уроков обучающиеся освоят основную концепцию вычислительного мышления и смогут разрабатывать алгоритмы для решения конкретных задач. Помимо того, что ученики смогут создавать автономные алгоритмы, они также улучшат навыки решения задач, аналитические и экспериментальные навыки, а также навыки творческого мышления. Уроки сочетают в себе естественные науки и математику, а также лингвистику, искусство и дизайн, так как обучающимся предлагается модифицировать своих роботов и карты, чтобы рассказать свою собственную историю. В учебной программе уделяется особое внимание STEAM как дополнению к STEM.

В целом, по сравнению с уроком 1-го уровня уроки 2-го уровня направлены на улучшение алгоритмического, систематического и абстрактного мышления обучающихся. Эти навыки помогут подготовиться к работе с более сложными языками программирования, такими как Scratch или Python.

# Задание 1 – Добавь на консоль: расширь свой код

## Цель задания

В этом упражнении обучающимся предлагается дополнительная консоль для написания более длинных программ, которые позволят им улучшить навыки систематического мышления.

## Материалы



## Этапы обучения

1. Учитель зачитывает инструкцию к каждому примеру.
2. Обучающиеся стараются выполнить задачу, используя всего одну программу.

## Пример 1

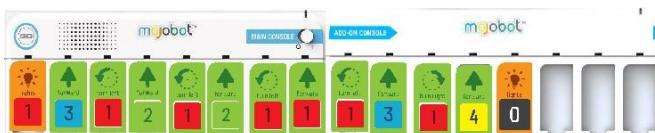
Моджобот грустит, поэтому он выходит из Дома (Home), затем идет на 2 блока на север, один раз поворачивает направо и далее идет вперед на 3 блока. Затем Моджобот ждет 2 секунды, и его лампочка загорается зеленым светом. Теперь Моджобот счастлив. Он возвращается домой тем же маршрутом.



## Пример 2

Моджобот должен доставить пиццу по 4 адресам. Каждый раз, когда пицца доставлена, его лампочка загорается красным, а когда доставлены все заказы — выключается.

Маршрут доставки: Офисные здания – Дом – Автомастерская – Спорткомплекс



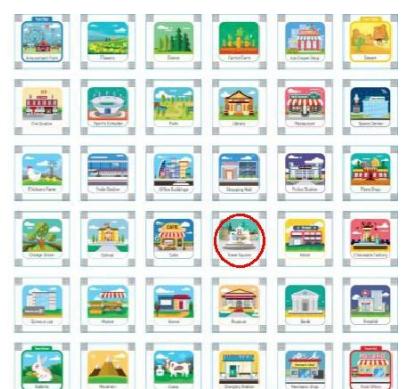
### Пример 3

Моджобот берет Жетон ископаемых в Пустыне (Desert) и кладет его в Музей (Museum), затем забирает Жетон научного образца в Научной лаборатории и доставляет его в Торговый центр (Shopping Mall) для научной выставки.



## Пример 4

Моджобот находится на Городской Площади (Town Square), где неожиданно начинает танцевать. После танца его лампочка загорается зеленым, красным, желтым, а затем синим. Каждый цвет горит 2 секунды. Затем он выключает лампочку и снова танцует. После этого он становится счастливым.



## Задание 2 – Повтори и умножь

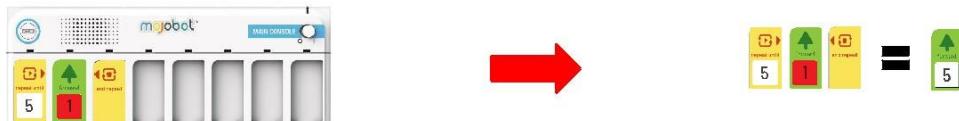
### Цель задания

Это задание направлено на то, чтобы познакомить обучающихся с концепцией повторения и цикла в программировании. Они узнают, что повторение с помощью циклического преобразования аналогично умножению в математике.

### Материалы



### Пример 1



Помести Моджобота на блок «Больница» (Hospital), по направлению на запад.

Моджобот идет вперед на 1 блок, повторяет 5 раз.

Результат:  $1+1+1+1+1 = 5$ ,

«1 плюс 1» 5 раз или  $5 \times 1 = 5$  блоков

**Пример 2** Помести Моджобота на блок «Больница» (Hospital) по направлению на запад.



Моджобот идет вперед на 5 блоков, повторяет 2 раза.  
 Результат:  $5+5=10$  блоков,  
 «5 плюс 5» 2 раза считается как  $2\times5=10$  блоков.

**Пример 3** Помести Моджобота на блок «Ресторан» (Restaurant) по направлению на север.



Моджобот идет налево на 1 блок, повторяет 5 раз.  
 $1+1+1+1+1=5$  блоков  
 «1 плюс 1» 5 раз считается как  $1\times5=5$  поворотов.

## Правильное использование повторений.

Пояснение: ученики заполняют пробелы цифрами, чтобы получить верный ответ

1.



=



+



+



+



Моджобот идет вперед на \_\_\_\_\_ блока(ов), считается как \_\_\_\_\_ X \_\_\_\_\_

2.



=



+



+



Моджобот идет вперед на \_\_\_\_\_ блока(ов), считается как \_\_\_\_\_ X \_\_\_\_\_

3.



=



+

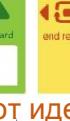


+



Моджобот идет вперед на 3 блока, считается как \_\_\_\_\_ X \_\_\_\_\_

4.



=



+



+



+



Моджобот идет вперед на 12 блоков, считается как \_\_\_\_\_ X \_\_\_\_\_

5.



=



+



+



+



Моджобот идет вперед на \_\_\_\_\_ блока(ов), считается как \_\_\_\_\_ X \_\_\_\_\_

6.



=



+



Моджобот идет налево на 4 блока, считается как \_\_\_\_\_ X \_\_\_\_\_

Преподаватель может использовать это упражнение в качестве ориентира и усложнять задание при необходимости.\*\*

## Задание 3 – Повторяй и создавай модели

### Цель задания

Это задание направлено на то, чтобы продемонстрировать обучающимся, что повторение и зацикливание можно использовать для формирования циклических шаблонов, таких как движение вперед и назад или движение вокруг своей оси.

### Материалы

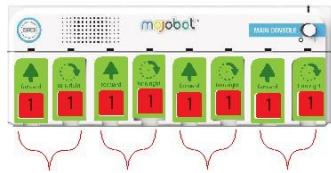


### Этапы обучения

1. Преподаватель приводит пример того, как можно использовать «Repeat» (повтор). Сначала покажите, как задать модель движения без использования «Repeat», а затем сделайте то же самое, используя «Repeat». Пусть ученики посмотрят и обсудят преимущества команды «Repeat» и метода «зацикливания».
2. Пусть ученики потренируются на примерах.

Примечание: Не забывайте ставить робота по направлению к северу в начале каждого круга.

