



РОБОТОТЕХНИКА ДЛЯ МАЛЫШЕЙ

РАБОЧАЯ ТЕТРАДЬ 2

УЧЕБНО - ИНЖЕНЕРНЫЙ
КОМПЛЕКС

ОТ КОНСТРУИРОВАНИЯ ДО РОБОТОТЕХНИКИ



#1276

274 ДЕТАЛИ

4+

ИЗОБРЕТАТЕЛЬСТВУ
МОЖНО НАУЧИТЬСЯ

20 ЗАНЯТИЙ





МЫ РАДЫ ПРЕДСТАВИТЬ ВАМ УЧЕБНО - ИНЖЕНЕРНЫЙ КОМПЛЕКС Gigo. ОТ КОНСТРУИРОВАНИЯ ДО РОБОТОТЕХНИКИ.

УЧЕБНО - ИНЖЕНЕРНЫЙ КОМПЛЕКС – это средства обучения и методический комплекс, основанные на союзе естественных наук, технологий, инженерии и математики, известном под аббревиатурой STEM, общим объемом 520 учебных часов, для детей от 3-х до 16-ти лет, основанный на модульном принципе, что позволяет использовать комплекс как полностью, так и его отдельные модули. Учебно - инженерный комплекс создан по принципу от простого к сложному, сочетая учебную программу на основе естественно-научного и инженерно-технических профилей с заданиями из повседневной жизни.

УЧЕБНО - ИНЖЕНЕРНЫЙ КОМПЛЕКС состоит из секций: Креативная Лаборатория, Научные Эксперименты, Зеленая энергия, Робототехника и Новые Технологии и Соревнования - ГринМех. Каждая из секций подразделяется на модули/курсы. Модуль/курс включает в себя: программу курса, учебное пособие для преподавателя, рабочие карточки для учеников, наборы деталей для сборки, интерактивные схемы сборки Gigo.

Данный модуль называется "Робототехника для малышей" (#1276), входит в секцию "Робототехника и Новые Технологии" и является первым робототехническим набором для малышей дошкольников. Учебное пособие к данному модулю включает в себя 20 занятий для детей младшего дошкольного возраста и 20 занятий для старших дошкольников, набор предназначен для детей от четырех лет. В учебном пособии для педагога подробно описан материал каждого занятия, принципы кодирования и приводятся интересные и исторические факты из робототехники. На занятиях дети конструируют роботов и составляют программный код из карточек, играют и наглядно видят, как робот исполняет команды, заданные ими. Данный модуль предусматривает возможность работы детей над собственными творческими проектами, создание роботов и игровых персонажей не по схеме, а по собственному воображению ребенка.

Изобретательству можно научить – это концепция УЧЕБНО - ИНЖЕНЕРНЫЙ КОМПЛЕКСА Gigo. Мы объясняем ОЧЕНЬ сложные физические и химические процессы ОЧЕНЬ простым языком. Мы ответим на САМЫЕ сложные детские вопросы: «Почему горит лампочка? Откуда берется электричество? Как работают солнечные батареи?»
КАК МЫ ЭТО ДЕЛАЕМ?

Мы делаем опыты, конструируем, исследуем, обсуждаем и конечно же соревнуемся. Мы не боимся ставить эксперименты и получать ответы на свои вопросы.

ЗАЧЕМ МЫ ЭТО ДЕЛАЕМ?

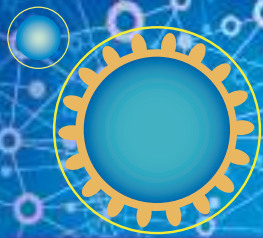
Мы хотим развить интерес к науке, к экспериментам, к творчеству, что бы дети не боялись делать ошибки и всегда стремились идти дальше к поставленной цели. Мы уверены, что дети могут с энтузиазмом получать научные знания с помощью увлекательного практического опыта, развивающего их способности решать задачи, а также интерес к науке.

Наша миссия состоит в том, чтобы помочь детям применить полученные новые знания в повседневной жизни, способствуя развитию инновационных навыков, новых компетенций и способностей.

Содержание

Цели и задачи	1	9. Поднять мяч	49
Содержание	2	10. Конкурсная работа (2)	51
Описание комплекта	3	11. Спаси кошку от пожара на кухне	53
Основные действия	7	12. Спаси кошку и вернись на станцию	59
Список деталей	13	13. Спаси кошку и птицу	63
1. Прятки! Помогите Семёну найти Платона	15	14. Пожарно - спасательная полоса препятствий	67
2. Дайте время, чтобы найти в какой комнате спрятался пудинг Паша?	21	15. Конкурсная работа (3)	69
3. Спросите Машу – молочницу, как приготовить яблочный пирог!	25	16. Найди кубики того же цвета	71
4. Делаем яблочный пирог!	29	17. Найди кубики равного значения	73
5. Конкурсная работа (1)	33	18. Найди кубики в последовательности	75
6. Прямой выстрел	35	19. Найти кубики, добавляя до переменной	77
7. Игра Ромы – решение дня	41	20. Конкурсная работа (4)	79
8. Поле условий	45	Графический код	81



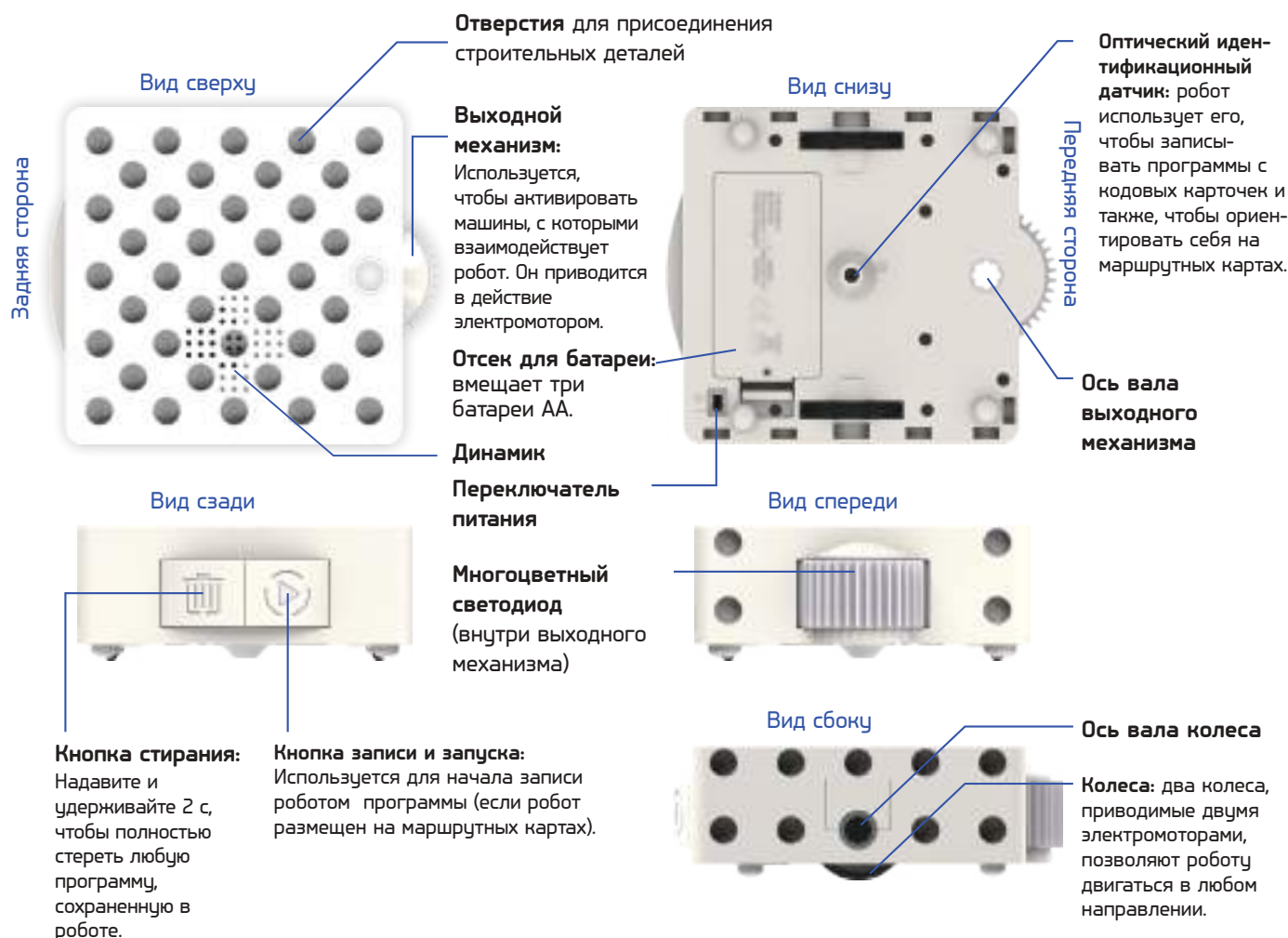


Описание комплекта

Добро пожаловать в увлекательное обучение программирования робота! Во-первых, давайте рассмотрим основные части этого комплекта: роботизированный базовый блок, кодовые карточки, рамки для кодовых карточек и маршрутные карты.

Роботизированный базовый блок

Это база для всех роботов, которых вы можете собрать с помощью этого комплекта. Пошаговые инструкции размещены в этой книжке, начиная со стр.16. Они показывают, как собирать строительные детали на базовый блок робота или для других моделей, которые можно сконструировать вместе с роботами. Роботизированный базовый блок имеет высокую функциональность! Вот все его функции:



Режимы работы робота

Робот имеет два режима работы.

По умолчанию робот находится в обычном режиме. В уроках 1–15 используйте только обычный режим. Робот должен быть установлен в математический режим, чтобы делать уроки математики в конце этой книги. Узнайте, как использовать математический режим, начиная со стр. 71.

Работа в обычном режиме

В обычном режиме, когда вы переводите выключатель питания робота во включенное положение, робот готовится к записи. После этого робот может записать программу. После того, как запись программы выполнена, робот готов начать ее выполнение. Поместите робота на стартовую кодовую карточку, и программа начнет работать. Когда программа заканчивается, робот готов к запуску той же программы или записи новой.

Кодовые карточки


Для программирования базового блока робота вам не нужен компьютер или планшет. Все, что вам нужно, это кодовые карточки и рамки для них! Есть 61 различных кодовых карточек. В комплект входит несколько копий некоторых из них. Есть 108 двусторонних кодовых карточек (в общей сложности 216 сторон). Вы пишете программу, выкладывая последовательность кодовых карточек в рамки. Затем робот проезжает по кодовым карточкам одна за другой. Пока он это делает, оптический датчик идентификации, расположенный на дне робота, сканирует небольшой узор точек, которые вы едва можете увидеть на карточках. Микропроцессор робота запрограммирован на преобразование этого рисунка в инструкции, которым он может следовать.

	Каждая программа всегда начинается с кодовой карточки Start .		Каждая программа всегда заканчивается кодовой карточкой End .				
		Есть карточки, которые заставляют робота двигаться .			Есть карточки, заставляющие поворотный механизм робота вращаться .		
		Есть карточки, которые предлагают роботу издавать звуки .			Есть карточки, которые предлагают роботу включить свет определенным способом или нужным цветом.		
		Есть цифровые карточки, которые предлагают роботу повторить следующую карточку столько раз, сколько на ней изображено.					Есть простые карточки циклов, функций, условий и событий. Ты обо всем этом позже узнаешь из данного руководства.

Вы узнаете правила и возможности других кодовых карточек, следуя инструкциям в уроках этого руководства.
На стр. 9 приведен полный список всех кодовых карт, входящих в комплект, и их функций.




Как использовать простые циклы

В этом комплекте есть два набора простых карточек циклов: зеленые и красные. Это означает, что вы можете использовать до двух циклов в одной программе. Чтобы настроить цикл, вы всегда должны использовать две карточки цикла одного цвета (или две зеленые карточки, или две красные). Одна такая карточка помещается в начало цикла, а другая - в конце цикла. Числовая карточка должна быть размещена сразу после первой карточки цикла. Эта числовая карточка показывает, сколько раз другие кодовые карточки, размещенные после нее, но до второй карточки цикла, будут выполнены (запущены).

 Вы не можете разместить более одной числовой карточки после первой карточки цикла. Вы не можете поместить числовую карточку после второй карточки цикла. Оба этих размещения приведут к ошибке. Вы можете вкладывать один цикл в другой. Вы можете экспериментировать с простыми циклами в Уроке 6.

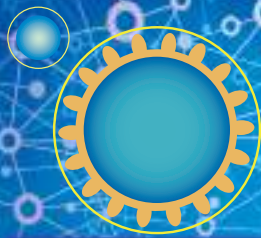
Как использовать функции

На языке кодирования в этом наборе функции показаны красным, зеленым и синим цветом. Эти функции всегда используются с базовыми картами. Например, красная функция выполняется, когда робот сканирует красную базовую карту. Вы можете узнать, как использовать эти функции в Уроке 4.

   **Красная функциональная карточка всегда используется с красной базовой картой**

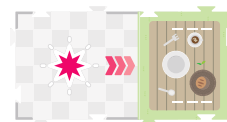
Вы можете иметь до 15 кодовых карточек в функции. Кодовые карточки Движение вперед и Движение назад не работают по назначению; если вы попытаетесь использовать их, вы получите ошибку. Кодовые карточки Поворот по часовой, Поворот против часовой стрелки и Пауза в движении не работают в основной программе. Вы можете использовать их только в функциях или условных утверждениях.

Функции программируются с помощью карт Function Start (Запуск функций): 



Описание комплекта

Функция запускается, когда робот сканирует звезду на соответствующей базовой карте маршрута, когда он направлен в том же направлении, что и три стрелки на базовой карте маршрута, и когда в его памяти записана соответствующая программа соответствующей функции.



Робот должен быть запрограммирован так, чтобы он был обращен в сторону положения взаимодействия (то есть, обращен в том же направлении, что и три стрелки). Робот может либо войти на карту маршрута уже обращенный в этом направлении, либо его можно повернуть с помощью кодовой карточки поворота, чтобы повернуть его в этом направлении после входа на карту. Когда робот сканирует базовую карту маршрута, он сначала ориентируется на звезду. Затем робот продвигается к точке взаимодействия, следуя за тремя стрелками.

Затем функция запускается. Наконец, робот снова возвращается к звезде.

Если вы хотите использовать выходной механизм с моделью на базовой карте, вам нужно закрепить модель полосками карты, чтобы она оставалась на месте:



Как использовать карточки условий и событий

На языке кодирования в этом комплекте условные операторы устанавливаются как подпрограммы, отдельно от основной программы. Условное утверждение должно начинаться с карточки If. Эта карточка имеет маленький зеленый флажок для обозначения начала подпрограммы. Эта карточка должна следовать одна, и только одна из четырех карточек событий.

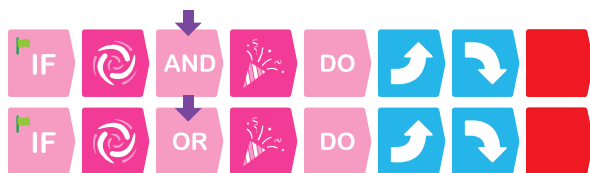


Условное утверждение также должно иметь карточку Do. Её также можно назвать визитная карточка. Карточка Do должна сопровождать кодовые карточки для выкладки кода,



который должен запускать робот, если утверждение If истинно. Другими словами она необходима, если робот сканирует карту (ы) событий, чтобы утверждение If было истинным. Вы можете использовать карту Else после оператора Do, чтобы сообщить роботу, что делать, если оператор If не правда. Вы можете иметь до 15 кодовых карточек после карточки Do и до 15 кодовых карточек после карточки Else.

Вы также можете добавить карточку End и Or в условие. При использовании эти карточки должны следовать по одной, и только по одной из четырех карточек событий.

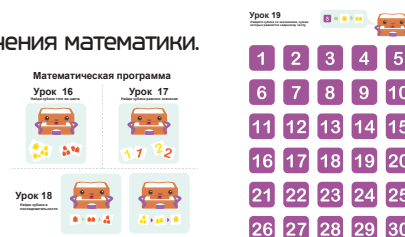


Вы можете поэкспериментировать с условными утверждениями и событиями, которые начинаются в уроке 7.

Режим урока математики

Базовый блок робота можно переключать в специальные режимы для обучения математики.

В этих режимах робот ведет себя иначе, чем в обычном режиме работы. Вы переключаете робота в эти режимы путем сканирования дополнительной невидимой управляющей графики, напечатанной на стр. 81 и 82.



Это как кодовые карточки, но они напечатаны в этой книге, а не на отдельных картах. В математическом режиме вы программируете робота так же, как и раньше, но на этот раз с целью решения поставленной математической задачи. В математическом режиме, когда робот достигает конца своей программы, он будет воспроизводить музыку и загораться в зависимости от того, было ли окончательное решение правильным или неправильным: музыка арфы и разноцветные огни будут воспроизводиться, если решение было правильным. Музыка трубы и красно-оранжевые огни будут воспроизводиться, если решение было неправильным.

В математическом режиме используются карты с напечатанными на них оранжевыми и желтыми кубиками. Эти карточки обозначают цифры от 1 до 5 оранжевым и желтым цветом. Чтобы завершить каждый урок по математике, напишите код для решения указанной проблемы, переместив робота в карты с конкретными номерами и, наконец, с синей, красной или зеленой карты базовой функции.

Примечание. В математическом режиме нельзя использовать кодовые карточки функций или карты условных кодов. Робот не будет реагировать на карты событий в математическом режиме. Чтобы выйти из математического режима, нажмите и удерживайте кнопку стирания в течение двух секунд.