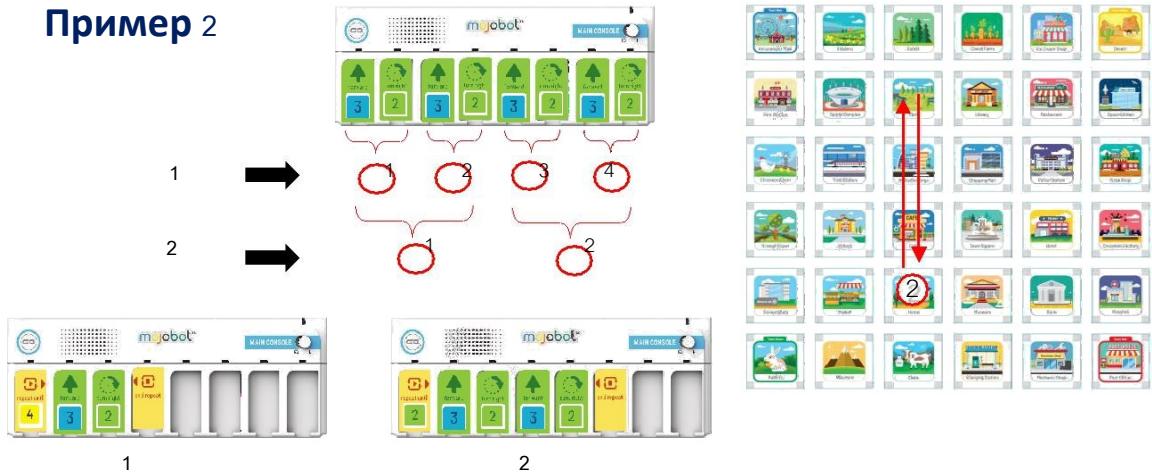
#### Пример 1



1 2 3 4



## Пример 2



## Задание: используй Repeat (повтор)

Учащиеся используют тег «Repeat» для создания следующих моделей движения.



Diagram illustrating a mojobot program example. It shows a sequence of icons for a Repeat loop:

1. Zoo, Library, Restaurant, Shopping Mall, Police Station, Desert
2. Zoo, Library, Restaurant, Shopping Mall, Police Station, Desert
3. Zoo, Library, Restaurant, Shopping Mall, Police Station, Desert
4. Zoo, Library, Restaurant, Shopping Mall, Police Station, Desert

Red arrows indicate the flow from the first icon in each row to the last icon in the row, forming a loop.

## Задание 4 – Больше повторений

### Цель задания

Это задание помогает обучающимся тренироваться использовать повторяющиеся и циклические команды, чтобы они могли развивать навыки решения задач с помощью циклов.

### Материалы



### Этапы обучения

- Подготовьте учеников к работе в группе или индивидуально, с использованием “бумажной консоли”
- Выполните задания, используя тег «Repeat».

### Пример задания



Моджобот должен забрать посылку с Вокзала (Train Station) и доставить ее на рынок, а затем вернуться на Вокзал и подождать 2 секунды.

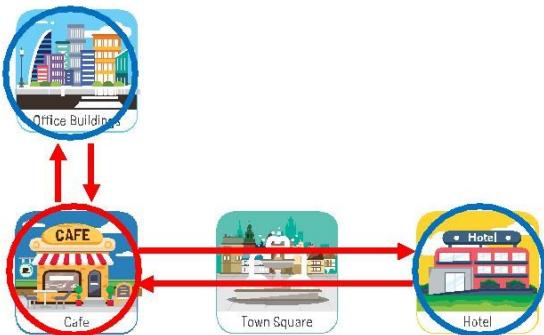
Затем Моджобот должен забрать новую посылку и доставить ее к Цветам (Flowers), после чего вернуться к Вокзалу и подождать 2 секунды. Моджобот должен повторить эти действия 3 раза.

Примечание: начинайте по направлению на север



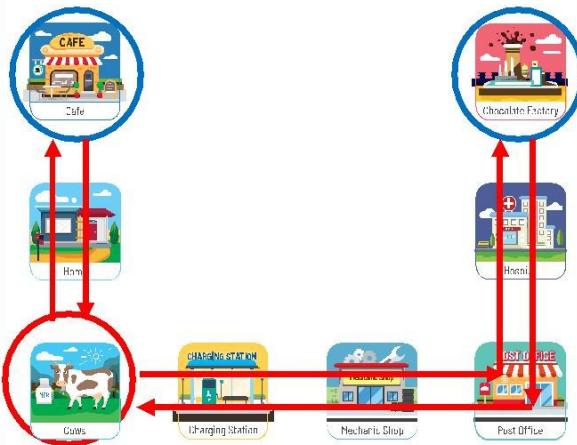
## Задание: Используй Repeat (повтор)!

1.



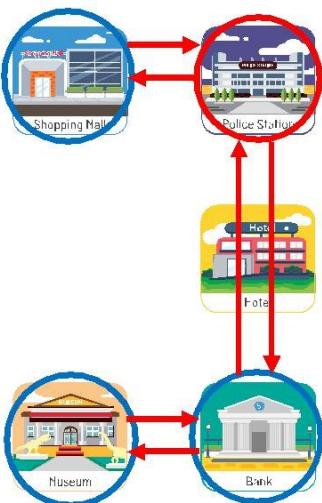
Моджобот работает в Кафе (Cafe) и должен доставить заказы покупателям в Отель (Hotel). Выполнив доставку, Моджобот возвращается в Кафе. Затем он должен доставить заказы в Офисные здания (Office Building) и снова вернуться в Кафе. Работы много, поэтому Моджобот работает весь день!  
(Подсказка: используй “repeat infinity” (повторять бесконечно))

2.



Моджобот открывает молочную ферму в блоке «Коровы» (Cows). Он доставляет молоко в Кафе, затем возвращается к Коровам, чтобы взять еще молока и доставить его на Шоколадную Фабрику (Chocolate Factory), после чего снова возвращается к Коровам. Моджобот делает это 5 раз в день.

3.



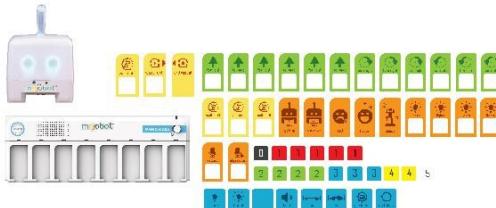
Офицер Моджобот должен выйти из Полицейского Участка (Police Station), чтобы проверить площадь рядом с Торговым Центром (Shopping Mall) до того, как вернуться в Полицейский Участок и продолжить патрулировать местность рядом с Банком (Bank) и Музеем (Museum). После того, как он закончит, ему нужно вернуться на отдых. Все эти действия нужно выполнять 3 раза в день.

## Задание 5 – Застенчивый робот

### Цель задания

С помощью этого задания вводятся две новые команды: команда “wait until” («ждать до») и “near” («близко») для входных данных датчика. Обучающиеся узнают, как использовать данные датчика для создания автономного робота, который очень стесняется!

### Материалы

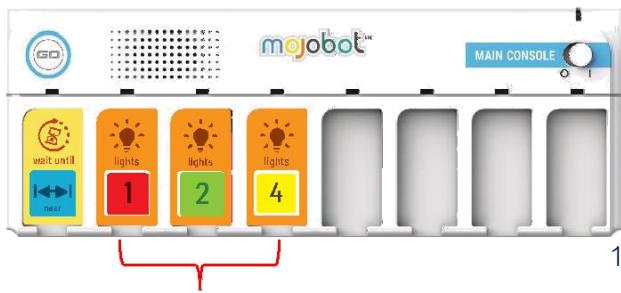


### Этапы обучения

1. Преподаватель объясняет задание: создать застенчивого робота.
2. Пусть ученики украсят робота так, чтобы он выглядел застенчивым.
3. Преподаватель показывает на примере, как использовать команду “wait until” в комбинации с командой “near” датчика.

### Пример задания

#### Задание 1 Учимся использовать Wait until с Near



##### 1.1 Расположи Моджобота в

любом месте на карте

##### 1.2 Нажми кнопку «Go» (старт)

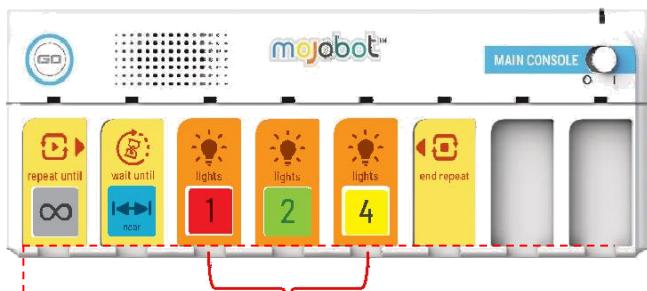
##### 1.3 Проверь лампочку на основной консоли

##### 1.4 Поднеси руку к лицу Моджобота

Можно использовать любые теги из категорий “Movement” (движение) и “Action” (действие).