Маршрутные карты

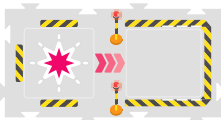
На маршрутных картах робот всегда проигрывает (или запускает) свои программы. Маршрутные карты также имеют невидимые узоры точек, напечатанные на них. Робот использует оптический датчик идентификации для считывания этих рисунков, которые сообщают ему, на какой карте он находится, и помогают ему ориентироваться и двигаться в правильных направлениях на маршрутных картах. Для каждой программы, которую вы пишете, вы всегда выкладываете сетку маршрутных карт, чтобы робот мог запустить свою программу.



Робот всегда начинает свою программу на маршрутной карте Start (Старт).



Некоторые карты представляют события, которые вызывают специальные функции.



Несколько маршрутных карт больше, чем другие. Они называются базовые маршрутные карты. Вы присоединяете определенные модели к этим картам с помощью полосок для маршрутных карт, для того чтобы робот мог взаимодействовать с ними.

Всего в наборе 38 различных карт, в том числе 2 стартовые карты, 6 базовых маршрутных карт и 4 маршрутные карты событий. Маршрутные карты являются двухсторонними, так что на самом деле есть только 19 отдельных карт, каждая с одной картой маршрута на лицевой стороне и одной на обратной.

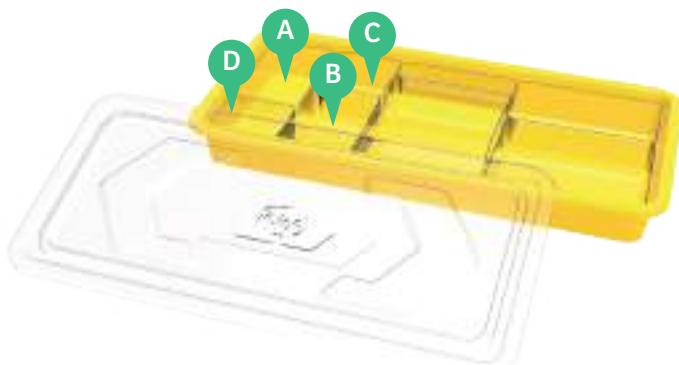
Маршрутные карты имеют взаимосвязанные вкладки, такие как у пазла, чтобы можно было их собрать и держать вместе. Обратите внимание, что вы должны либо использовать лицевые стороны, либо обратные стороны карт одновременно! Потому что вкладки будут правильно совмещаться, только если все карты перевернуты на совместимые стороны. На стр. 11 приведен полный список всех карт, входящих в комплект, и их функций.

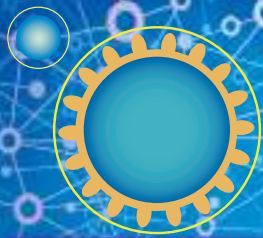
Как хранить карты

Мы разработали полезный лоток, чтобы вы могли легко найти и использовать кодовые карточки. Мы предлагаем вам разделить их на две категории:

Кодовые карточки: при хранении их следует размещать в отсеках для хранения, помеченных буквами А и В на картинке ниже. При необходимости их можно легко вынуть и поместить в меньшие ячейки, отмеченные С и D.

Карты: все маршрутные карты и рамки кодовых карт можно хранить в большой области лотка, чтобы защитить их от повреждений.





Основные действия

Включение и выключение робота

1. Убедитесь, что родители или взрослые установили батарейки в базовый блок робота.
2. Переключите переключатель питания, расположенный на нижней части базового блока робота, в положение «включено».
3. Робот загорится и начнет воспроизводить звуки запуска.
4. Робот теперь готов к записи. Он будет пульсировать синим светом.

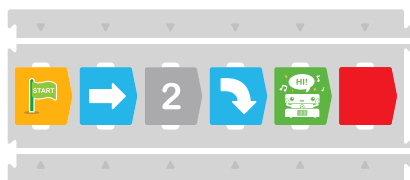
Если вы не используете робота в течение 5 мин., нужно выключить питание, сдвинув переключатель в положение «выключено», чтобы сэкономить энергию батареи или вынуть батареи для безопасности. При выключении робота программы стираются.

Если вы не используете робота в течение пяти минут, он автоматически перейдет в спящий режим. Программы сохраняются, пока робот спит. Вы можете нажать любую кнопку, чтобы активировать базовый блок. Когда батареи разряжаются, робот предупредит вас мигающим оранжевым светом и издаст звук о низкой мощности.

Запись программы

Вы программируете робота, выкладывая серию кодовых карточек для робота, чтобы он проехал по ним и записал программу. Вот как это сделать:

1. Убедитесь, что робот включен и готов к записи.
2. Разложите серию кодовых карточек в рамке для них. Основная программа может иметь до 30 кодовых карточек, не включая карточки Начало (Start) и конец (End).



Подпрограммы, или функции, представлены на стр. 4. Функции могут иметь до 15 кодовых карточек. Если ваш стол слишком короткий, чтобы разместить все кадры подряд, нет проблем! Вы можете записать любую программу в сегментах. Робот не остановит запись, пока не отсканирует карту с кодом конца. Таким образом, вы можете отсканировать одну строку кода, и робот сделает паузу в

конце. Затем вы можете переместить робота в другой ряд, и он автоматически продолжит запись. Если ваш робот сканирует максимальное количество карточек, но не сканирует карточку завершения, робот автоматически завершит программу.

3. Поместите робота прямо над кодовой карточкой «Пуск» в начале строки, лицом к остальной части строки.
4. Нажмите кнопку записи.
5. Робот будет пульсировать красным светом, его кнопка записи будет пульсировать зеленым, и он будет воспроизводить музыку, показывая, что он записывает. В то же время робот будет двигаться вперед по кодовым карточкам, сканируя их и записывая программу.
6. Для каждой успешно записанной кодовой карточки робот будет воспроизводить звук.
7. Если у робота возникнут какие-либо проблемы во время записи, он будет мигать оранжевым и красным и воспроизводить звук ошибки. Это может произойти, если робот движется в неправильном направлении или если кодовые карточки были расположены в неправильном порядке.
8. Когда робот достигает карты Конец (End) и сканирует ее, он прекращает движение и воспроизводит звук окончания записи.
9. Теперь робот будет готов запустить свою программу. Индикатор кнопки записи теперь будет гореть постоянно зеленым светом.
10. Если есть функция подпрограммы для программирования, поместите робота на карту Function Start (запуск функции) и нажмите Record Button (кнопку записи). Робот запоминает одну основную программу и до трех функций одновременно в своей памяти.

Запуск программы

После того, как программа была записана, вы можете запускать ее. Вот как это сделать:

Поместите робота на карту Start (стартовую карту), обращенным в направлении стрелок.

1. Нажмите кнопку Run («Запуск»). «Запись» и «Запуск» - это одна и та же кнопка. Робот знает, стоит ли он на записи или запуске в зависимости от того, находится ли он на кодовой карточке или на маршрутной карте.
3. Теперь робот будет выполнять программу. Сначала робот будет двигаться на месте на карте Start, чтобы сориентироваться. Не трогайте его! Затем будет перемещаться по маршрутным картам по той программе, которая записана. Важно, чтобы он оставался выровненным по картам на протяжении всей программы. Если в это время на роботе не записана основная программа, он будет мигать поочередно красным и оранжевым светом и воспроизводить предупреждающий звук. Когда робот выполняет программу, он будет воспроизводить фоновую музыку, если только программа не скажет ему воспроизводить другую музыку.
4. Когда робот достигает и сканирует определенные карты маршрутов, такие как карты Событие (событий) или карта Base карт маршрута, он может запускать особые действия или функции.

После запуска программы робот все еще ее помнит; программа не стирается автоматически после запуска. Вы можете снова запустить программу или записать или перезаписать ее или функцию.

Перезапись программы

Робот может одновременно выполнять только одну основную программу и одну из функций каждой подпрограммы. Если вы хотите, чтобы робот записал новую программу или функцию (начиная с карты Start или с одной из карт Function Start), когда программа или функция уже сохранены, робот перезапишет старую основную программу или функцию.

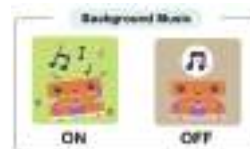
Это означает, что старая программа стирается, а новая ее заменяет. Если вы хотите пересмотреть основную программу или функцию, вы можете перезаписать их по одной за раз; другие программы сохраняются.

Стирание программ

Чтобы полностью стереть все программы в роботе (и выйти из математического режима), нажмите и удерживайте кнопку Erase (стереть) в течение двух секунд или дольше. Индикатор робота будет мигать красным в течение нескольких секунд, а затем остановится, показывая, что память программы была очищена.

Фоновая музыка

Если вы хотите включить или выключить фоновую музыку, отсканируйте изображение кода фоновой музыки на стр. 81. Фоновая музыка включена по умолчанию. Графика выглядит так:



Уроки

Лучший способ узнать, что делают все кодовые карточки и как они работают вместе, следуя урокам, изложенным в этом пособии. Для каждого урока вы сначала строите несколько моделей. Пошаговые инструкции по сборке печатаются перед уроками, в которых они используются. Затем вы раскладываете сетку маршрутных карт в точности так, как показано в уроке, а также выкладываете последовательности кодовых карточек в рамке(ках). Затем запишите и запустите программу и посмотрите, что делает робот. Все ли работало верно? Поздравляем! Если нет, вы должны пройти процесс отладки, чтобы исправить физическую модель, кодовые карточки и / или карты маршрутов, пока робот не будет работать отлично!

Для вашей справки: этот пакет предоставляет пример программ S4A. Пример версии S4A версии 1.6. Обратите внимание, что могут быть некоторые Интерфейсные различия между альтернативными версиями программы.

<https://www.dropbox.com/sh/4h4z3wiq9hk93ue/AADBK7YkZQpuXYf-pPjDT2wja?dl=0>



QR КОД

Основные действия

Определение кодовых карточек

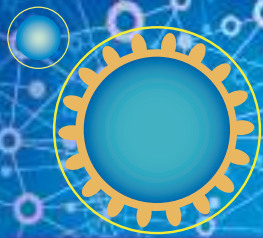
Каждая кодовая карточка представляет собой функцию или часть кода, который сообщает мотору, свету и динамике робота, что делать. Здесь приведены спецификации для каждой кодовой карточки и их количество в комплекте, считая обе стороны.

Название изображения	Описание	Кол-во
	Старт (Start) Каждая основная программа должна начинаться с этой карточки. Используется только в основной программе.	4
	Конец (End) Каждая программа, включая основную и функциональную, должна заканчиваться этой карточкой.	10
	Старт «Красной функции» Красная функциональная программа должна начинаться с этой карточки. Эта функция вызывается, когда робот сканирует совпадающую базовую маршрутную карту (красная звезда).	2
	Старт «Зеленой функции» Зеленая функциональная программа должна начинаться с этой карточки. Эта функция вызывается, когда робот сканирует совпадающую базовую маршрутную карту (зеленая звезда).	2
	Старт «Синей функции» Синяя функциональная программа должна начинаться с этой карточки. Эта функция вызывается, когда робот сканирует совпадающую базовую маршрутную карту (синяя звезда).	2
	If (условный элемент) Это стартовая карточка для условной (if-then) функции. Когда робот сканирует карту событий Событие, которая удовлетворяет условию, функция выполняется.	2
	Do (условный элемент) Эта карточка может использоваться только с картой if в условной функции. Если условие выполняется, запускается последовательность после карты Do.	2
	Else (условный элемент) Эта карточка может использоваться только с картой if в условной функции. Если условие не выполняется, запускается последовательность после карты Else. Примечание: двигаться вперед (Move Forward), (двигаться назад) Move Обратная сторона, поворот направо (Turn Right), поворот налево (Turn Left) и пауза (Pause Movement) не могут использоваться в инструкции Else после карты Else.	2
	And (условный элемент) Эта карточка может использоваться только с картой If в условной функции. При использовании двух условий должны быть выполнены для запуска функции.	1
	Or (элемент условия) Эта карточка может использоваться только с карточкой If в условной функции. Для запуска функции должно выполняться хотя бы одно условие.	1
	Событие 1 (условный элемент) Используемая в условной функции, эта карточка определяет условие, которое должно быть выполнено для запуска функции. Робот должен отсканировать соответствующую карточку.	1
	Событие 2 (элемент условия) Используемая в условной функции, эта карточка определяет условие, которое должно быть выполнено для запуска функции. Робот должен отсканировать соответствующую маршрутную карту.	1
	Событие 3 (элемент условия) Используемая в условной функции, эта карточка определяет условие, которое должно быть выполнено для запуска функции. Робот должен отсканировать соответствующую маршрутную карту.	1
	Событие 4 (элемент условия) Используемая в условной функции, эта карточка определяет условие, которое должно быть выполнено для запуска функции. Робот должен отсканировать соответствующую маршрутную карту.	1

Название изображения	Описание	Кол-во
	Простой зеленый цикл Start/End Эти две карточки позволяют вам повторить последовательность кодов, расположенных между ними определенного количества раз, соответствующее числовой карточке.	4
	Простой красный цикл Start/End Эти две карточки позволяют вам повторить последовательность кодов, расположенных между ними определенного количества раз, соответствующее числовой карточке.	4
	Двигаться вперед Эта карточка говорит роботу переместиться вперед на одну маршрутную карту. Она может использоваться только в основной программе. Она может быть повторена с помощью числовой карточки.	24
	Двигаться назад Эта карточка говорит роботу переместиться вперед на одну маршрутную карту. Она может использоваться только в основной программе. Она может быть повторена с помощью числовой карточки.	24
	Поворот направо (по часовой стрелке) Эта карточка говорит роботу повернуть на 90 градусов направо. Она может быть повторена с помощью числовой карточки.	18
	Поворот налево (против часовой стрелки) Эта карточка говорит роботу повернуть на 90 градусов налево. Она может быть повторена с помощью числовой карточки.	18
	Пауза в движении Эта карточка говорит роботу остановиться на одну секунду. Она может использоваться только в основной программе. Она может быть повторена с помощью числовой карточки.	4
	Вращать выходной механизм по часовой стрелке Эта карточка говорит роботу включить передачу и вращать выходной механизм по часовой стрелке одну секунду. Она может использоваться только в функции. Она может быть повторена с помощью числовой карточки.	5
	против часовой стрелки. Эта карточка говорит роботу включить передачу и вращать выходной механизм против часовой стрелке одну секунду. Она может использоваться только в функции. Она может быть повторена с помощью числовой карточки.	5
	Пауза выходного механизма Эта карточка говорит роботу сделать паузу в работе выходного механизма на одну секунду. Она может использоваться только в функции. Она может быть повторена с помощью числовой карточки.	4
	Воспроизвести звук: Привет!(Hi!) Эта карточка говорит роботу воспроизвести звук Привет! Hi. Она может быть повторена с помощью числовой карточки.	2
	Воспроизвести звук: А-хх Эта карточка говорит роботу воспроизвести звук А-хх, как если бы он был счастлив. Она может быть повторена с помощью числовой карточки.	2
	Воспроизвести звук: А? Эта карточка говорит роботу воспроизвести звук А?, как, если поставить под сомнение. Она может быть повторена с помощью числовой карточки.	2
	Воспроизвести звук: А-аа Эта карточка говорит роботу воспроизвести звук А-аа, как будто он расстроен. Она может быть повторена с помощью числовой карточки.	2
	Воспроизвести звук: мыши Эта карточка говорит роботу пищать подобно мыши. Она может быть повторена с помощью числовой карточки.	2
	Воспроизвести звук: пингвина Эта карточка говорит роботу воспроизвести вопящий крик пингвина. Она может быть повторена с помощью числовой карточки.	2

Название изображения	Описание	Кол-во
	Воспроизвести звук: аплодисменты Эта карточка говорит роботу воспроизвести звук ликующей толпы. Она может быть повторена с помощью числовой карточки.	2
	Воспроизвести звук: фабрика Эта карточка говорит роботу воспроизвести звук машин на фабрике. Она может быть повторена с помощью числовой карточки.	2
	Воспроизвести звук: пожарный шланг Эта карточка говорит роботу воспроизвести звук пожарного шланга, из которого бьет струя воды. Она может быть повторена с помощью числовой карточки.	2
	Воспроизвести звук: сирена Эта карточка говорит роботу воспроизвести звук сирены аварийного автомобиля. Она может быть повторена с помощью числовой карточки.	2
	Цвет света: синий Эта карточка говорит роботу изменить цвет света выходного механизма на синий на одну секунду. Она может быть повторена с помощью числовой карточки.	2
	Цвет света: фиолетовый Эта карточка говорит роботу изменить цвет света выходного механизма на фиолетовый на одну секунду. Она может быть повторена с помощью числовой карточки.	2
	Цвет света: розовый Эта карточка говорит роботу изменить цвет света выходного механизма на розовый на одну секунду. Она может быть повторена с помощью числовой карточки.	2
	Цвет света: красный Эта карточка говорит роботу изменить цвет света выходного механизма на красный на одну секунду. Она может быть повторена с помощью числовой карточки.	2
	Цвет света: оранжевый Эта карточка говорит роботу изменить цвет света выходного механизма на оранжевый на одну секунду. Она может быть повторена с помощью числовой карточки.	2
	Цвет света: желтый Эта карточка говорит роботу изменить цвет света выходного механизма на желтый на одну секунду. Она может быть повторена с помощью числовой карточки.	2
	Цвет света: зеленый Эта карточка говорит роботу изменить цвет света выходного механизма на зеленый на одну секунду. Она может быть повторена с помощью числовой карточки.	2
	Цвет света: радуга Эта карточка говорит роботу циклически переключать цвет света выходного механизма на семь цветов радуги по полсекунды каждый. Она может быть повторена с помощью числовой карточки.	2
	Световой эффект: дискотека Эта карточка говорит роботу, чтобы свет в выходном механизме мигал очень быстро и был малинового цвета по умолчанию в течение трех секунд. Она может быть повторена с помощью числовой карточки.	2
	Световой эффект: аварийный свет автомобиля Эта карточка говорит роботу, чтобы свет в выходном механизме светился, как сигнал аварийной машины, малиновым цветом по умолчанию в течение трех секунд. Она может быть повторена с помощью числовой карточки.	2
	Световой эффект: падающая звезда Эта карточка говорит роботу, чтобы свет в выходном механизме мигал быстро-медленно-быстро малиновым цветом по умолчанию в течение трех секунд. Она может быть повторена с помощью числовой карточки.	2
	Световой эффект: мерцающая звезда Эта карточка говорит роботу, чтобы свет в выходном механизме горел непрерывно с небольшим мерцанием фиолетовым цветом по умолчанию в течение трех секунд. Она может быть повторена с помощью числовой карточки.	2

Название изображения	Описание	Кол-во
	Световой эффект: светлячок Эта карточка говорит роботу, чтобы выходной механизм светился, как у светлячка, малиновым цветом по умолчанию в течение трех секунд. Она может быть повторена с помощью числовой карточки.	2
	Световой эффект: медленное мигание Эта карточка говорит роботу, чтобы свет в выходном механизме мигал медленно включаясь и выключаясь малиновым цветом по умолчанию в течение трех секунд. Она может быть повторена с помощью числовой карточки.	2
	Световой эффект: мигание со средней частотой Эта карточка говорит роботу, чтобы свет в выходном механизме мигал со средней частотой малиновым цветом по умолчанию в течение трех секунд. Она может быть повторена с помощью числовой карточки.	2
	Световой эффект: быстрое мигание Эта карточка говорит роботу, чтобы свет в выходном механизме быстро мигал малиновым цветом по умолчанию в течение трех секунд. Она может быть повторена с помощью числовой карточки.	2
	Световой эффект: ускорение Эта карточка говорит роботу, чтобы свет в выходном механизме мигал с ускорением малиновым цветом по умолчанию в течение трех секунд. Она может быть повторена с помощью числовой карточки.	2
	Световой эффект: замедление Эта карточка говорит роботу, чтобы свет в выходном механизме мигал с замедлением малиновым цветом по умолчанию в течение трех секунд. Она может быть повторена с помощью числовой карточки.	2
	Карточки с цифрами от 1 до 9 Эти карточки говорят роботу повторить коды карточек по количеству раз, напечатанному на числовой карточке, когда числовая карточка помещается сразу после карточки с кодом в последовательности. Это работает только со следующими кодовыми карточками:	2
		2
	<ul style="list-style-type: none"> Начало простого цикла (но не «конец» простого цикла) Двигаться вперед и двигаться назад Поворот направо и поворот налево Пауза в движении 	2
	<ul style="list-style-type: none"> Вращение выходного механизма по часовой стрелке и против часовой стрелки Пауза выходного механизма Все карточки, воспроизводящие звук Все карточки цветов света Все карточки световых эффектов 	2
	Нельзя размещать последовательно больше, чем одну числовую карточку (одну за другой без прерывания) в программе, иначе робот выдаст ошибку.	2
	<ul style="list-style-type: none"> Номер 1: выполнить предшествующую кодовую карточку 1 раз Номер 2: выполнить предшествующую кодовую карточку 2 раза Номер 3: выполнить предшествующую кодовую карточку 3 раза Номер 4: выполнить предшествующую кодовую карточку 4 раза Номер 5: выполнить предшествующую кодовую карточку 5 раз Номер 6: выполнить предшествующую кодовую карточку 6 раз Номер 7: выполнить предшествующую кодовую карточку 7 раз Номер 8: выполнить предшествующую кодовую карточку 8 раз Номер 9: выполнить предшествующую кодовую карточку 9 раз 	2
		2
		2
		2



Основные действия

Перечень маршрутных карт

В этот комплект входят четыре основных типа маршрутных карт. Маршрутные карты не являются взаимозаменяемыми, так как у каждой есть специальный невидимый рисунок. Вы должны убедиться, что используете правильные карты в правильных местах. Вот как они выглядят.

Маршрутные карты старта	
Лицевая сторона	Обратная сторона

Базовые маршрутные карты	
Лицевая сторона	Обратная сторона
Карта красной функции/Лицевая сторона	Карта красной функции/Обратная сторона
Карта зеленой функции/Лицевая сторона	Карта зеленой функции/Обратная сторона
Карта синей функции/Лицевая сторона	Карта синей функции/Обратная сторона

Маршрутные карты событий*	
Лицевая сторона	Обратная сторона
Событие 2	Событие 4
Событие 3	Событие 1

*Note: Событие map cards cause the robot to perform a default action when they are scanned and no matching Событие КОД card has been used.

Общие маршрутные карты	
Лицевая сторона	Обратная сторона

Общие маршрутные карты	
Лицевая сторона	Обратная сторона

Общие маршрутные карты	
Лицевая сторона	Обратная сторона

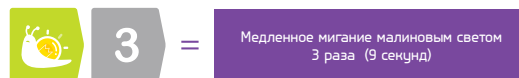
Комбинирование световых карт

Карточки цветов света и световых эффектов могут быть скомбинированы вместе в программе, чтобы получить более сложные результаты.

Поместите числовую карточку после карточки цвета света, чтобы изменить количество раз включения, продлевая время свечения.



Поместите числовую карточку после карточки светового эффекта, чтобы изменить количество раз, которое световой эффект выполняет, тем самым продлевая его.



Примеры

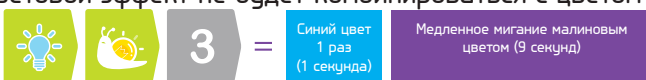
Если вы скомбинируете карточки вместе, как показано ниже, световой эффект запускается два раза, а затем свет меняется на синий и запускается три раза, что в данном случае составляет около трех секунд.



Если вы скомбинируете карточки вместе следующим образом, как показано ниже, цвет светового эффекта будет синим вместо малинового по умолчанию, и он будет работать три раза, или в этом случае около девяти секунд. Когда карточка со световым эффектом появляется перед карточкой цвета света, цвет светового эффекта меняется.



Если вы поместите карточки, как показано ниже, световой эффект не будет комбинироваться с цветом света. Синий свет будет светиться в течение одной секунды, а затем световой эффект будет запущен три раза в малиновом цвете по умолчанию. Когда карточка со световым эффектом появляется после карточки с цветом света, эти две карточки не комбинируются.



Не уверены, как все это комбинировать? Если есть сомнения, попробуйте!

Советы по устранению неполадок

Если ваш робот не записывает:

- Убедитесь, что вы запускаете свою программу с помощью карточки Start, запуска функции или кодовой карточки.
- Убедитесь, что батареи вашего робота заряжены, и робот не выдает предупреждение о низкой мощности.
- Убедитесь, что робот направлен в правильном направлении, следуя стрелкам на рамке кодовой карточки.

Если ваш робот ведет себя странно или не работает должным образом:

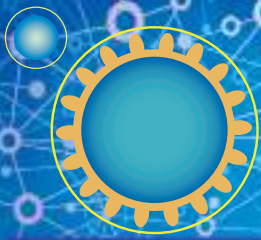
- Убедитесь, что батареи достаточно заряжены. Когда батареи разряжаются, робот предупреждает вас мигающим оранжевым светом и воспроизводит звук индикатора низкой мощности.
- Пыль, пятна или выцветание на поверхности карточек могут помешать чтению рисунков невидимых кодов. Пожалуйста, держите карточки в чистоте и сухости.
- Если ваш робот не может записать кодовую карточку запуска функции или кодовую карточку If, робот может быть в математическом режиме. Удерживайте кнопку «Стереть» в течение двух секунд, чтобы вернуться в обычный режим.

Если ваш робот мигает оранжевым и останавливается в середине строки кодовых карточек:

- Если у робота возникнут какие-либо проблемы во время записи, он будет мигать оранжевым и красным и воспроизводить звук ошибки. Проверьте кодовые карточки и убедитесь, что они в правильном порядке.

Если ваш робот делает неожиданные движения, когда он запускает программу:

- Это нормально. Робот калибрует свою позицию. Если вы поместите робота в центр карты Start, у него будет меньше время калибровки.
- Процесс калибровки повышает точность движений робота. Не перемещайте робота во время процесса калибровки.
- Робот выполняет быструю калибровку при нажатии кнопки «Выполнить» (Run). Робот выполняет стандартную калибровку при первом нажатии кнопки Run после включения робота или при нажатии и удерживании кнопки Run в течение двух секунд. Основная программа выполняется после калибровки.



Список деталей



Список деталей:

№	Название детали	Номер детали	Кол.	№	Название детали	Номер детали	Кол.
1	Элемент осевой	7026-W10-H1K	5	29	Призма плоская (белая)	880-W10-S1W	4
2	Штифт малый синий	7344-W10-C2B	30	30	Призма плоская (синяя)	880-W10-S1B2	2
3	Втулка соединительная, 2см	7413-W10-T1B	5	31	Призма плоская (малиновая)	880-W10-S1K1	4
4	Червячное колесо	7344-W10-A1S1	1	32	Призма плоская (оранжевая)	880-W10-S1O3	4
5	Шестеренка Z20	7026-W10-D2K	4	33	Призма вогнутая (белая)	880-W10-D1W	4
6	Шестеренка Z60	7026-W10-W5B	2	34	Призма вогнутая (синяя)	880-W10-D1B2	4
7	Ось мотора короткая	7026-W10-L1D	2	35	Призма вогнутая (оранжевая)	880-W10-D1O3	6
8	Ось средняя, 6 см	7413-W10-M1D	1	36	Кнопка фиксатор	7061-W10-W1W	12
9	Ось длинная, 10 см	7413-W10-L2D	1	37	Трубка полая, 3 см	7400-W10-G1D	2
10	Круглый стержень, 3 отверстия	7404-W10-C1W	2	38	Шаровая деталь	7128-W10-E1K	1
11	Круглый стержень, 7 отверстий	7404-W10-C2W	2	39	Четырехгранная пирамида	7128-W10-E4O1	1
12	Удлиненный стержень, 7 отверстий	7404-W10-C3W	1	40	Глаз	7128-W22-2	8
13	Арочная рамка малиновая, 5x5	7411-W10-F1K	2	41	Ключ разборочный	7061-W10-B1Y	1
14	Арочная рамка белая, 5x5	7411-W10-F1W	2	42	Втулка соединительная, 3 см	7413-W10-U1S	1
15	Кубик (белый)	880-W10-A1W	16	43	Каркас Семёна, нижняя часть	7442-W10-G2T1	1
16	Кубик (синий)	880-W10-A1B2	20	44	Каркас Семёна, верхняя часть	7442-W10-G1T1	1
17	Кубик (малиновый)	880-W10-A1K1	8	45	Коробка передач Семёна, верх	7442-W10-F1T1	1
18	Кубик (салатовый)	880-W10-A1G1	8	46	Коробка передач Семёна, низ	7442-W10-F2T1	1
19	Кубик (фиолетовый)	880-W10-A1P1	8	47	Рука Семёна, левая	7442-W10-H1P	1
20	Кубик (оранжевый)	880-W10-A1O3	12	48	Рука Семёна, правая	7442-W10-H2P	1
21	Кубик, 6 отв. (белый)	880-W10-N1W	4	49	Глаза Семёна, наклейка	R20#7442	1
22	Кубик, 6 отв. (оранжевый)	880-W10-N1O3	2	50	Картинка для вырезания	K16#7442	1
23	Призма выпуклая (белая)	880-W10-R1W	18	51	Полоски для маршрутных карт	K41#7442	1
24	Призма выпуклая (синяя)	880-W10-R1B1	12	52	Маршрутные карты	K16#1276-1	1
25	Призма выпуклая (малиновая)	880-W10-R1K1	6	53	Базовые маршрутные карты	K16#1276-2	1
26	Призма выпуклая (салатовая)	880-W10-R1YG	12	54	Кодовые карточки	K16#1276-3	1
27	Призма выпуклая (фиолетовая)	880-W10-R1P	4	55	Рамки для кодовых карточек	K16#1276-4	1
28	Призма выпуклая (оранжевая)	880-W10-R1O3	16	56	Роботизированный базовый блок	7442-W85-A	1

Советы и приёмы:

Вот несколько советов по сборке и использованию моделей. Внимательно прочитайте их перед началом.


Для советов по сборке, пожалуйста, обратитесь к 



Рис. 1. Короткие штифты

Используйте ключ разборочный как рычаг, чтобы извлечь короткие штифты, как показано на рис. 1.

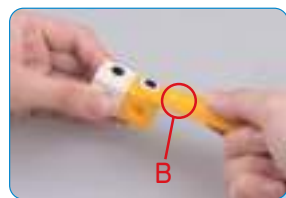
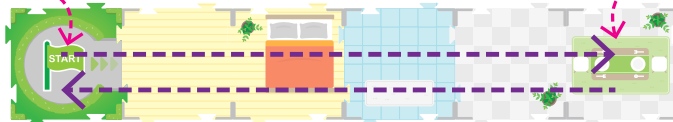


Рис. 2. Глаза

Используйте ключ разборочный как рычаг, чтобы извлечь глаза, как показано на рис. 2



Семён - это робот. А роботы бывают разных форм и размеров. В этот раз Семён превратился в бутерброд с кремом и желе! У Семёна

есть колеса, которые приводятся в движение электродвигателем внутри его роботизированной базовой части. У Семёна также есть малиновые руки, соединенные с механизмами, которые связаны с электродвигателем. Семён и его два хороших друга – мороженое Платон и пудинг Паша любят играть в прятки. Первая программа Семёна – пойти в ресторан, чтобы найти Платона, а затем вернуться к стартовой карте.

Концепция кодирования

Последовательность (или расположение элементов) является одной из самых важных структур в компьютерном программировании. Последовательность представляет собой набор шагов или расположение команд в определенном порядке. Компьютеры выполняют шаги последовательности по порядку, по одному, с целью выполнения конкретной задачи. Когда вы выкладываете кодовые карточки, чтобы сообщить Семёну - роботу что делать и когда это делать, вы создаете последовательность.

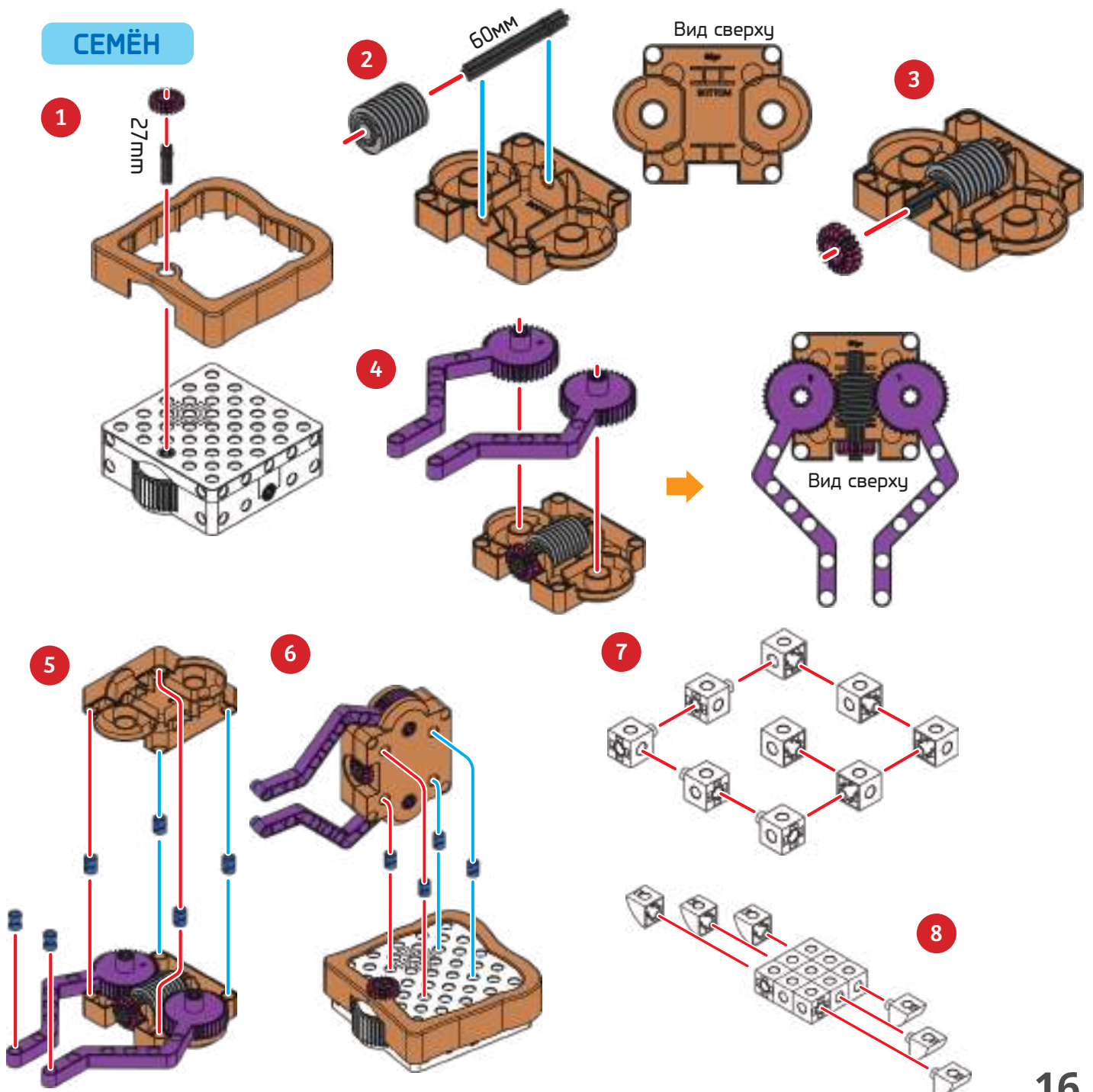
Мозговой штурм

Попробуйте запрограммировать маршрут от двери дома до кухни!