## Пример задания

**Задание 2** Используй зацикливание Wait until и Near, чтобы Моджобот работал автономно.



1.1 Расположи Моджобота в любом месте на карте

1.2 Нажми кнопку «Go» (старт)

1.3 Проверь лампочку на основной консоли

1.4 Поднеси руку к лицу Робота и посмотри на результат

1.5 Еще раз поднеси руку к лицу Робота

Можно использовать любые теги из категорий “Movement” (движение) и “Action” (действие).

### Пояснение

Второй пример показывает, что при выполнении бесконечного цикла каждый раз, когда вы приближаетесь к Роботу, его лампочка мигает.

## Задание

Обучающиеся должны написать программу для застенчивого Робота. Поощряйте творческий подход при выполнении этого задания. Можно изменить внешний вид робота, используя бумагу, добавить звук и движения, используя действия и запись звука. Каждый застенчивый робот будет обладать собственными забавными чертами.

Программа работает с «Wait to near» внутри цикла, который будет продолжаться внутри бесконечного цикла. Каждый раз при начале нового цикла, при использовании команды “Wait until near” Робот будет находиться в состоянии ожидания до тех пор, пока датчик не распознает какой-нибудь предмет поблизости. Моджобот стесняется и убегает, когда кто-то подходит!



## Датчик расстояния

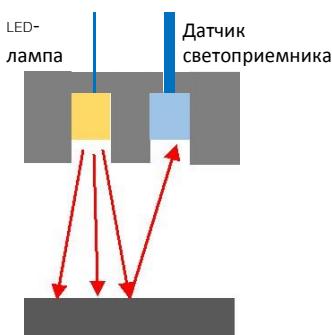
Датчик расстояния используется для измерения расстояния между двумя объектами. Существуют различные типы датчиков расстояния. Моджобот использует «Инфракрасный свет», который излучает светодиод (светоизлучающий диод). Когда инфракрасный свет попадает на объект, он отражается обратно в фотодиод или фоторезистор, который преобразует интенсивность света в электрические сигналы. При высокой яркости света создается электрический сигнал с высоким напряжением, который отправляется на компьютер, управляющий Роботом.

Датчик предусматривает возможность создания программ для проверки значений датчиков. Таким образом можно распознать, далеко или близко находится объект. Другими словами, можно «спросить» у датчика, характеризуется ли измеренное расстояние как «близко» или «далеко». Расположив руку перед лицом робота, вы услышите звук. Так Моджобот реагирует на значение “Near” (близко), полученное с датчика. Затем попробуйте убрать руку, и вы снова услышите звук, который будет являться реакцией на значение “Far” (далеко).

### Датчик расстояния:

- “Near” («Близко») – датчик показывает это значение в пределах 1 блока, примерно на расстоянии 3-14 см.
- “Far” («Далеко») – датчик показывает это значение на расстоянии более блока, дистанция составляет более 15 см.

LED-лампа Световой сигнал датчика



Чем ближе объект, тем больше отражается свет.



# Задание 6 – Танцующий робот

## Цель задания

В этом задании мы создадим автономного танцующего робота. Это можно сделать используя команду “wait until” («ждать пока») и данные датчика звука.

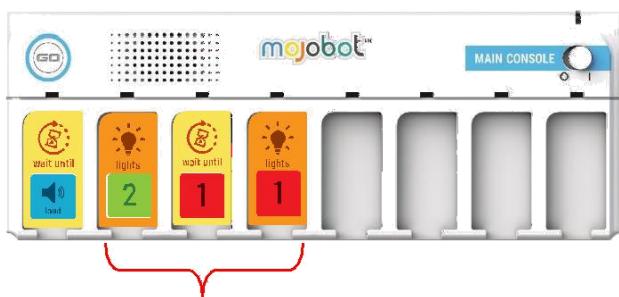
## Материал



## Пример задания

### Задание 1

Учимся использовать `Wait until` с тегом датчика звука `>>loud<<`



1.1 Расположи Моджобот в любом месте на карте

1.2 Нажми кнопку Go (старт)

1.3 Посмотри на лампочку на основной консоли

1.4 Хлопни в ладоши рядом с Роботом

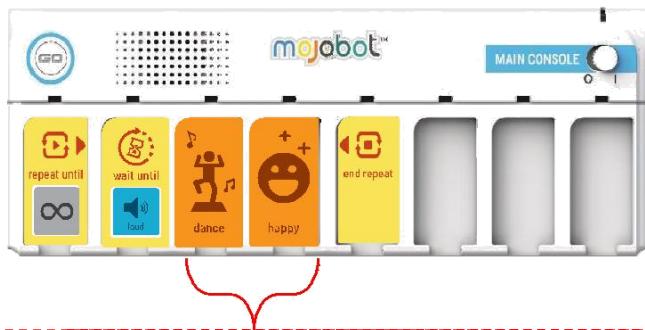
Можно использовать любые теги из категорий “Movement” (движение) и “Action” (действие).

## Задание

Обучающиеся будут писать программу для Танцующего Робота, дайте им время найти решение самостоятельно. Если у них возникли затруднения, преподаватель может дать им правильный ответ.

Постарайтесь вдохновить учеников использовать свой творческий потенциал при выполнении этого задания. Начните с того, чтобы предложить им трансформировать внешний вид Робота. Для этого ученики могут использовать бумагу, цветные карандаши, прозрачную ленту, трубочки и др. Кроме того, они могут создавать движения, звуки и действия, чтобы их Танцующий Робот был красочным, веселым и уникальным.

Программа DanceBot содержит команду “Wait until loud” внутри цикла, который будет воспроизводиться непрерывно. При использовании команды “wait until loud” Робот будет ждать, пока датчик звука не зафиксирует громкий звук, затем перейдет к выполнению следующей команды. После танца цикл снова вернется к команде “wait until loud”, так что Моджобот продолжит танцевать, если громкая музыка все еще играет.



1.1 Расположи Моджобота в любом месте на карте

1.2 Нажми кнопку «Go» (старт)

1.3 Проверь лампочку на основной консоли

1.4 Хлопни в ладоши рядом с Роботом

1.5 Хлопни в ладоши рядом с Роботом еще раз

Можно использовать любые теги из категорий “Movement” (движение) и “Action” (действие).

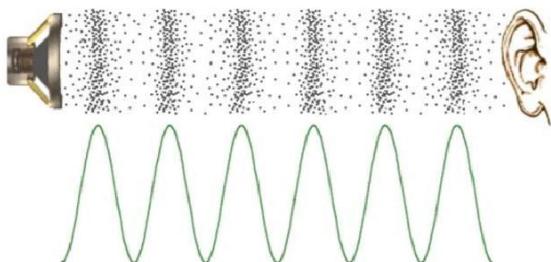
### Пояснение

Вы заметите, что если в комнате тихо, Моджобот не будет двигаться, а код остановится на команде “Wait”. Когда раздастся громкий звук, робот начнет танцевать.

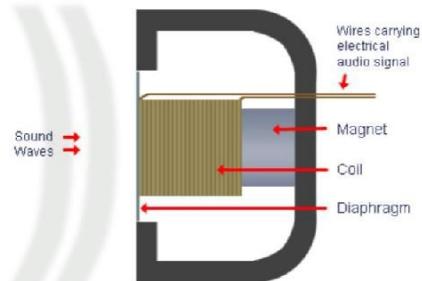
## Работа датчика звука

Звуковой датчик представляет собой датчик, который преобразует звуковые вибрации в электрический сигнал. Фактически, звуковой датчик имеет тот же принцип работы, что и микрофон. Принцип работы диафрагмы аналогичен вибрирующей резинке. Звук производится за счет вибрации объектов, а затем передается при столкновении молекул в воздухе, пока не достигнет человеческого уха или датчика.