Список деталей

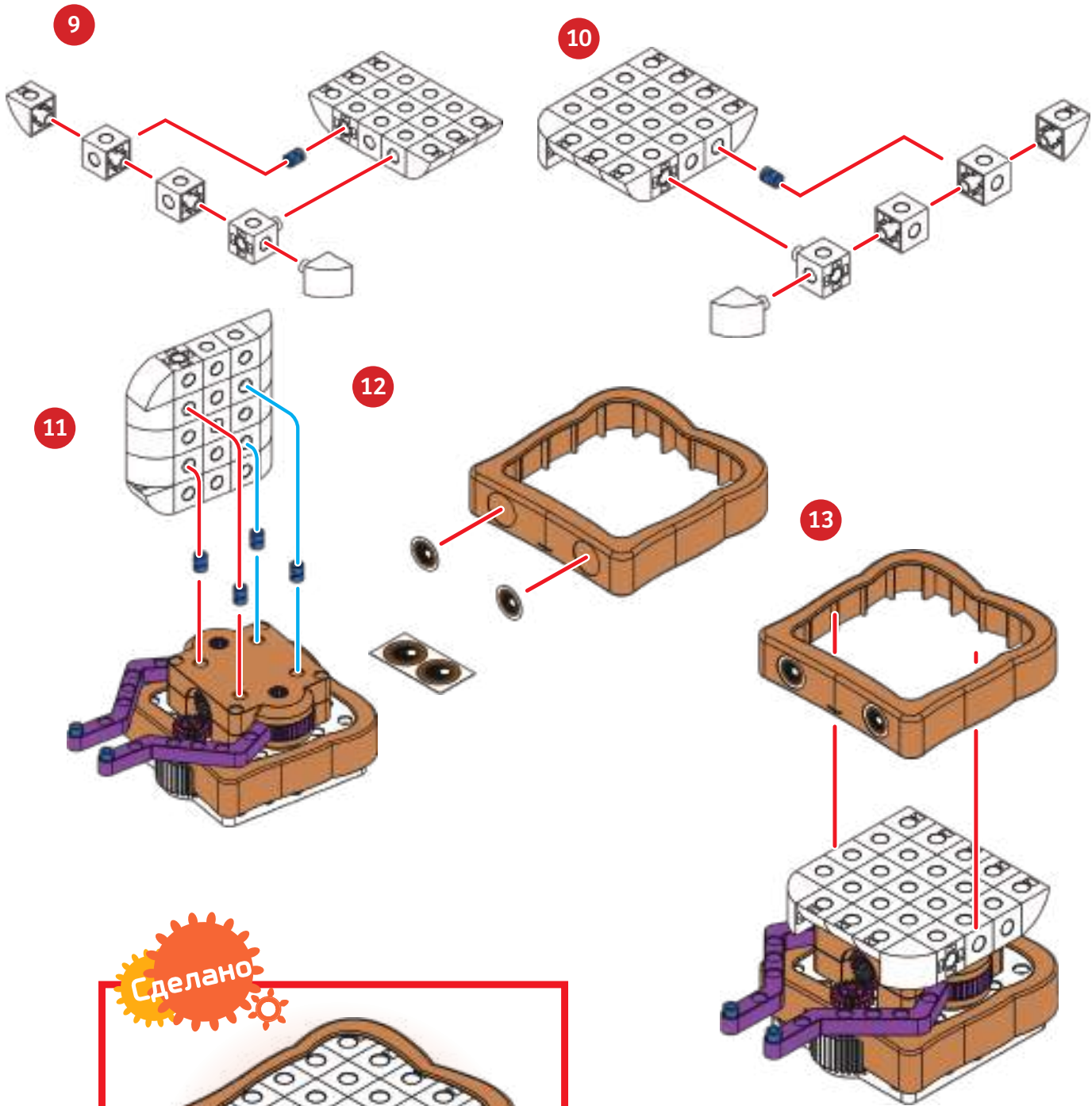
2  x16	4  x1	5  x2	7  x1	8  x1	15  x15	23  x10	43  x1	44  x1	45  x1
46  x1	47  x1	48  x1	49  x1	56  x1					

СЕМЁН

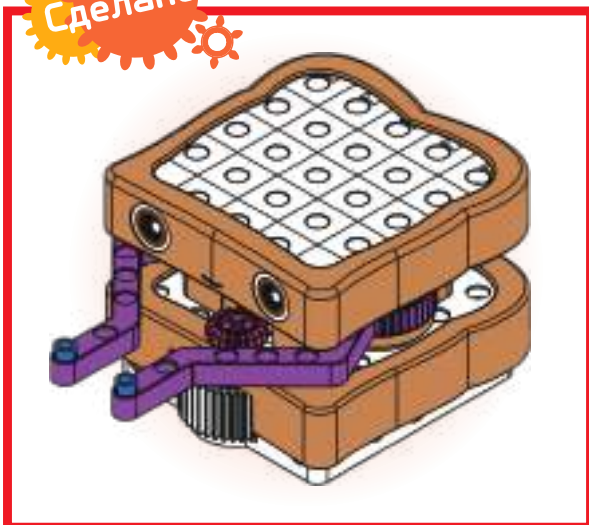


1

Прятки! Помогите Семёну найти Платона.



Сделано



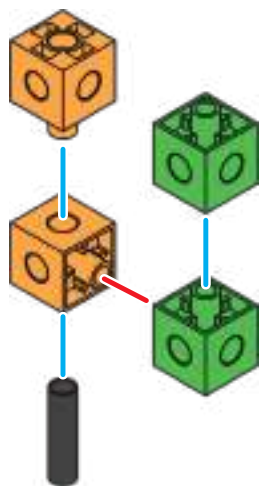
Умное руководство
Веб-сервис

Список деталей

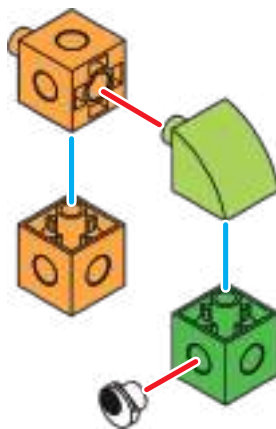


ПЛАТОН

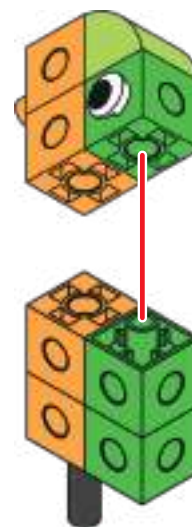
1



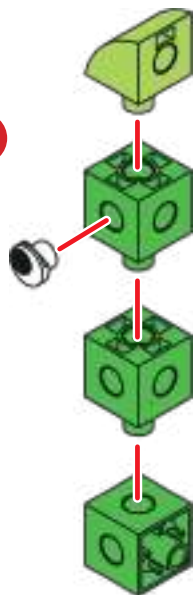
2



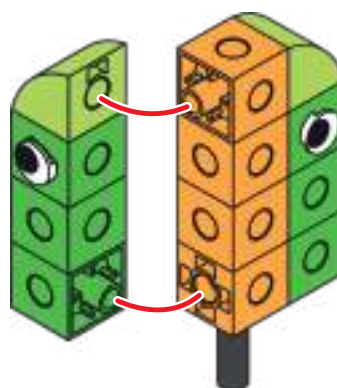
3



4



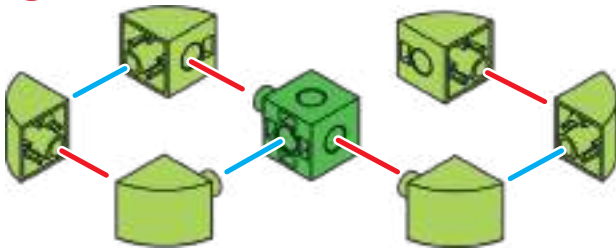
5



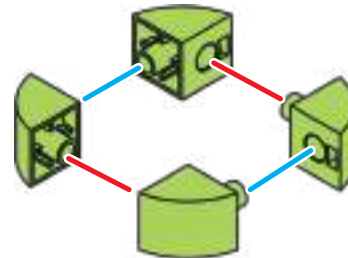
1

Прятки! Помогите Семёну найти Платона.

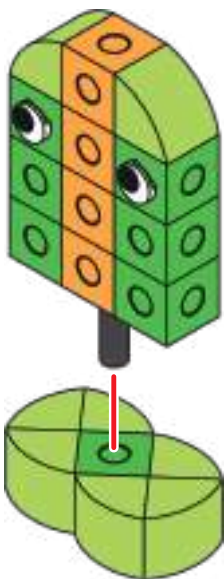
6



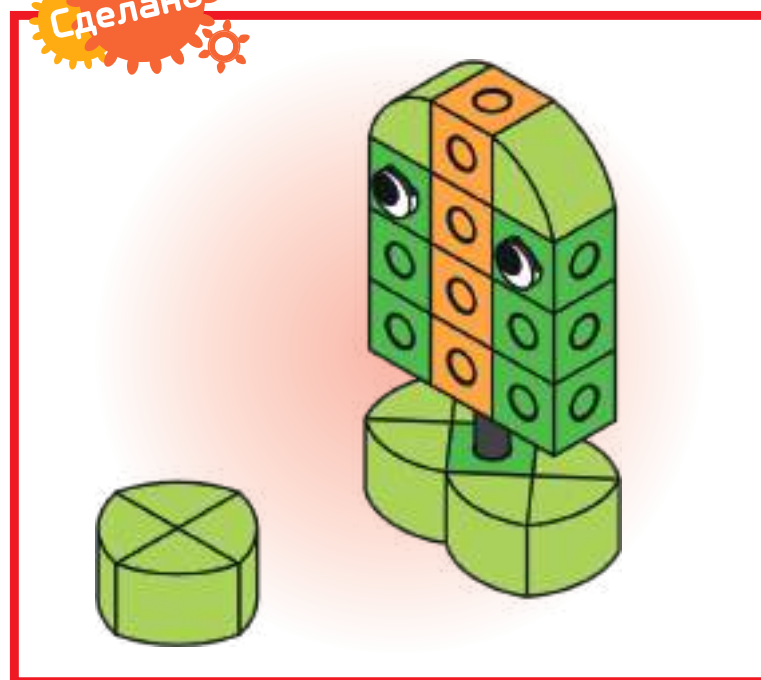
7



8

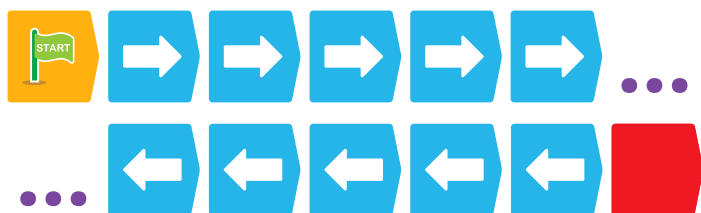


Сделано



Умное руководство
Веб-сервис

[код]



Примечание: точки здесь означают, что программа продолжается на следующей строке, потому что она слишком длинная, чтобы поместиться на одной строке.

Что происходит?

Эта программа использует последовательность из пяти карточек «Вперед» и пяти «Назад». Это приводит к программе, которая перемещает Семёна вперед на пять карт, а затем назад на пять карт.

Попробуйте это! Измените способ возвращения Семёна к началу маршрутной карты. Поручите Семёну двигаться с самого начала маршрутной карты на кухню, затем повернуть назад и вернуться на стартовую карту.



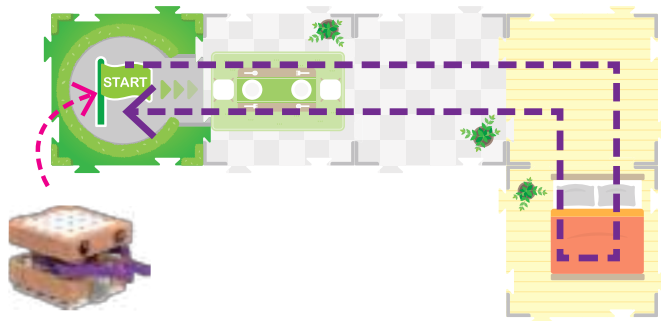
.....

.....



Видео работы модели





Пудинг Паша спрятался так хорошо, что Семён не мог найти его! Наконец Семён пошел в самую внутреннюю комнату и нашел Пашу. Какую самую короткую программу

вы можете написать, чтобы Семён вошел в комнату, где находится Паша? И оставайтесь там две секунды, пока он осматривает комнату, затем возвращайтесь на стартовую карту?

Концепция кодирования

Отладка - это процесс поиска и предотвращения нежелательного поведения в программе, компьютере или роботе. Когда и где робот начал делать что-то другое, не то, что вы хотели сделать? Могли ли кодовые карточки вызвать это? Если это так, попробуйте проверить кодовые карточки по инструкции и внесите изменения. Могли ли маршрутные карты вызвать это? Если это так, попробуйте проверить маршрутные карты по инструкции и внесите изменения. Могла ли конструкция модели вызвать это? Если так, попробуйте проверить модель на соответствие инструкциям и внесите изменения.

Мозговой штурм

Какое правильное существительное используется для ошибок, найденных в компьютерных программах?

Список деталей Семёна и шаги сборки:

Пожалуйста, обратитесь к Уроку 1.



Список деталей

2



х3

16



х2

19



х2

27



х4

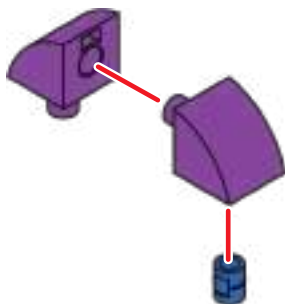
30



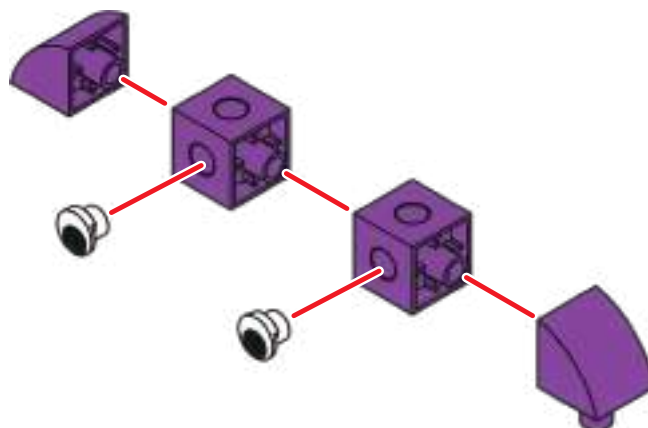
х2

ПАША

1



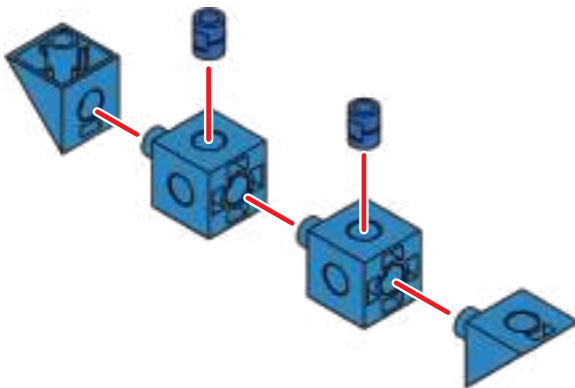
2



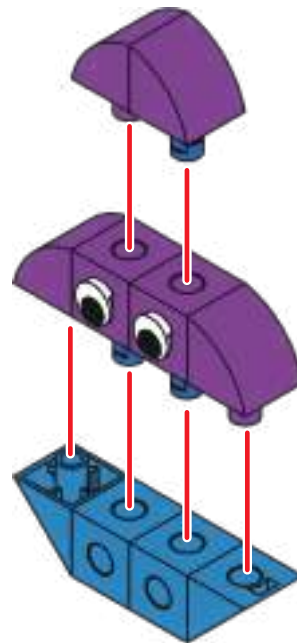
2

Дайте время, чтобы найти в какой комнате спрятался пудинг Паша?

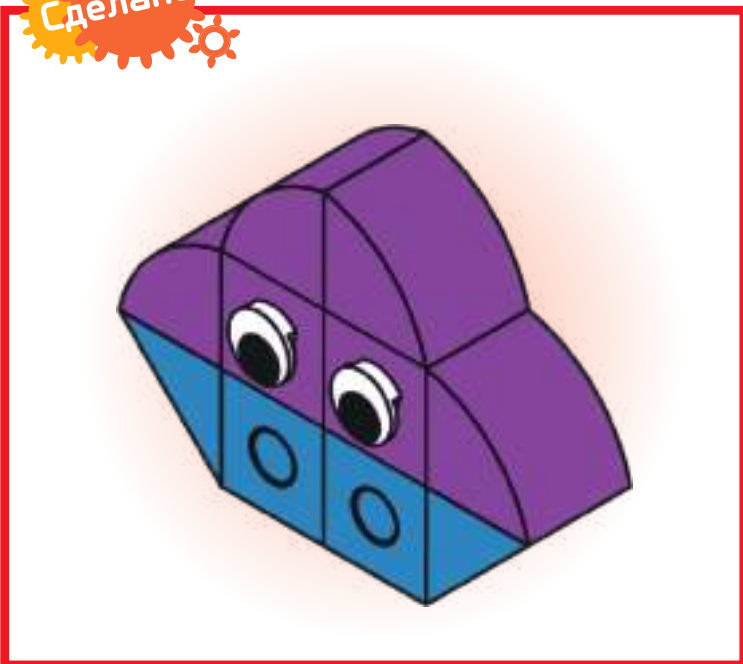
3



4

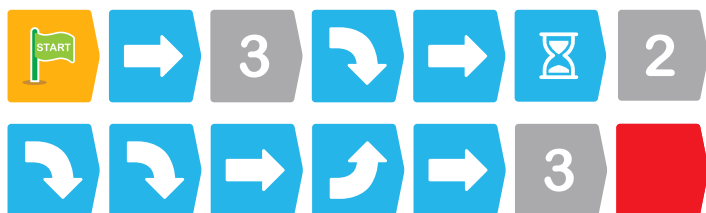


Сделано



Умное руководство
Веб-сервис

[код]



Что происходит?

В этой программе вы будете использовать кодовые карточки Поверните влево и Поверните направо. Сначала Семён перемещается на три карты вперед, затем поворачивает направо и движется вперед на еще одну маршрутную карту в спальню.

Затем числовая карточка 2 выполняет паузу дважды, заставляя Семёна сделать паузу на две секунды вместо одной.

Затем Семён поворачивается направо два раза (на 180 градусов) двигается вперед на одну карту из спальни, поворачивает налево, а затем выходит из дома.

Напишите программу, которая заставит Семёна остаться в комнате на 5 секунд.



.....

.....



Видео работы модели





Семён хочет сделать яблочный пирог, но он даже не знает, как начать. Однако его хорошая подружка

Маша-молочница - мастер по приготовлению десертов! Семён решил обратиться к ней за помощью, получив рецепт яблочного пирога высшего класса. Напишите программу, в которой Семён увидит Машу. Как мы можем запрограммировать это действие с использованием минимального количества карточек с инструкциями? Попробуй сделать это, используя только три вида карточек: движение вперед, номер и / или цикл.

Концепция кодирования

Цикл является еще одним из наиболее важных структур в компьютерном программировании. Цикл - это набор шагов, который повторяется несколько раз. Циклы могут быть запрограммированы на:

- повторение одного или нескольких действий установленного количества раз;
- повторение действия навсегда (бесконечный цикл);
- повторение только тогда, когда происходит что-то еще (пока цикл) или пока не произойдет что-то другое (до цикла).

Циклы важны, потому что программистам часто нужно иметь определенные задачи или операции, которые повторяются несколько раз. Если программист должен был писать один и тот же код снова и снова, код был бы длинным и запутанным. Поэтому программисты обычно работают как следует над созданием самого короткого, понятного и простого кода.

Мозговой штурм

Какой тип цикла используется в этом уроке? Это просто счетный цикл, бесконечный цикл или условный цикл?

Список деталей Семёна и шаги сборки:

Пожалуйста, обратитесь к Уроку 1.



Список деталей

2



x2

20



x8

32



x4

40



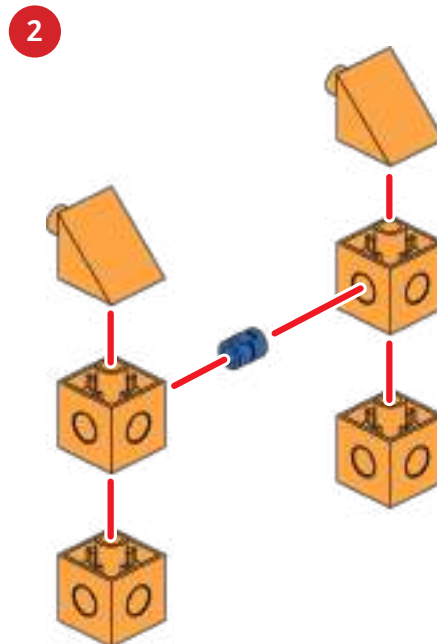
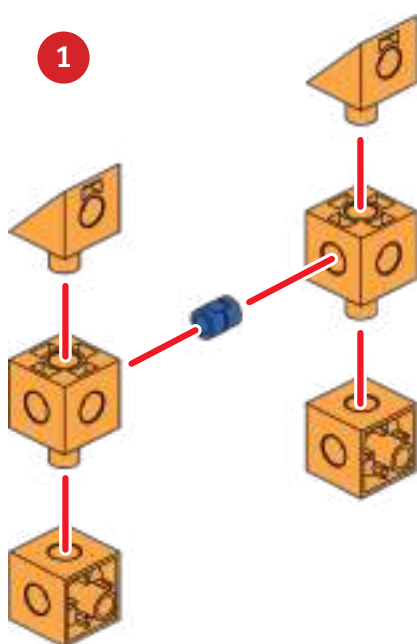
x2

50



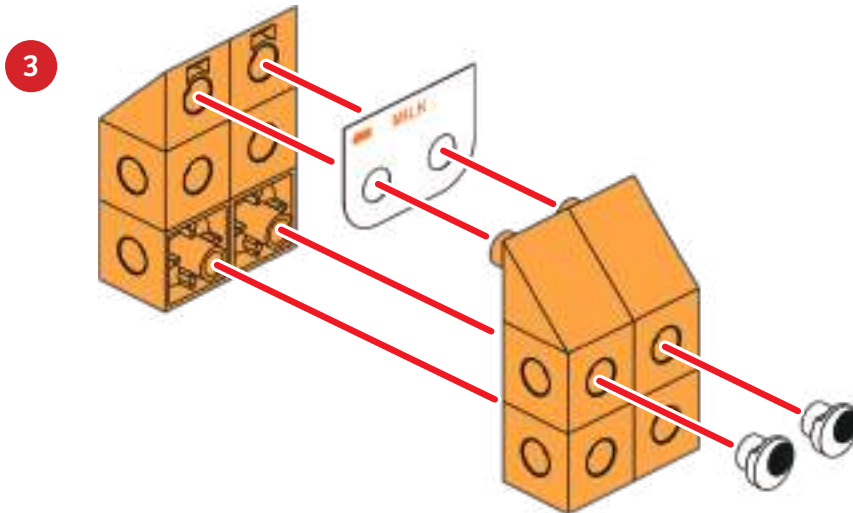
x1

МАША-МОЛОЧНИЦА



3

Спросите Машу – молочницу, как приготовить яблочный пирог!

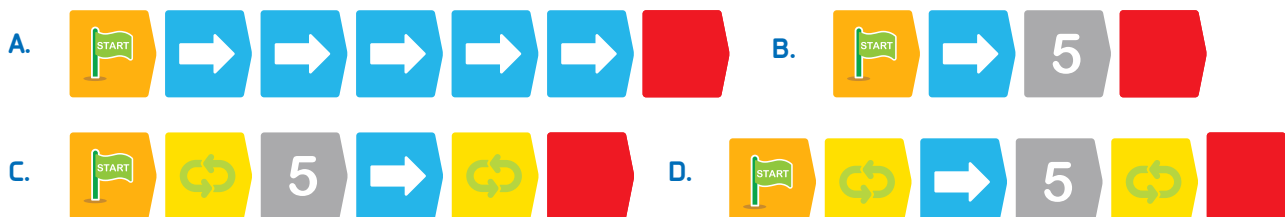


Сделано



Умное руководство
Веб-сервис

[код]



Что происходит?

- A. Пять карточек «Двигаться вперед» заставляют Семёна двигаться вперед на пять маршрутных карт к Маше.
- B. Кодовая карточка с номером 5 выполняет карточку «Двигаться вперед» пять раз, перемещая Семёна к Маше-молочнице.
- C. Зеленый простой цикл выполняется пять раз из-за числовой кодовой карточки с номером 5. Цикл определяется как одна карточка «Двигаться вперед», поэтому Семён перемещается пять раз и направляется к Маше.
- D. Зеленый простой цикл выполняется только один раз, но числовая карточка с номером 5 повторяет команду «Двигаться вперед» пять раз.

Попробуйте это: установите внутренний цикл в команде цикла этого урока. Позвольте Семёну идти вперед и назад всего 4 раза. (Подсказка: вы можете использовать функцию вложенного цикла.)



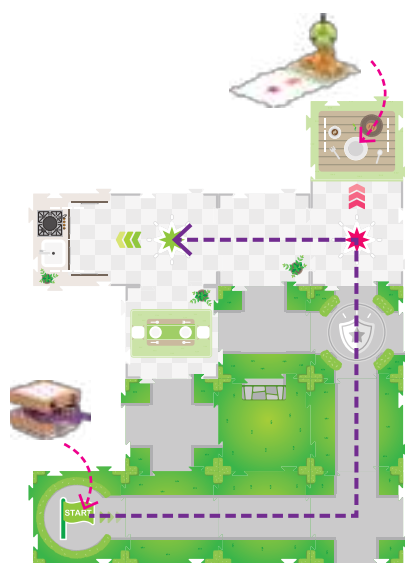
.....

.....



Видео работы модели





Семён хочет сделать яблочный пирог. Он идёт через город за яблоком, раскрывает свои объятия, берет яблоко со стола несет до кухни и отпускает его там. Начни Семёна с объятиями, и поместите яблоко на пьедестал.

Далее попробуйте включить некоторые звуковые эффекты. Пусть Семён перейдет от красной функции базовой карты, захватит яблоко и скажет «Ах!», а затем перейдет к зеленой функции базовой карточки, положит яблоко, а потом скажи "А?"

Концепция кодирования

Функция - это набор шагов, которые можно использовать снова и снова в большой программе. Функция написана один раз и ей дают имя или ярлык. Тогда функция может быть использована в программе, когда это требуется, устраняя при этом необходимость написания кода функции более одного раза. Это помогает сделать код коротким и понятным. Много функций уже встроены в языки программирования и компьютеры. Функции также иногда называют подпрограммами.

Мозговой штурм














Какие карточки с инструкциями можно использовать для обучения Семёна, чтобы открыть и закрыть руки?

Список деталей Семёна и шаги сборки:

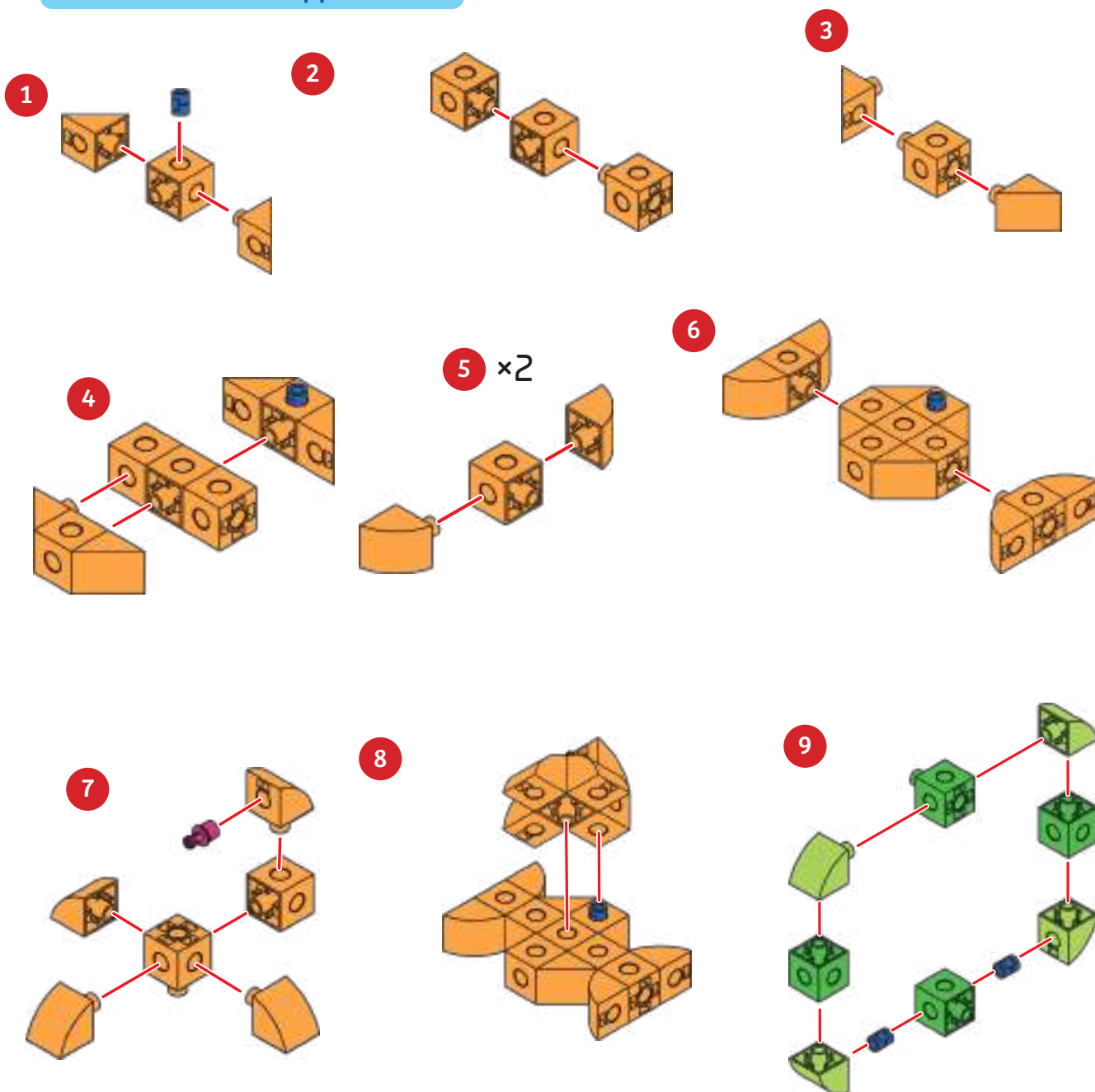
Пожалуйста, обратитесь к Уроку 1.



Список деталей

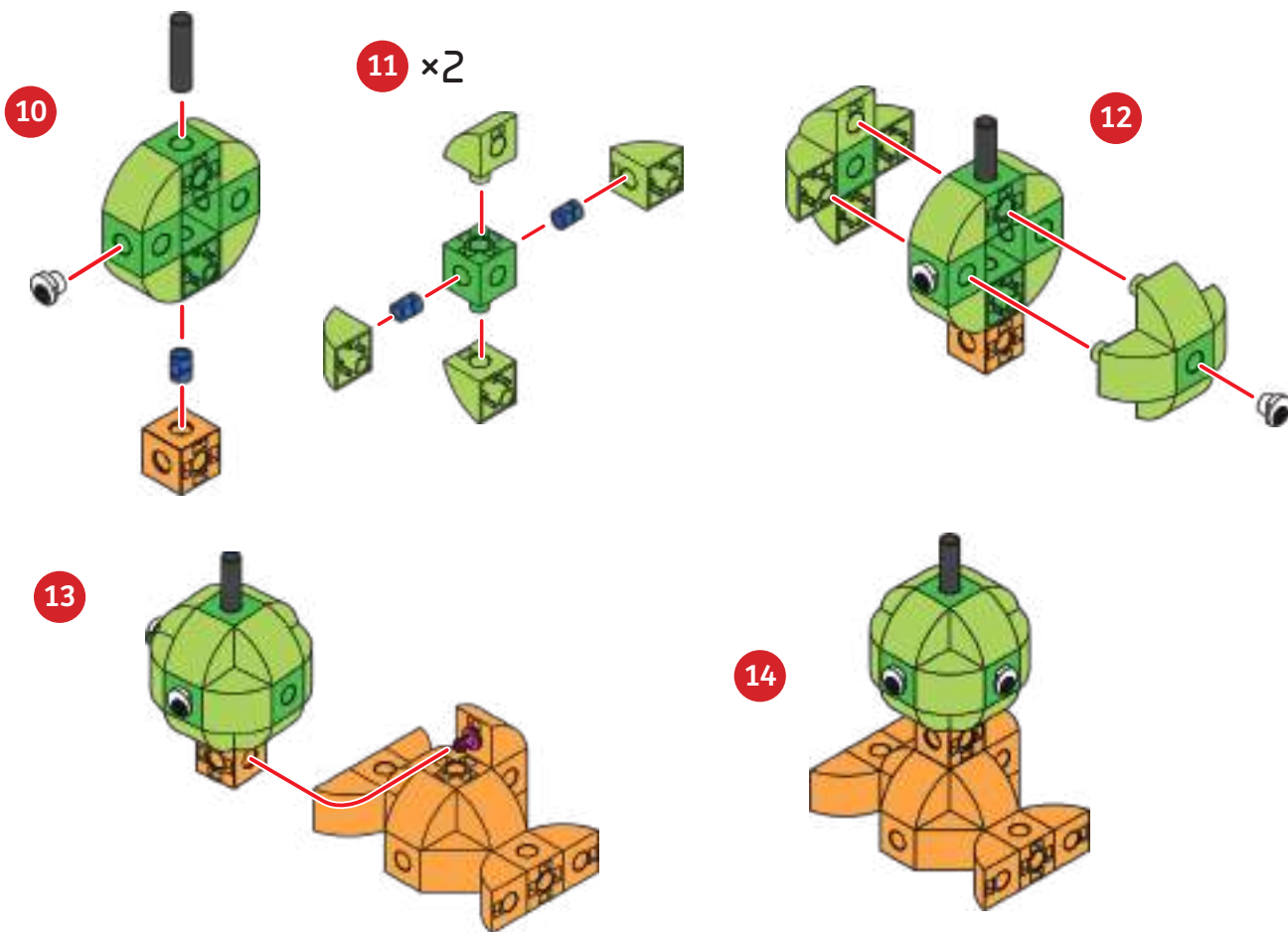
 x1	 x8	 x6	 x9	 x1	 x12	 x8	 x4	 x4	 x1
 x2	 x1	 x1							

ЯБЛОКО НА ПЬЕДЕСТАЛЕ

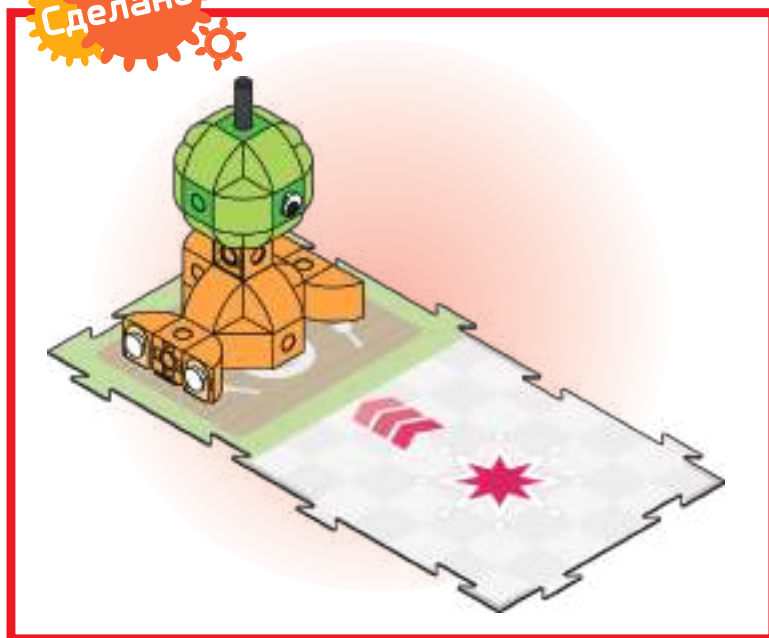


4

Делаем яблочный пирог!