Вибрации в воздухе вызывают вибрацию диафрагмы, которая заставляет катушку двигаться внутрь и наружу, пока магнит неподвижен. Это событие называется Электромагнетизмом, движение провода (катушки) через стационарное магнитное поле производит электрический ток, протекающий в катушке. Наконец, электрический сигнал в катушке поступает на компьютер для обработки. Чем сильнее электрический сигнал, тем громче звук.



Звуковые волны - это колебания молекул, будь то твердые молекулы или жидкое, такие как воздух.



Принцип работы датчика звука



Расположение микрофона

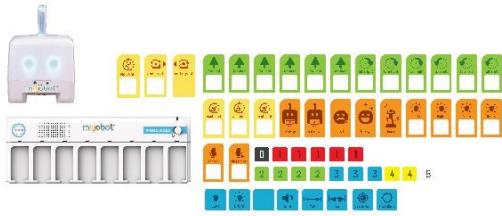
# Задание 7 – Робот-доставщик

## Цель задания

В этом задании обучающиеся смогут применить свое знание команды “wait until” для создания автономного Робота-доставщика жетонов.

Ученики научатся программировать датчик жетонов при помощи тегов “Yes Token” («Есть жетон») и “No Token” («Нет жетона»).

## Материалы



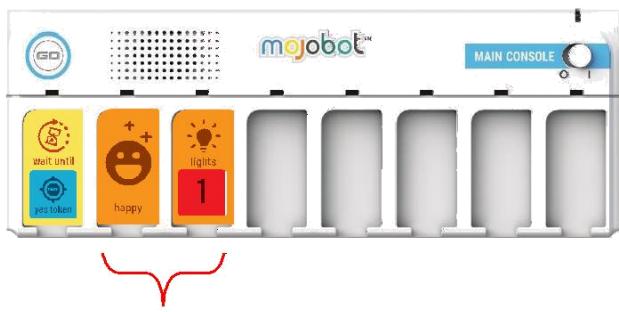
## Этапы обучения

1. Преподаватель объясняет задачу Робота-доставщика.
2. Преподаватель показывает на примере, как использовать команды “wait until” и “yes token” для датчика.
3. Ученики самостоятельно пишут программу для Робота-доставщика.

## Пример задания

### Задание 1

Учимся использовать **Wait until** с **>>loud Yes Token<<**



1.1 Расположи Моджобота в любом месте на карте

1.2 Нажми кнопку «Go» (старт)

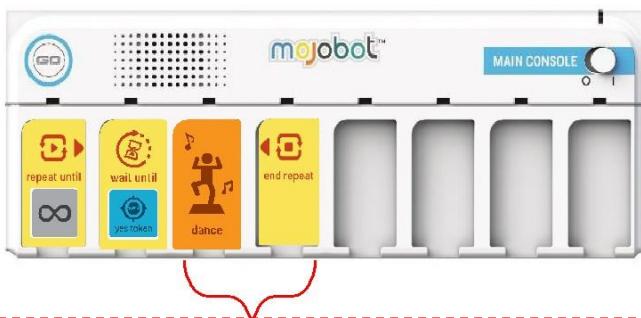
1.3 Проверь лампочку на основной консоли

1.4 Попробуй положить жетон под Робота

Можно использовать любые теги из категорий “Movement” (движение) и “Action” (действие).

## Пример задания

**Задание 2** Учимся использовать **Wait until** с >> **Yes Token** << (Автономный робот)



1.1 Расположи Моджобота в любом месте на карте

1.2 Нажми кнопку «Go» (старт)

1.3 Проверь лампочку на основной консоли

1.4 Попробуй положить жетон под Робота

Можно использовать любые теги из категорий “Movement” (движение) и “Action” (действие).

### Пояснение

Из второго примера видно, что благодаря “repeat infinity” (бесконечному повторению) цикла Моджобот будет начинать танцевать каждый раз, когда обнаружит внизу жетон.

## Задание

### Робот-доставщик

### Моджобот стоит на Вокзале (Train Station) лицом к северу.

Сегодня у Моджобота самый счастливый день в году, потому что его добрый дедушка приезжает в гости. Моджобот и его семья едут на Вокзал, чтобы его встретить. По прибытии Моджобот замечает, что дедушка привез с собой много подарков и вызывается помочь отнести их Домой (Home). Загвоздка в том, что Моджобот может взять только один подарок за раз. Он ждет, пока дедушка передаст подарок, затем несет его Домой, после чего возвращается на Вокзал и ждет, пока дедушка даст следующий подарок. (Примечание. Всего 7 подарков, которые нужно отнести домой: Сюрприз, Яйца, Апельсины, Книги, Молоко, Почта, Морковь)

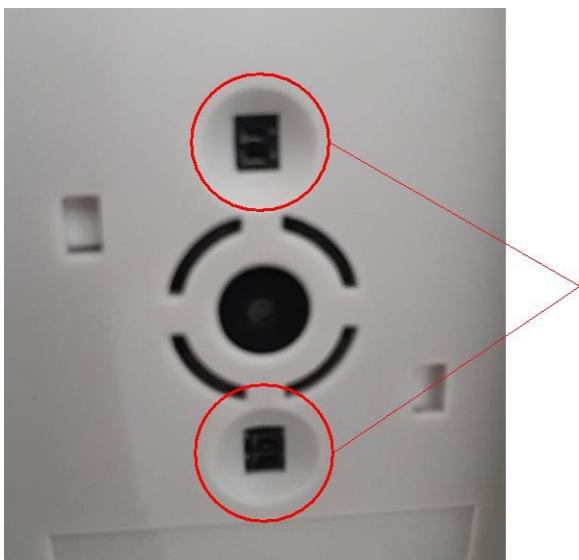
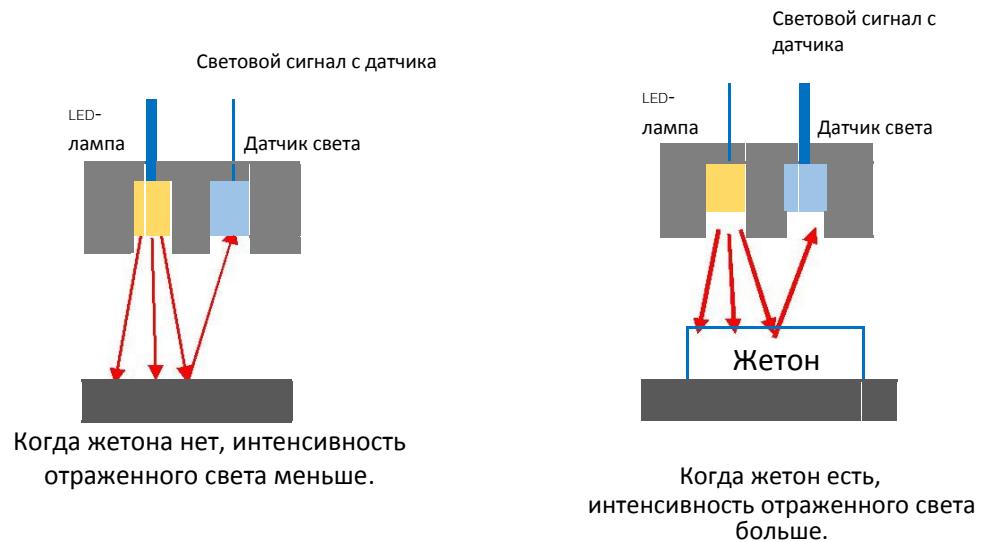


В задании ученикам предлагается написать такую программу, чтобы Моджобот мог перенести несколько жетонов, по одному за раз. Каждый раз, когда цикл заканчивается, Моджобот должен возвращаться на Вокзал и ждать, пока ему не передадут следующий предмет. Чтобы передать предмет, нужно медленно положить соответствующий жетон под Робота.

## Работа датчика жетонов

Моджобот распознает наличие жетона, используя датчик расстояния, расположенный снизу. Датчик расстояния использует принцип инфракрасного излучения и отражения света. Распознаваемое расстояние - более 1 см и менее 1 см. Если внизу есть жетон, датчик определит более короткое расстояние и создаст меньшее напряжение.

Положив жетон под Робота, вы услышите сигнал, оповещающий о том, что он его обнаружил. Чтобы чувствительный датчик работал корректно, не используйте его под воздействием прямых солнечных лучей, так как мощное инфракрасное излучение спровоцирует ошибочное обнаружение датчиком жетона.



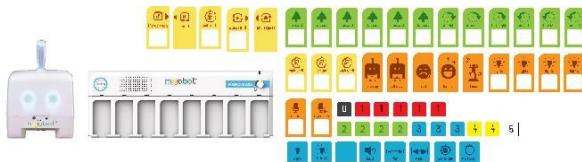
Под роботом находятся 2 пары инфракрасных датчиков. Чтобы обнаружить жетон, необходимо активировать обе пары.

# Задание 8 – Интеллектуальная лампа

## Цель задания

Цель этого задания состоит в том, чтобы познакомить обучающихся с “if statement” (оператором «если») и датчиком света. Эти знания ученики смогут применить для создания интеллектуальной лампы. Интеллектуальная лампа – это лампа, которая автоматически включается в темное время суток и так же автоматически выключается, когда светло, для экономии энергии.

## Материалы



## Этапы обучения

1. Преподаватель объясняет задачу - создать интеллектуальную лампу.
2. Ученики украшают Робота при помощи цветной бумаги, карандашей, лент и т.д.
3. Преподаватель показывает, как использовать “if statement”, используя теги с входными данными датчика: “bright” (светло) и “dark” (темно).
4. Ученики самостоятельно пишут программу согласно заданию.
5. Ученики показывают свою работу одноклассникам.

## Пример задания

### Задание 1 Учимся использовать >>if true then<< с >>Dark<<



1.1 Расположи Моджобота в любом месте на карте

1.2 Используй картон, чтобы накрыть датчик света сверху и нажми кнопку «Go» (старт)

1.3 Убери картон и снова нажми кнопку «Go» (старт)

Вы увидите, что если темно, программа будет выполнять эти команды

Если светло, программа пропустит и выполнит только

## Пример задания

### Пример 2 Учимся использовать >>if true then<< с >>Bright<<



- 1.1 Расположи Моджобота в любом месте на карте
- 1.2 нажми кнопку «Go» (старт) (поставь Робота на открытое пространство)

Снова нажми кнопку «Go»

Вы увидите, что если темно, программа будет выполнять

Если светло, программа пропустит,



И выполнит только

### Пример 3 учимся использовать >>if true then<< с >>Bright<< и >>Dark<< (автономный Робот)



Вы увидите, как Моджобот принимает решение

- 1.1 Выполни показанный код.
- 1.2 Используй руку или другой предмет, чтобы блокировать свет, поступающий на датчик наверху Робота

- 1.3 Убери руку или другой предмет, чтобы свет попал на датчик.

Если светло, Моджобот включит зеленую лампочку.

Если темно, Моджобот включит синюю лампочку.



## Работа датчика света

Измерение света производится при помощи электрических компонентов и схем, которые преобразуют интенсивность света в электрический сигнал, используя такие компоненты, как фотодиод, фоторезистор или фототранзистор.

Обычно транзистор, диод или резистор являются компонентами общей электрической цепи. Но фотодиод представляет собой особый тип диода, электрические свойства которого зависят от интенсивности или яркости света. Обычно, чем ярче свет, тем выше напряжение электрического сигнала. Датчик света будет работать в частотном спектре видимого света, в отличие от инфракрасных датчиков, работающих в инфракрасном спектре.



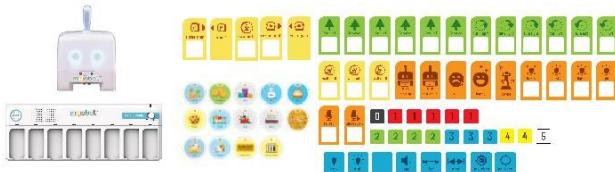
Если посмотреть внимательно, можно увидеть черный силуэт кольца - это датчик света, прикрепленный внутри, под верхней крышкой Робота.

## Задание 9 – Моджобот ищет жетоны

## Цель задания

Это задание позволяет понять принцип работы “if statement” (оператора «если») с помощью выполнения исследовательских заданий. В этом упражнении на примерах показано, как применяется “if statement” в качестве ответвления в путях принятия решений.

## Материалы



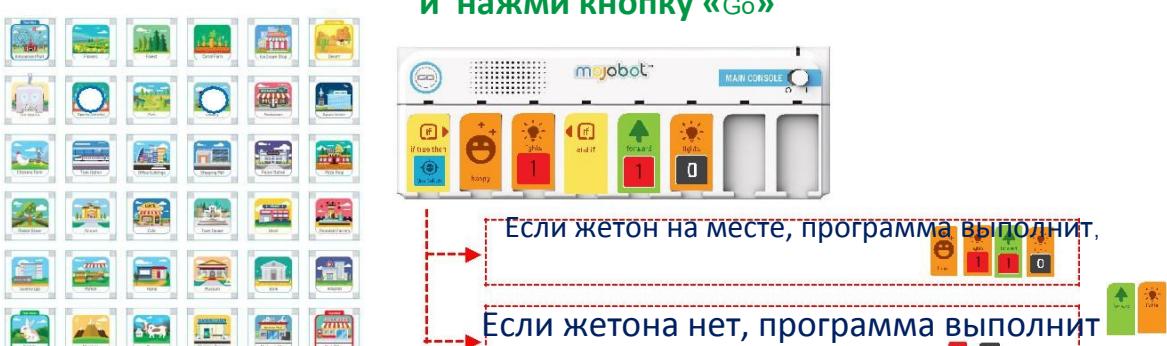
## Этапы обучения

1. Преподаватель показывает теги программирования “if statement” на боковых входах датчиков жетонов.
  2. Обучающиеся пробуют выполнить задания, используя “if statement” и датчики жетонов.
  - 3.

## Пример задания

**Пример 1** Размести Робота в Пожарной Части (Fire Station) лицом к Спорткомплексу (Sports Complex). Расположи жетоны так, как показано на рисунке.

**Расположи теги так, как показано на рисунке  
и нажми кнопку «Go»**



В этом задании преподаватель может нажать кнопку «Go» 3-4 раза посмотреть на результат в случаях, когда жетон есть и когда его нет.

## Пример задания

### Пример 2 Расположи Робота Дома (Home) лицом к северу карты.

Расположи теги так, как показано на рисунке и нажми кнопку «Go»



Если жетон есть, Моджобот направится в Библиотеку (library).

Если жетона нет, Моджобот направится в Магазин Пиццы (pizza shop).

\*\*\* Попробуйте оба сценария: с жетоном и без него. Пусть ученики посмотрят и обсудят результаты.

## Задача: выстроить понимание

1. Школьный день окончен и Моджобот едет из Школы (School) Домой. Если Дома есть Пицца, он берет ее, чтобы съесть, а затем весело танцует, но если Пиццы нет, у него все равно отличное настроение, потому что сегодня не нужно делать домашнее задание.