Сделано

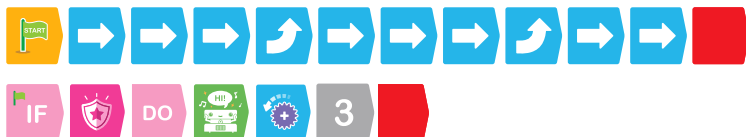


Теперь прикрепите машину к этой базовой карте с помощью пластиковых полосок.



Умное руководство
Веб-сервис

[код]



Пожалуйста, обратитесь к Уроку 7 для использования условных элементов.
Почему здесь есть условный элемент? Попробуйте: что произойдет, если вы удалите условный элемент?
Вы откроете для себя важность «последовательностей» в программе!

Что происходит?

Основная программа перемещает Семёна по улицам к карте событий, затем к красной функциональной звезде и, наконец, к зеленой функциональной звезде. Условный элемент гласит, что если Семён сканирует карту События 1, то Семён должен открыть свои руки и сказать "Привет!"

Красная функция приближает руки Семёна к яблоку и воспроизводится звук "Ах". Красная функция использует карту, чтобы повернуть выходной механизм направо, это может заставить Семёна закрыть руки. В сочетании с числовой кодовой картой, у Семёна достаточно времени, чтобы закрыть руки и, схватив воспроизвести звук «Ах».

Зеленая функция заставляет Семёна отпустить яблоко на кухне и воспроизвести звук "А?". Это может занять несколько попыток, прежде чем заставить его работать правильно. Это нормально в кодировании и робототехнике.

Попробуйте, измените программу, чтобы Семён воспроизвел звуковой эффект «Аплодисменты» дважды после того, как он оставит зеленое яблоко на кухне.



Видео работы модели



1



Собрал модель

2



Участвовал в эксперименте

3



Выполнил творческое задание

5

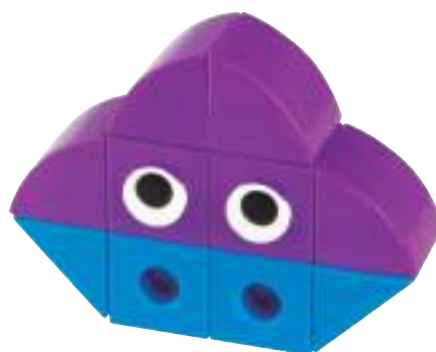
Конкурсная работа 1

Соберите маршрутную карту и поместите на нее Пудинг Пашу в любое место. Семён начнет движение с карты старта и пройдет мимо Паши, чтобы поздороваться. Когда Семён увидит зеленое яблоко на столе, он раскроет руки, схватит зеленое яблоко и принесет его на кухню, чтобы сделать вкусный яблочный пирог.

(Примечание: вам нужно будет добавить новую функцию, чтобы Семён сказал «Привет!»)



1. Семён



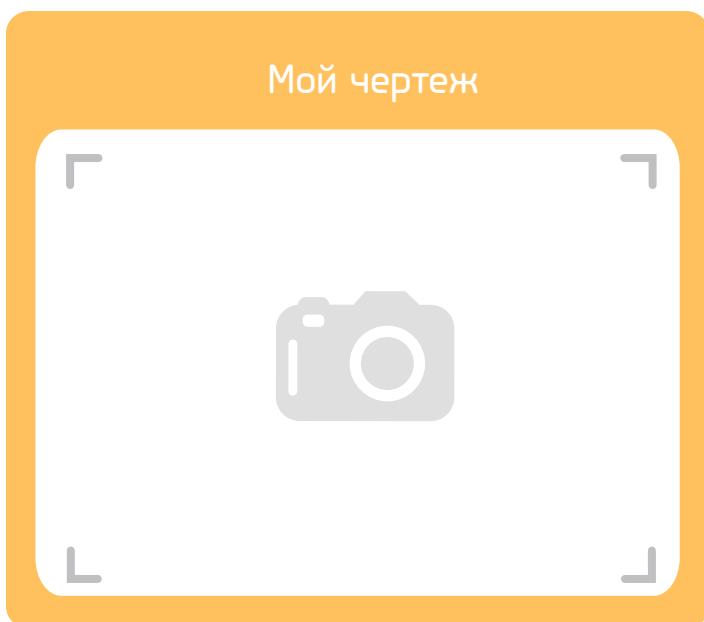
2. Пудинг Паша



3. Яблоко на пьедестале

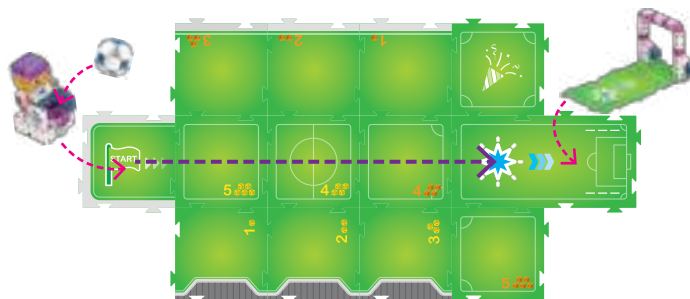


4. Маршрут





Теперь наш робот – футболист! Его зовут Рома. Во время тренировки по футболу Рома пытается бежать прямо по всей длине поля и пинать мяч в цель. Вы можете запрограммировать Рому, чтобы он мог перемещаться к цели и запустить мяч? В начале загрузите мяч для запуска в механизм Ромы.

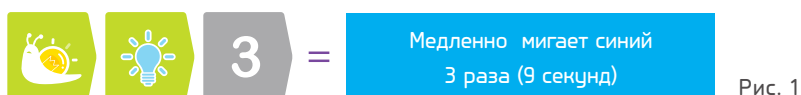


Концепция кодирования

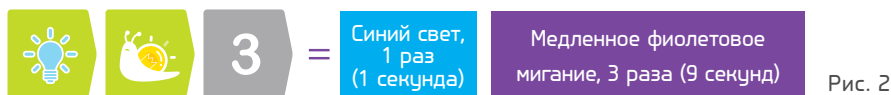
В дополнение к функции выходного зубчатого колеса, мы по-прежнему исследуем и другие функции, которые могут быть использованы для запуска различных звуковых и световых эффектов. Последовательность кодовых карточек означает, что вы можете создавать более сложные результаты.

ПРИМЕР:

Если вы сложите вместе карточки следующим образом (рис. 1), цвет светового эффекта будет синим вместо фиолетового по умолчанию, и он будет работать три раза или около девяти секунд в этом случае.



Если вы разместите карточки следующим образом (рис. 2), световой эффект не будет в сочетании со светлым цветом. Синий свет будет светить в течение одной секунды и затем световой эффект будет запущен три раза в фиолетовом цвете по умолчанию.



Мозговой штурм

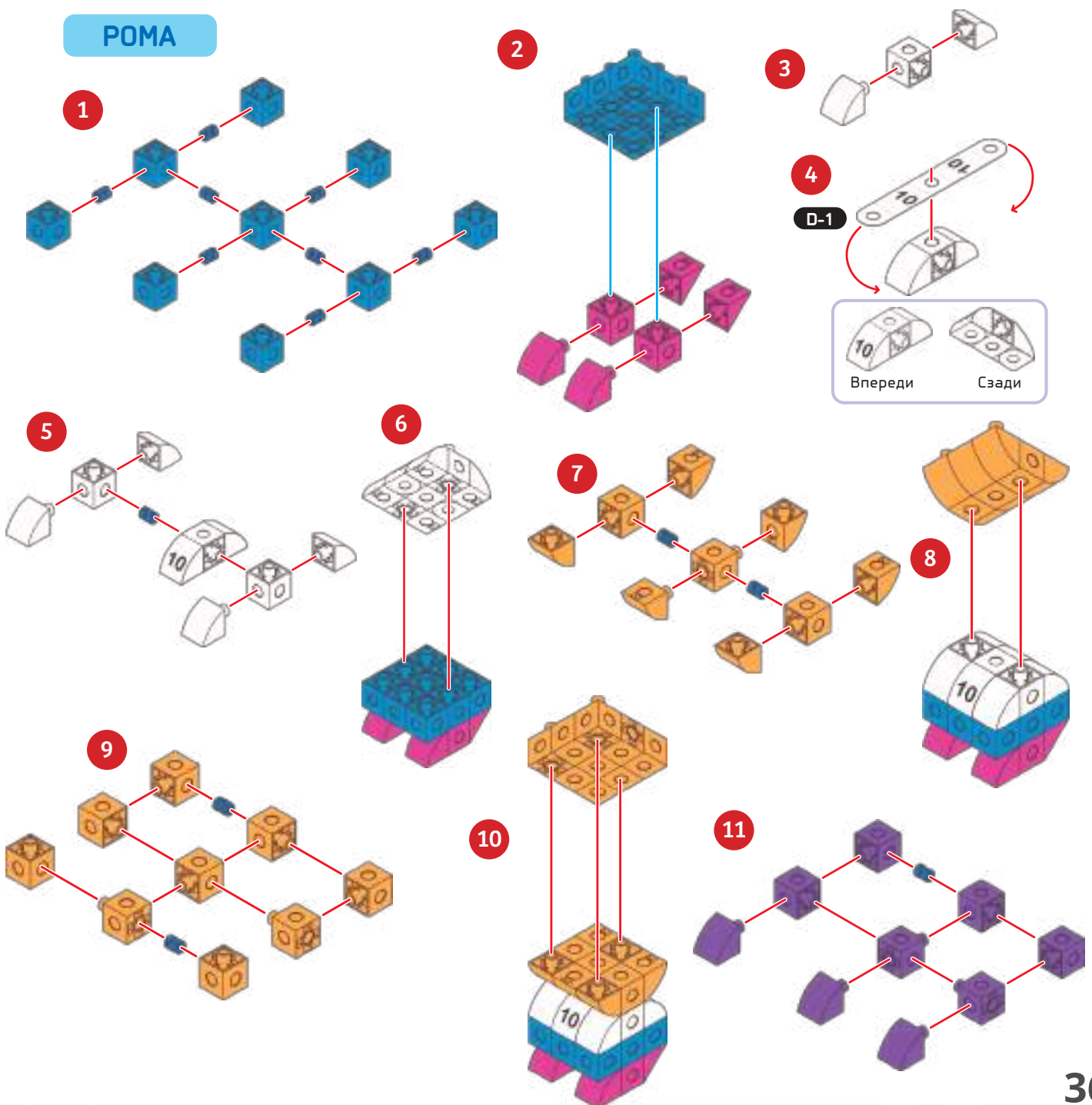


Какой в результате световой эффект получится от этой последовательности кодовых карточек?

Список деталей

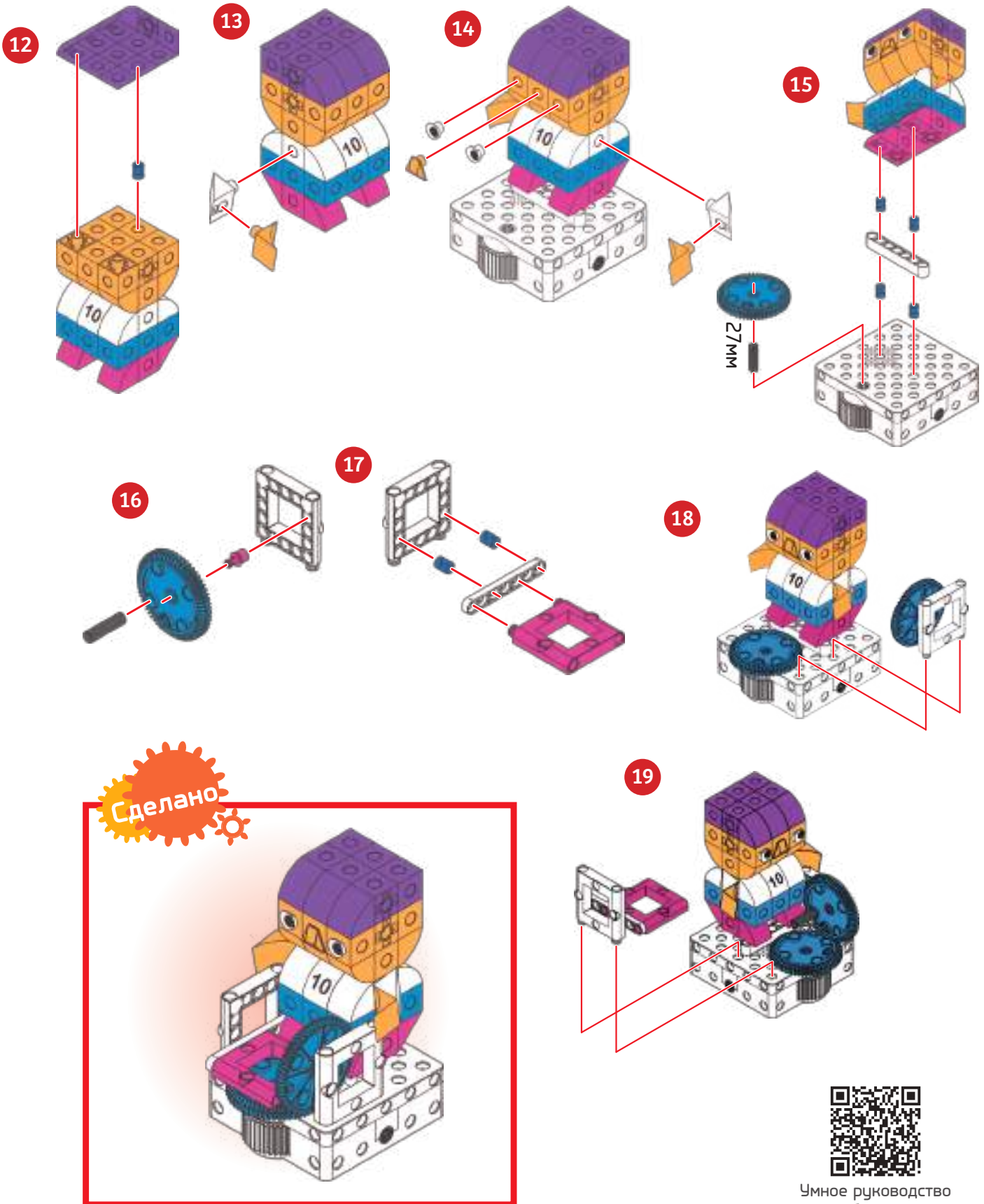
1	2	6	7	11	12	13	14	15	16	17	19	
x1	x21	x2	x1	x1	x1	x1	x2	x3	x9	x2	x6	
20	23	25	27	28	29	31	35	37	39	40	50	56
x12	x6	x2	x3	x6	x2	x2	x2	x1	x1	x2	x1	x1