2. Моджобот находится на Рынке (Market) и проверяет, есть ли жетон «Пикник» (Picnic). Если он есть, то он поднимет его и идет на Рынок, а если нет, то отправляется в Ресторан (Restaurant).

## Задача: выстроить понимание с использованием двух дополнений

3. Мама просит Моджобота сходить на Почту (Post Office) и проверить, нет ли для них писем. Он выходит из Дома и направляется на Почту. Если есть Письмо (Mail), то он приносит его Домой и кладет, а потом говорит маме: «вот письмо для тебя». Если писем нет, то он возвращается Домой и говорит маме: «Писем нет, мама» (“there is no Mail mom”). (Примечание: используйте программу записи голоса, нужно записать два отдельных трека до запуска программы)

## Задание 10 – Робот-бык

### Цель задания

В этом задании обучающимся нужно использовать “if statement” для создания Робота-быка. Это Робот, который атакует любой предмет, оказавшийся рядом.

### Материалы



### Этапы обучения

1. Преподаватель объясняет задачу Робота-быка. Робот-бык распознает предмет перед собой, а затем бежит, чтобы толкнуть его.
2. Пусть ученики украсят Робота так, чтобы он был похож на быка.
3. Покажите, как использовать `>>if true then<< c >>Near<<` и `>>Far<<`
4. Пусть ученики самостоятельно напишут программу для Робота.
5. Пусть ученики покажут свои работы одноклассникам.

### Пример задания

**Пояснение** Покажите обучающимся, как использовать команды `>> If True Then << c >>Near<<`.

**Пример 1** Используйте команду `>>if true then<< c >>Near<<`

Разместите теги так, как показано на рисунке и нажмите «Go»



Если перед Роботом есть объект, он начнет танцевать.

Если объекта нет, Робот не будет танцевать.

\*\*\* Покажите ученикам оба сценария: когда перед Роботом есть предмет и когда нет.

## Пример

**Пояснение** Покажите обучающимся, как использовать команды >> If True Then << с датчиком входа >> Far <<

### Задание 2 Используйте команду >>if true then<< с >>Near<<

Разместите теги так, как показано на рисунке и нажмите «Go»



Если предмет далеко, Моджобот будет танцевать и в конце будет «счастлив».

Если не далеко, Моджобот не будет танцевать.

Покажите ученикам оба сценария: когда рядом с Роботом есть предмет и когда нет.

## Задание

**Пояснение** Робот-бык будет постоянно поворачиваться, пока не увидит какой-либо объект. Если Робот увидит предмет или человека, то немедленно атакует его.

### Задание 3 Программа для Робота-быка

Разместите теги так, как показано на рисунке и нажмите «Go»



Если объект находится близко, Робот атакует.

Если предмет перед Роботом находится далеко, то он будет поворачиваться направо.

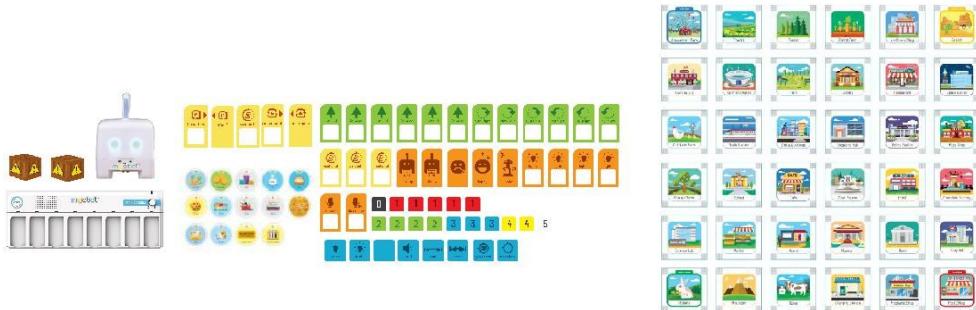
\*\*\* Пусть ученики изменят внешний вид и поведение Робота, чтобы сделать его уникальным при помощи звуков и света.

# Задание 11 – Моджобот уклоняется от столкновения

## Цель задания

Цель этого задания – улучшение навыков программирования с помощью создания программы, которая научит Робота уклоняться от предметов. Задача состоит в том, чтобы Моджобот двигался вперед до тех пор, пока не обнаружит препятствие или тупик. Если Моджобот попадает в тупик, он должен повернуться и двигаться дальше в другом направлении. Чтобы выполнить задание, обучающиеся должны использовать “if statement” и датчик расстояния.

## Материалы



## С чего начать...

1. Преподаватель объясняет задание: сделать так, чтобы Робот двигался вперед, пока не обнаружит препятствие, а затем разворачивался и двигался в другом направлении. Если он снова обнаружит препятствие, то он должен повернуться и снова повторить. В результате Моджобот будет двигаться между двумя препятствиями.
2. Преподаватель показывает на примере, как использовать “if statement” с “near” и “far”.
3. Ученики выполняют задание.

## Пример задания

### Задание 1 Размести Робота в Парке лицом к Офисным Зданиям и поставь препятствия так, как показано на рисунке.

Разместите теги так, как показано на рисунке и нажмите «Go»



Если впереди нет препятствий, Моджобот продолжит двигаться вперед.

Если есть препятствие, Моджобот сделает разворот.

\*\*\* Нажмите кнопку «Go» несколько раз, чтобы проработать оба варианта: когда препятствие есть и когда его нет.

## Задание

Моджобот будет двигаться вперед до тех пор, пока не обнаружит препятствие или тупик. Затем он повернется и начнет двигаться в противоположном направлении. Когда Робот обнаружит препятствие или тупик, он снова повернется и повторит действие в обратном направлении.

Разместите теги так, как показано на рисунке и нажмите «Go»



Чтобы убедиться, что перед ним нет препятствий, Моджобот будет двигаться вперед.

Если перед ним есть препятствие, Моджобот развернется, чтобы избежать столкновения.

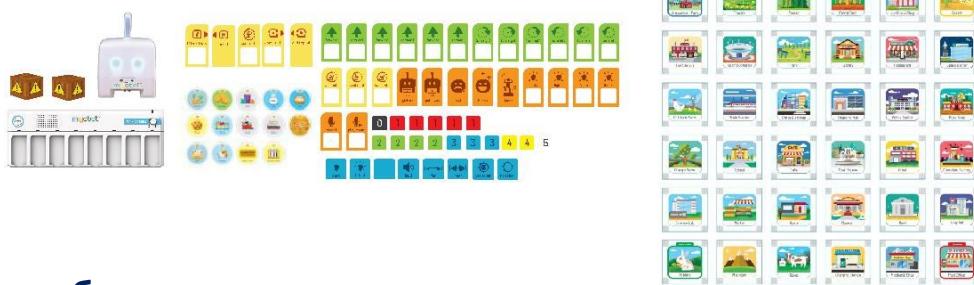
\*\*\* Пусть ученики сделают Робота уникальным, используя свет, звук и эмоции. Например, если препятствий нет, то загорается зеленая лампочка, а если впереди тупик - красная.

## Задание 12 – Робот-уборщик

## Цель задания

Это задание направлено на улучшение и углубление понимания и способности учащихся применять команды “if statement” в сочетании с входными данными датчиков для получения желаемого автономного поведения. Задача состоит в том, чтобы Моджобот автоматически очистил выбранный участок города от препятствий.

## Материалы

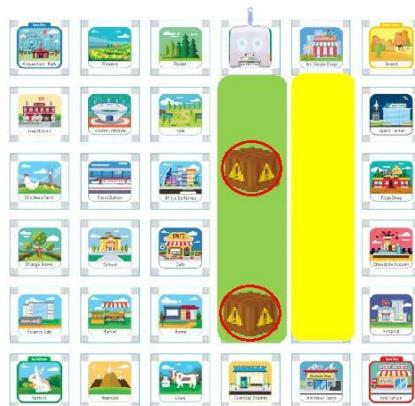


## Этапы обучения

1. Объясните задание ученикам.
  2. Пусть ученики попробуют выполнить задание. Они могут работать в группах или индивидуально, используя «бумажную консоль».

## Задание

Задача Моджобота – переместить все препятствия из зеленой зоны в желтую. Препятствия расположены в случайном порядке и могут оказаться где угодно в зеленой зоне. Ученики не знают заранее, где расположены препятствия, так что Робот должен быть готов к любой ситуации.



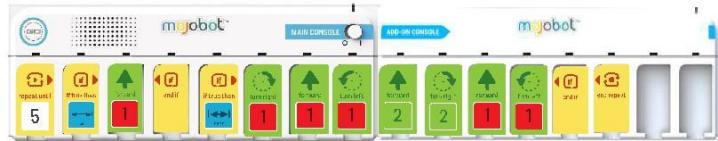
\*\*\* Обучающиеся могут использовать различные движения, чтобы расчистить путь Робота.

## Решение

Чтобы расчистить путь, Моджобот подходит к препятствию сбоку, а затем двигается вперед, чтобы передвинуть его в желтую зону.



## Пример расположения тегов для решения задачи



Задание считается выполненным, когда Моджобот переместил все препятствия из зеленой зоны в желтую.

## Дополнительное задание

Пусть ученики работают вместе, чтобы убрать все предметы с восточной стороны карты. Моджобот передвигает предметы с желтого поля на синее, а затем с синего поля – за пределы карты.



\*\*\* Продумайте порядок совершения действий, необходимых для выполнения этого задания. Сначала работа должна быть выполнена на зеленом поле, затем на желтом, и наконец, на синем.

## Задание 13 – Repeat Until («Повторять до тех пор, пока») Часть 1

### Цель задания

Это задание направлено на то, чтобы познакомить обучающихся с условными циклами. Эта цель может быть достигнута при помощи использования команды “Repeat Until” («Повторять до тех пор, пока») в сочетании с входными условиями датчика.

### Материалы



### Этапы обучения

- Преподаватель показывает и объясняет принцип работы условных циклов и команды “Repeat Until” в сочетании с входными данными датчика.
- Ученики могут выполнять задание в группах или индивидуально.

### Пример задания

**Попробуйте использовать код, показанный на рисунке. Обратите внимание, что происходит, когда вы закрываете датчик освещенности**

(Примечание: Не выполняйте это задание под воздействием прямых солнечных лучей, так как необходимо закрыть датчики освещенности, чтобы Робот распознал темноту. На солнце это практически невозможно.)



Запустите программу. Пока светло, Робот будет выполнять цикл и постоянно двигаться вперед. Если вы закроете датчики освещения, программа выйдет из цикла и завершит работу, так что Робот остановится.



Во втором примере мы увидим то же поведение, что и в первом, за исключением того, что на этот раз, когда вы закроете датчик освещенности, чтобы код вышел из цикла, Моджобот будет отображать «грустные» эмоции до конца программы.

#### Пояснение

Каждый раз, когда программа выполняет команду «repeat until», она проверяет данные датчика. Если условие истинно, например, в этом случае темно, то программа выйдет из цикла, если нет, программа продолжит выполнение цикла «repeat».

## Задание

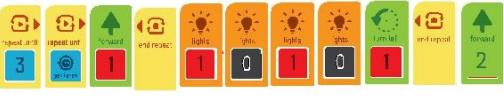
Для выполнения этого задания ученики должны использовать условные циклы. Им нужно использовать тег “Repeat Until” в сочетании с тегами входных данных датчика.

Задание	Шаблон тегов
<p>1. Моджобот находится Дома лицом к северу и начинает двигаться вперед до тех пор, пока не обнаружит препятствие. Затем он произносит несколько раз: «обнаружен тупик, не могу продолжать».</p> 	 <p>Код будет повторяться в первом цикле. За один цикл Робот будет двигаться вперед на 1 блок, пока не будет обнаружено препятствие. Затем код выйдет из первого цикла и перейдет во второй цикл, в котором повторно воспроизведет записанный звук.</p>
<p>2. Моджобот находится на Городской Площади лицом на север. Он вращается вокруг себя, и при обнаружении препятствия его лампочка начинает мигать разными цветами. Если убрать препятствие, Моджобот снова начнет вращаться.</p> 	 <p>Это два вложенных цикла, один из которых находится внутри другого. Внешний цикл - это бесконечный цикл, в котором Моджобот будет поворачиваться влево один раз в каждом круге цикла. Если Моджобот распознает, что объект далеко, он выйдет из внутреннего цикла. Поэтому код будет продолжать работать во внутреннем цикле, только если Моджобот обнаружит объект рядом.</p>

## Задание 14 – Repeat Until («Повторять до тех пор, пока») Часть 2

### Цель задания

Это задание направлено на то, чтобы ученики потренировались в использовании условных циклов и научились применять их для решения задач.

Задание	Шаблон тегов
<p>3. Моджобот находится на Городской Площади (Town Square) лицом на север. Он начинает танцевать каждый раз, когда слышит музыку. Если музыки нет, Роботу становится грустно.</p> 	 <p>В этом примере кода есть два вложенных цикла. Внешний цикл заставляет Моджобота «танцевать» один раз в каждом цикле. Но если звука нет, программа будет выполнять внутренний цикл и робот будет постоянно «грустить», пока не услышит звук, после которого программа выйдет из внутреннего цикла.</p>
<p>4. Моджобот находится на Рынке (Market) лицом на восток и отправляется на поиски жетонов. Когда жетон найден, робот поднимает его, танцует, а затем несколько раз мигает лампочкой. Расположите жетоны в случайном порядке так, чтобы ученики заранее не знали, где они находятся.</p> 	 <p>Здесь есть 2 цикла: 1-й цикл и 2-ой. В первом цикле Моджобот движется вперед до тех пор, пока не обнаружит жетон. После этого код выходит из первого цикла Моджобот поднимает жетон и танцует. Затем начинается 2-ой цикл, и Моджобот мигает лампочкой несколько раз.</p>
<p>5. Моджобот находится на Рынке (Market) лицом на восток. Покажите ученикам код на рисунке внизу. Ученики должны расположить три жетона так, чтобы при выполнении этого кода Моджобот смог вернуться Домой (Home).</p> 	 <p>Расположите три жетона так, как показано на картинке. Моджобот будет поворачивать налево каждый раз при обнаружении жетона.</p>

# Задание 15 - Подпрограмма

## Цель задания

Цель этого задания - познакомить обучающихся с подпрограммами и функциями. Ученики узнают, что они могут создать свою собственную подпрограмму (также называемую функцией), которая может помочь организовать и упростить программирование. Использование подпрограммы предполагает поиск шаблонов и повторений в коде, которые можно определить как подпрограмму, чтобы можно было вызывать набор действий с помощью всего 1 команды.

## Материалы

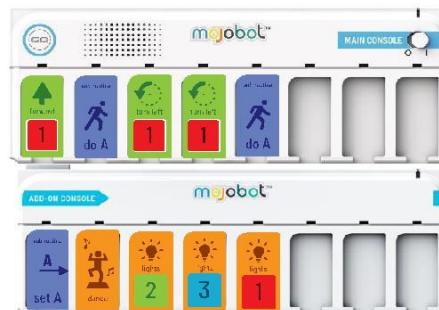


## Этапы обучения

1. Преподаватель показывает и объясняет принцип работы подпрограмм.
2. Ученики выполняют задание в группе или индивидуально.