РОМА



6

Прямой выстрел

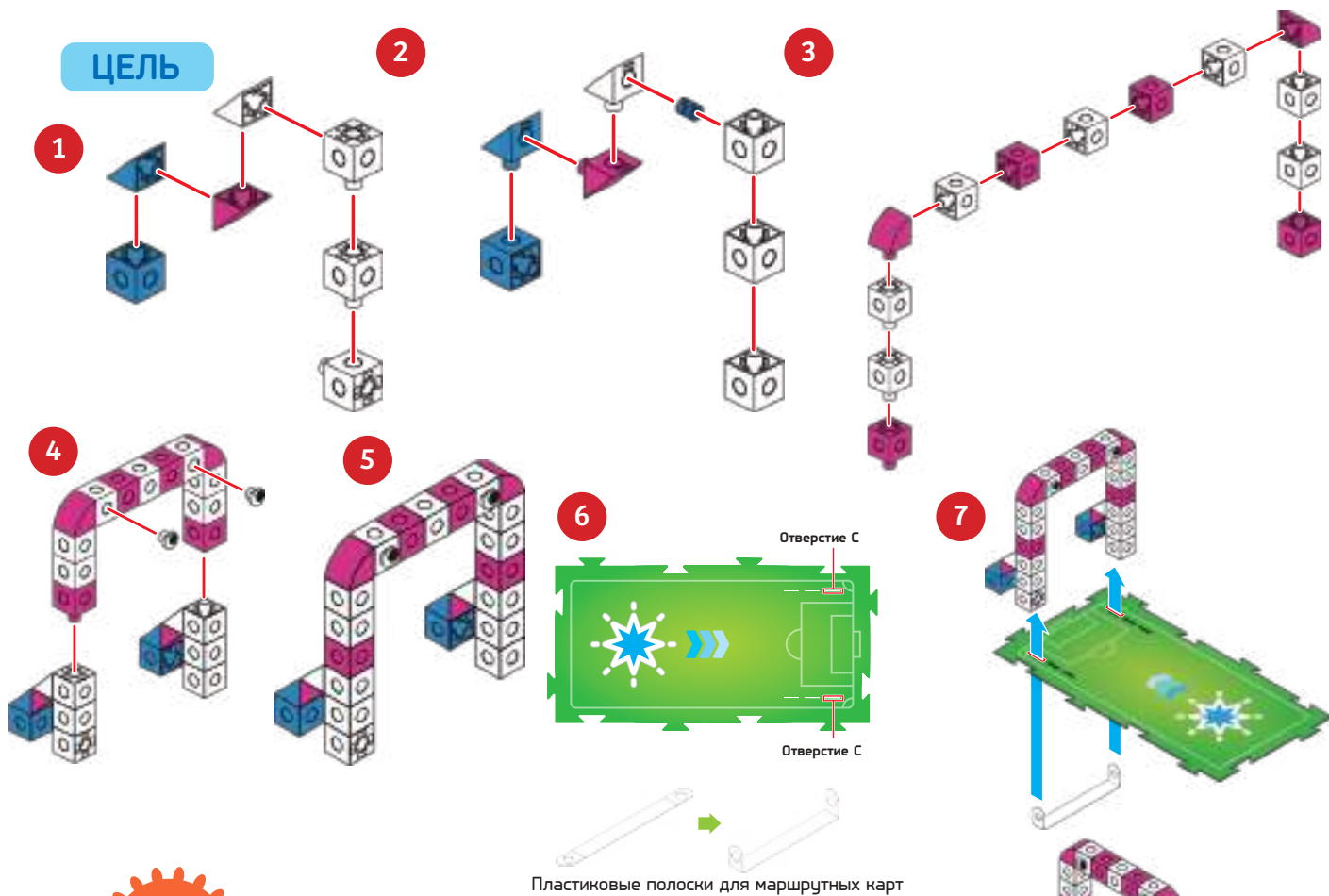


Умное руководство
Веб-сервис

Список деталей

2	15	16	17	25	29	30	31	36	40	51
x1	x13	x2	x4	x2	x2	x2	x2	x2	x2	x1
53										
x1										

ЦЕЛЬ



Сделано



8

Прикрепить короткой фиксирующей кнопкой



9



Умное руководство
Веб-сервис

6 Прямой выстрел

Список деталей

2



x2

16



x6

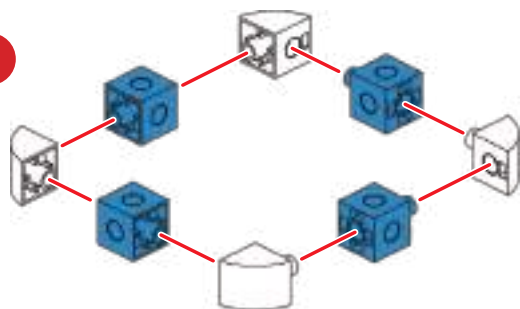
23



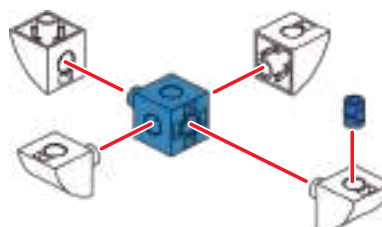
x12

ФУТБОЛЬНЫЙ МЯЧ

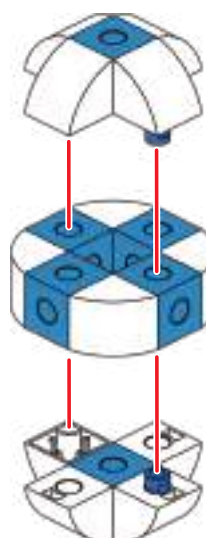
1



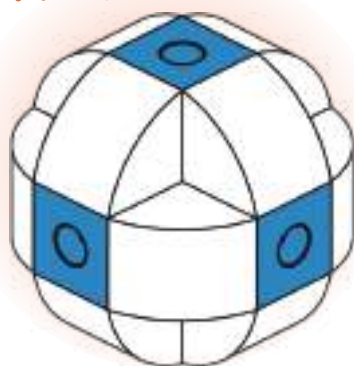
2 x2



3

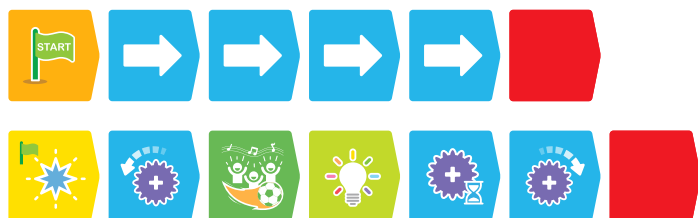


Сделано



Умное руководство
Веб-сервис

[код]



Что происходит?

Основная программа перемещает Рома по маршруту к базовой карте и соответственно к цели. Синяя функция заставляет выходную передачу вращаться влево и вправо для запуска мяч, а затем играть радостный звук с огнями.

Попробуйте это: соберите три длинных столбца с кубиками и замените первоначальную цель. Пусть Рома поиграет в боулинг! Посмотри, сколько колонок (кеглей) он может сбить.



.....

.....



Видео работы модели



1



Собрал модель

2

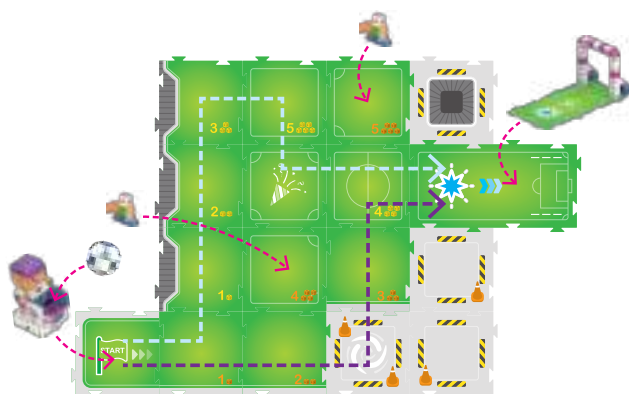


Участвовал в эксперименте

3



Выполнил творческое задание



Во время футбольного матча Рома перемещает мяч с одной стороны поля к другой, избегая противников.

Попробуйте это: напишите программу, чтобы Рома смог избежать противников, приостановив движение, и успешно направится к цели. Когда Рома прибывает в синей функции

базовой маршрутной карты, воспроизводится веселый звук и световой эффект. Поместите двух противников на карту, как показано на рисунке.

Концепция кодирования

Компьютеры и роботы часто должны принимать решения, чтобы правильно выполнять задачи или программы. Программисты используют условные операторы (или условные утверждения), чтобы дать компьютерам и роботам умение принимать решения.

Условный оператор - это часть программы, которая делает сообщение компьютеру или роботу для выполнения различных инструкций в зависимости от того, является ли указанное условие или набор условий **правда или ложь**. Условные утверждения часто просто называют **условные**.

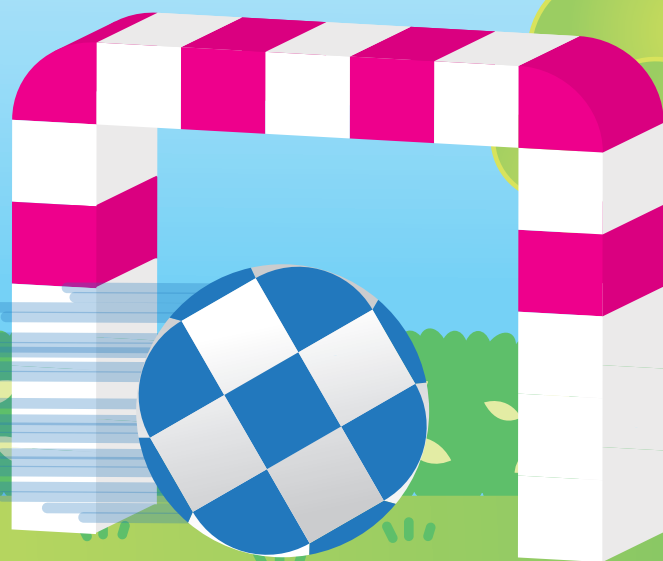
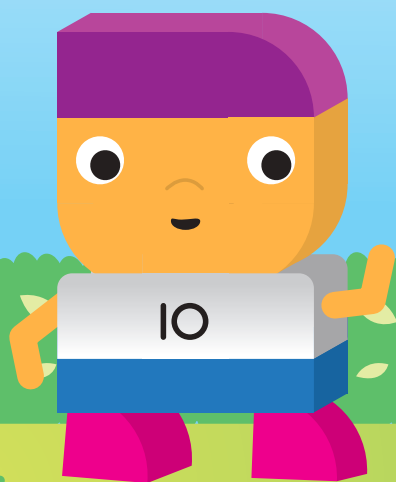
Условные операторы также известны как операторы **if-then**: Если условие истинно, тогда программа выполнит инструкции. Если условие не соответствует действительности, тогда программа не будет запускать инструкции.

Мозговой штурм

В футбольных играх есть несколько навыков для игроков защитников. Составьте список некоторых навыков, которые вы знаете.

Список деталей, этапы сборки Ромы, цели и футбольного мяча:

Пожалуйста, обратитесь к уроку 6.



7

Игра Ромы - решение дня

Список деталей

16



x2

18



x2

22



x2

24



x4

28



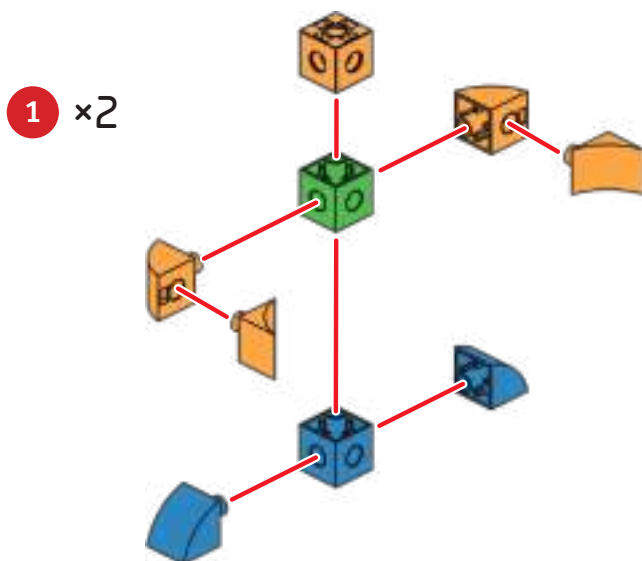
x4

35

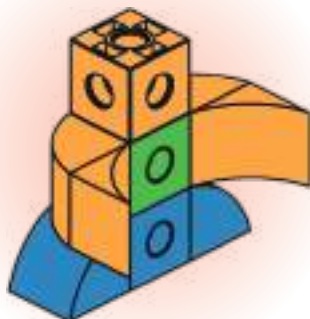


x4

ФУТБОЛЬНЫЕ ИГРОКИ

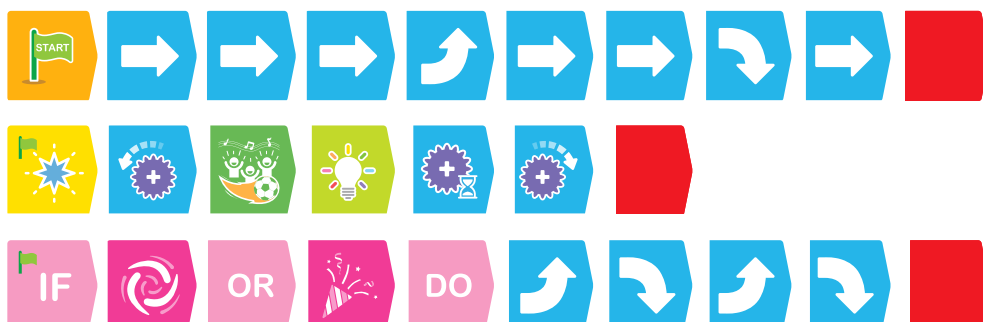


Сделано



Умное руководство
Веб-сервис

[код]



Что происходит?

Основная программа следует за фиолетовой линией к цели. Синяя функция работает так же, как и в предыдущем уроке, чтобы выпустить мяч. Теперь в коде есть условие (условный оператор). Это говорит следующее. Если робот сканирует карточки Элемент условий 3 или Элемент условий 2, затем робот должен повернуть налево и направо чередуясь два раза, как бы в искусном маневре, тогда основная программа продолжается. Противники размещаются рядом с картами событий.

Попробуйте это: найдите другой маршрут, чтобы добраться до цели. Пусть Рома избегает противников по-другому, но добирается до той же цели.



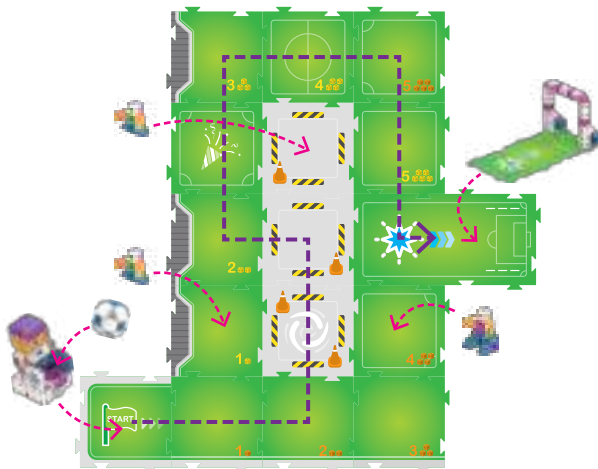
.....

.....



Видео работы модели





Теперь есть три противника на поле. Рома должен маневрировать мимо них, чтобы добраться до цели.

Напишите программу для Ромы, где он должен отсканировать обе карты событий и выполнять действия (поворот налево и направо), избегая противников. Путь Ромы один и тот же, от начала до цели.

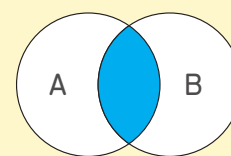
Когда Рома прибывает в синей функции базовой маршрутной карты, воспроизводится веселый звук и световой эффект. Прежде чем начать, разместите трех противников в месте, указанном на маршрутной карте.

Концепция кодирования

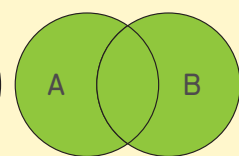
Программисты также могут комбинировать условия в разных путях. Например, программе может потребоваться два условия быть верным до запуска набора инструкций. Это называется И (And) операция, потому что и первое условие и второе условие должно быть истинным, чтобы все утверждение было истинным. Другим примером является операция Или (Or). В этом случае либо первое условие, либо второе должно быть правдой, чтобы все утверждение было правдой. **Операция** — конструкция в языках программирования, аналогичная по записи математическим операциям, то есть специальный способ записи некоторых действий.

Мозговой штурм

Взгляните на следующие две фигуры. Которая представляет функцию And? Которая представляет функцию Or?



Фигура 1



Фигура 2

Список деталей, этапы сборки Ромы, цели, футбольного мяча и игроков:

Пожалуйста, обратитесь к урокам 6 и 7.