## Пример

Подключите  
дополнительную консоль



Используйте Do A и  
Set A

## Результат

1. Моджобот делает 1 шаг вперед
2. Моджобот танцует и мигает зеленым, синим и красным цветами.
3. Моджобот поворачивает налево один раз.
4. Моджобот поворачивает налево один раз.
5. Моджобот танцует и мигает зеленым, синим и красными цветами..

## Задача

Используя код, показанный ниже, обучающиеся должны описать последовательность действий, которые будет выполнять Моджобот. Это задание направлено на то, чтобы ученики могли потренироваться читать коды с помощью подпрограмм.

	Основная консоль	Дополнительная консоль
1.		
2.		
3.		
4.		
5.		

# Задание 16 – Робот-администратор

## Цель задания

Это задание, в котором обучающиеся будут создавать робот-администратора, используя знания подпрограмм. Предложите им найти в коде повторяющийся шаблон, который можно выделить в подпрограмму.

## Материалы



## Этапы обучения

1. Преподаватель объясняет задачу.
2. Предложите ученикам украсить Моджобота с помощью бумаги, цветных карандашей, ленточек и т.д.
3. Пусть ученики запрограммируют Моджобота.
4. Пусть ученики покажут свою работу одноклассникам.



## Задание

**Пояснение** Моджобот находится в отеле и помогает бабушке принимать гостей. Его обязанность – приветствовать всех гостей, говоря, например: «Здравствуйте, добро пожаловать в наш отель, надеюсь, что вы будете чувствовать себя комфортно, хорошего отдыха». Моджобот приветствует каждого гостя, который оказывается рядом с ним. Также он произносит приветствие, если к нему обращаются или он получает жетон или его глядят по голове (датчик освещения).



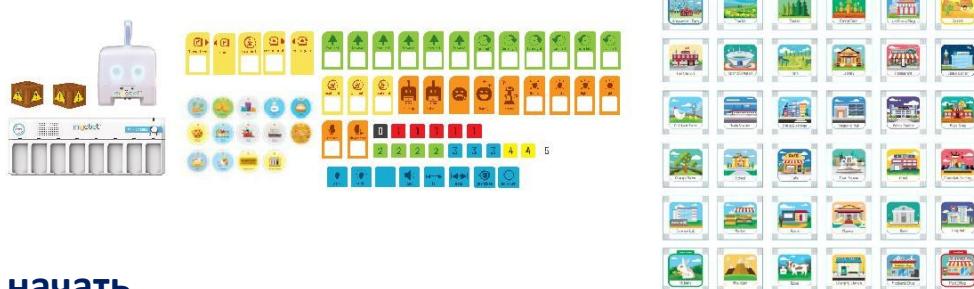
Пусть ученики выяснят, какую часть кода нужно выделить в подпрограмму. Обычно это код, который часто повторяется. В данном случае это действия приветствия, которые объединены в подпрограмму A.

## Задание 17 – Робот-спасатель

### Цель задания

Цель этого задания – сделать так, чтобы ученики применили свои знания подпрограмм для решения задач. По сюжету на окраине восточной части города происходит стихийное бедствие – пожар. Моджобот должен помочь при эвакуации, переместив предметы из восточной части города в западную.

### Материалы



### Как начать...

1. Преподаватель объясняет задание
2. Пусть ученики украсят робота так, чтобы он стал похож на пожарного.
3. Пусть ученики начнут создавать программу.
4. Пусть ученики покажут результат своей работы одноклассникам.

### Задание

**Пояснение** К сожалению, иногда происходят такие стихийные бедствия, как пожары и землетрясения. Моджобот должен помочь при эвакуации, переместив предметы из восточной части города в западную.



Жетоны могут появляться в случайном порядке в любом месте восточной части города. Задача Моджобота заключается в том, чтобы перенести все предметы из восточной части города в западную за один запуск программы!

Пусть ученики решат, какая повторяющаяся часть кода может быть объединена в подпрограмму. Здесь мы выделили в подпрограмму поднятие и доставку одного жетона с востока на запад.



В этом примере Моджобот будет двигаться вперед 5 шагов в цикле. После каждого движения вперед он будет проверять, есть ли внизу жетон. Если он найдет жетон, то поднимет его и доставит в западную часть города, а затем вернется на то же место, где его обнаружил. Процесс повторяется 5 раз, так что все жетоны должны быть перемещены из восточной части города в западную.

## Задание 18 – Финальная миссия

### Цель задания

Это задание является основным для проверки способности обучающихся применять навыки программирования для решения задач. Ученики могут использовать все свои знания и любые типы тегов для решения задачи. Можно использовать неограниченное количество тегов и слотов. Главное – выполнить задание и заработать максимум очков. Бонусное очко получит команда, которая использует наименьшее количество слотов.

### Материалы

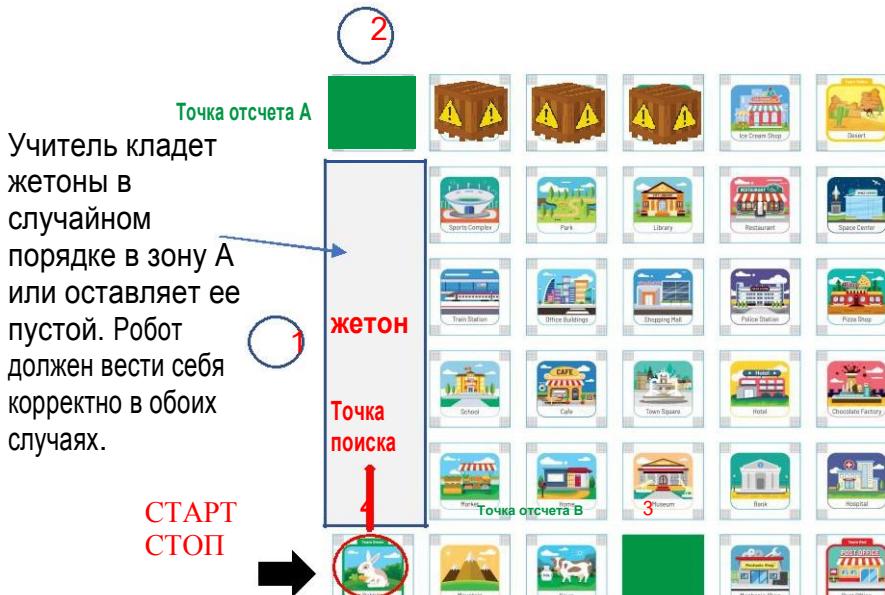


### Этапы обучения

1. Преподаватель делит учеников на группы по 4-5 человек и меняет карту на лунную карту.
2. Пусть ученики разработают план расположения тегов для выполнения этого задания.

### Задание

**Пояснение** Пусть ученики составят такой код, чтобы Моджобот выполнил задание как показано на карте.



**Пояснение** Моджобот стартует с блока Rabbits (Кролики) по направлению на север. За выполнение всех задач максимально можно получить 8 баллов.

1. Моджобот танцует на том месте, где он обнаружил жетон в зоне 1. Если в зоне 1 нет жетона, он не танцует. – 1 балл
2. Моджобот отчитывается в точке А. Если в зоне 1 был обнаружен жетон, лампочка загорается зеленым, если нет - красным. – 1 балл
3. Убрать все три препятствия с карты. – 1 балл за каждое (максимально 3 балла)
4. Перейти в зону 3 (точка отсчета В). – 1 балл
5. Подождать приказа офицера. Офицер отдает приказ, похлопав Моджобота по голове. – 1 балл
6. Вернуться в точку старта – 1 балл

## Ответ

Здесь предложено два варианта решения данного задания. Имейте в виду, что существует множество других путей.

### Решение задачи без использования подпрограммы



### Или с использованием подпрограммы



\*\* Есть множество путей решения данного задания, поэтому ученики могут разработать свои собственные варианты. Пусть они покажут их одноклассникам, а затем обсудят преимущества и недостатки каждого способа.