8 Поле условий

Список деталей

16



x1

17



x1

19



x2

21



x1

24



x4

28



x2

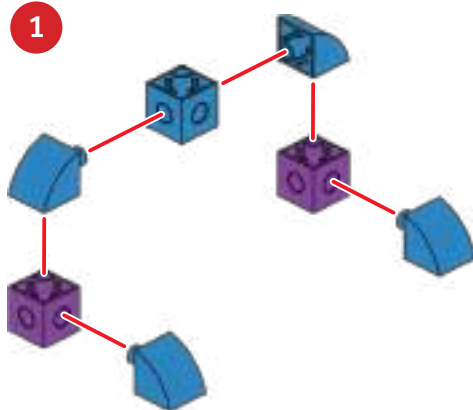
32



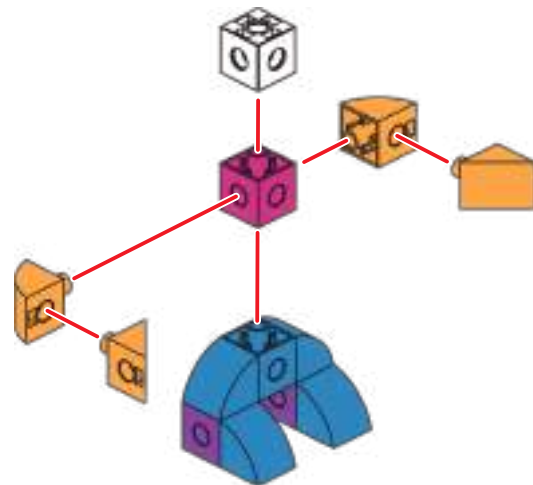
x2

ФУТБОЛЬНЫЙ ИГРОК

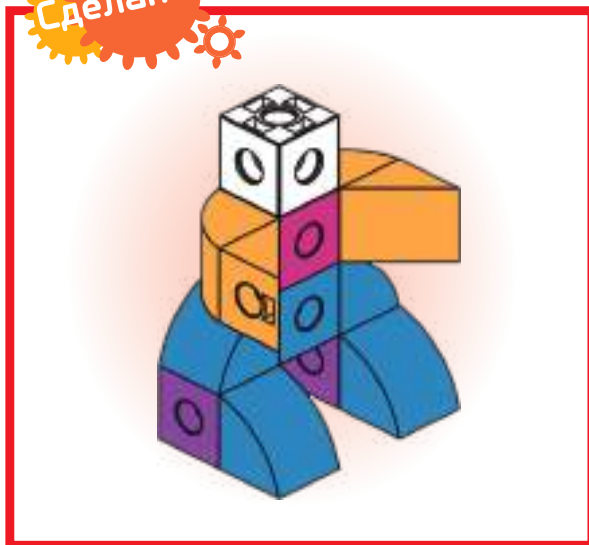
1



2

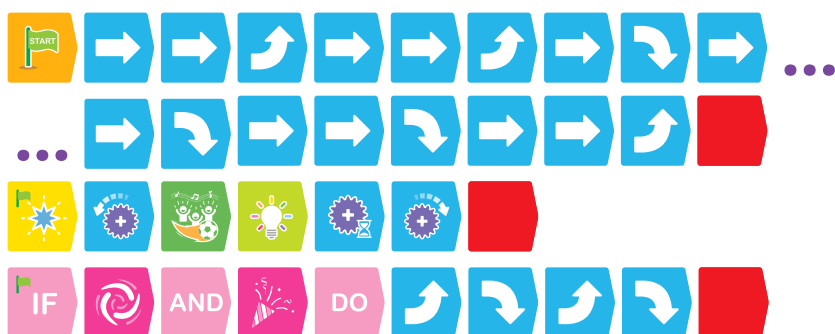


Сделано



Умное руководство
Веб-сервис

[код]



Что происходит?

Основная программа перемещает Рому от начала к цели, избегая противников. Синяя функция работает так же, как и в предыдущем уроке. Но на этот раз условное утверждение содержит карточку И (And), требующую, чтобы робот сканировал обе карты событий до того, как утверждение окажется верным.

Попробуйте это: перепроектируйте условное утверждение: поменяйте карточку И (And) с карточкой ИЛИ (Or). Что изменится в результате?



.....

.....



Видео работы модели



1



Собрал модель

2

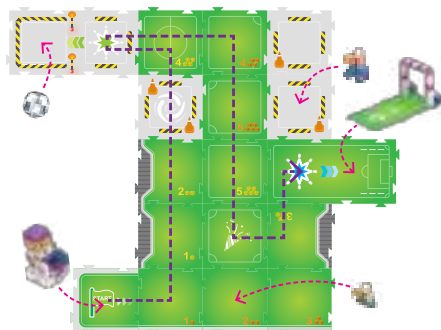


Участвовал в эксперименте

3



Выполнил творческое задание



Это день большой игры. Рома должен бежать к месту на поле, где он может забрать мяч. Когда он туда добирается, он делает паузу, и вы можете поставить мяч на пусковой установке Ромы. Затем Рома бежит к цели, стреляет, и ...!

Добавьте команду, чтобы заставить Рому ждать его игроков по команде, чтобы те передали ему мяч. Соблюдайте маршрутную карту и подумайте, как вы можете написать эту программу.

Концепция кодирования

В программировании мы используем события, функции и время последовательности, чтобы заставить робота выполнять правильные действия. Например, в этом уроке мы добавили прохождение и ловлю мяча. Программист логически упорядочит эти добавленные функции до того, как Рома проберется сквозь защиту противника и перед тем, как он будет стрелять по воротам. Подумайте об этом: если вы поставите функцию (условный элемент) прохода и лови после того, как Рома пройдет через защиту противника и ударит по воротам, что будет?

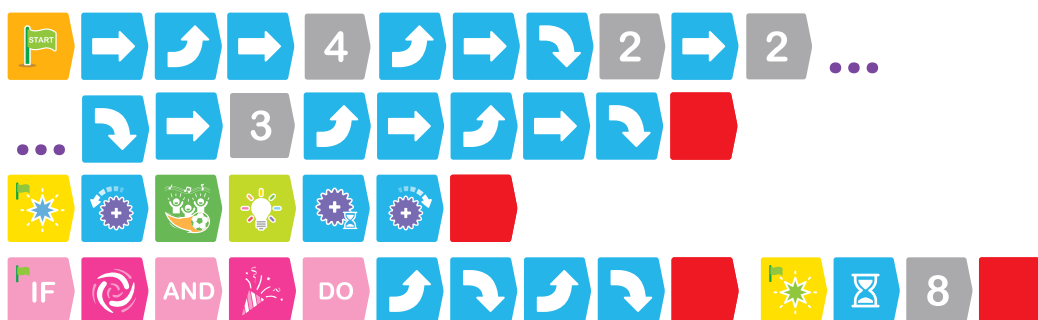
Мозговой штурм

Какие примеры временных последовательностей вы используете в повседневной жизни?

Список деталей, этапы сборки Ромы, цели, футбольного мяча и игроков:
Пожалуйста, обратитесь к урокам с 6 по 8.



[код]



Что происходит?

Напишите основную программу для перемещения Ромы на зеленую функциональную базовую карту. Рома делает паузу на базовой карте на 8 секунд, чтобы поймать мяч. Условное утверждение содержит карточку И (And), требующую от робота отсканировать обе карточки событий, прежде чем утверждение станет правдой. В заключении, Рома добирается до синей функциональной базовой карты и бьет по мячу.

Попробуйте это: смените программу с помощью условного утверждения, и заставьте Рому выполнять разные действия, избегая противников.



.....

.....



Видео работы модели



10

Конкурсная работа 2

Сделайте прямоугольную карту, поместите на нее двух противников и цель. Напишите программу, которая скажет футболисту забрать мяч с угла и пройти мимо любых противников путь. Тогда берите мяч и стреляйте в ворота!



1. Рома



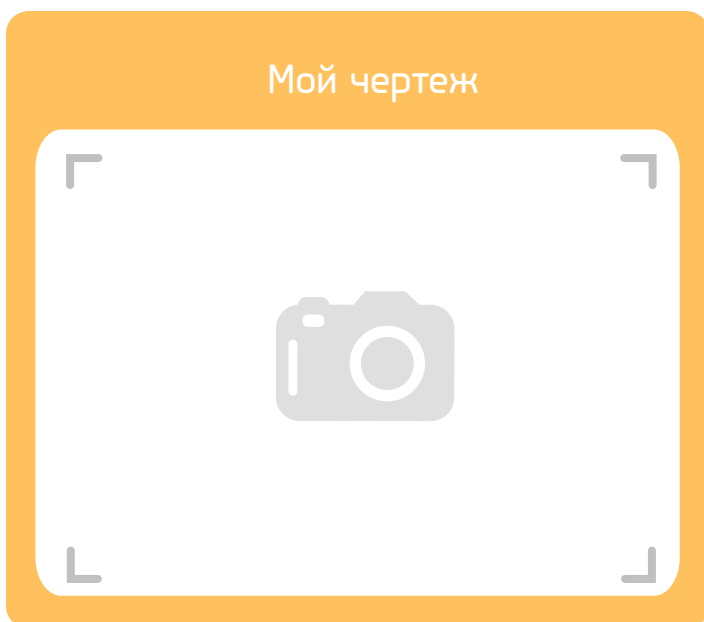
2. Цель

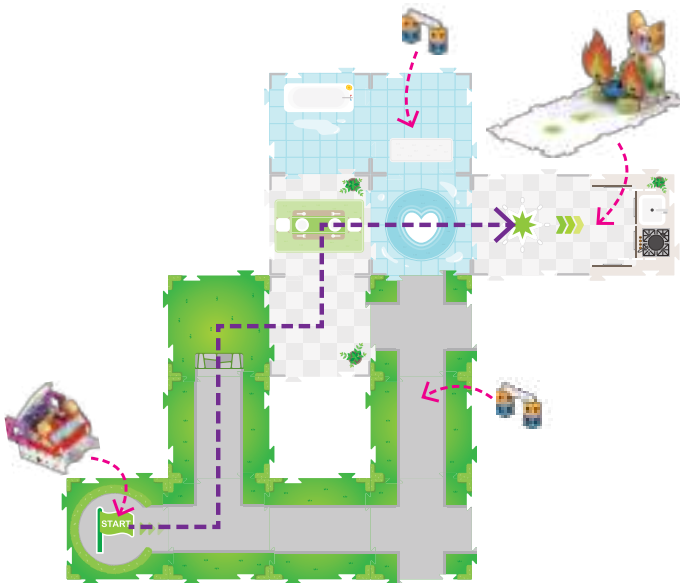


3. Футбольный мяч



4. Футбольные игроки





Теперь наш робот превращается в пожарную машину! Эта роботизированная пожарная машина может двигаться через город к костру и погасить огонь. Пожарная машина должен перемещаться от начала к кухне, избегая барьеров, и в итоге потушить огонь.

Подумай об этом. Как ты можешь организовать маршрут пожарной машины, избегая барьеров через дорогу?

Напишите программу, чтобы сигнальный маячок пожарной

машины мигал красным и синим светом по дороге на кухню. Когда пожарная машина прибывает на кухню, заставьте воспроизвести звук сирены и мигать красным и синим светом.

Концепция кодирования

Условие if-else. Эти условные утверждения имеют две ситуации. В том случае, если условие истинно, то будет запущен один набор инструкций. Если условие не верно, выполняется другой набор инструкций.

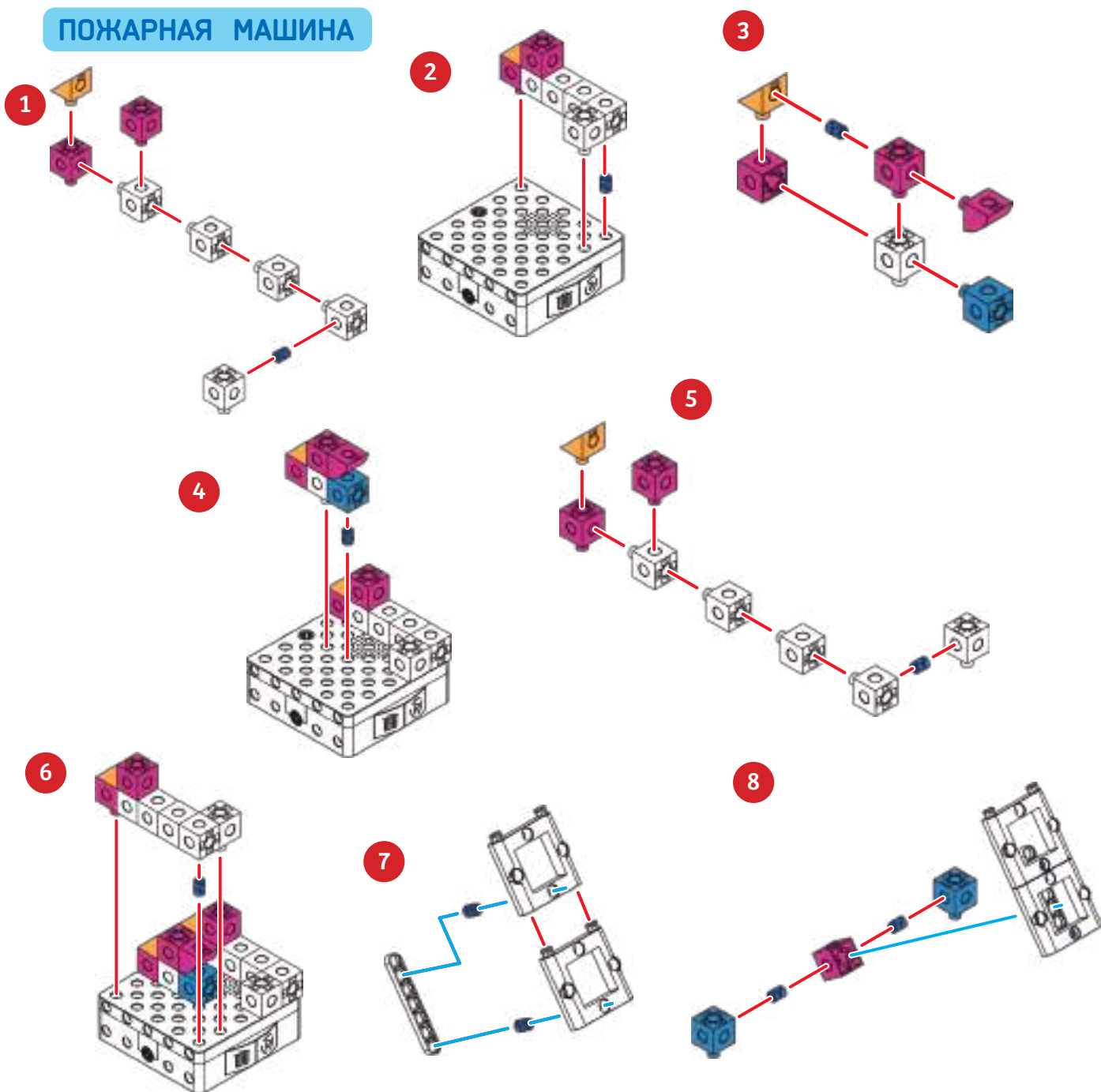
Мозговой штурм

Люди часто используют условные высказывания if-else (если – еще/ кроме этого) в разговоре. Попробуйте сделать предложение с «если А, то В; если не А (еще), то С». (Например: если завтра будет хороший день, мы пойдем в зоопарк; в противном случае мы останемся дома, чтобы поиграть со строительными блоками.)

Список деталей

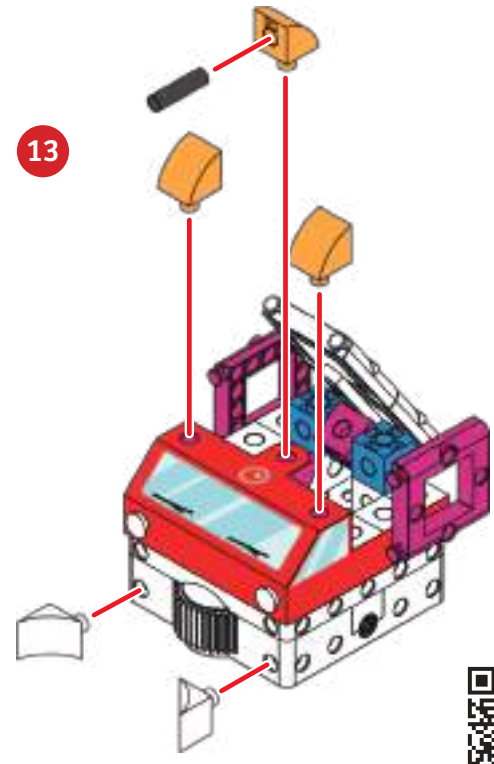
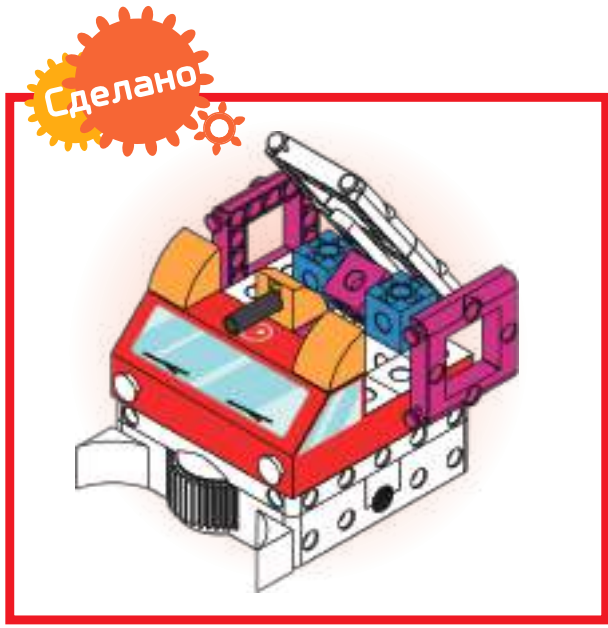
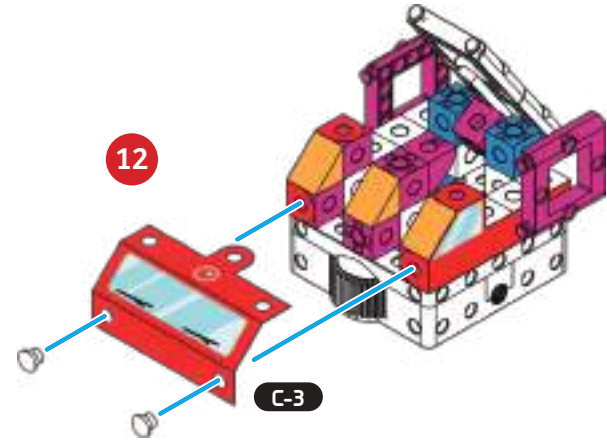
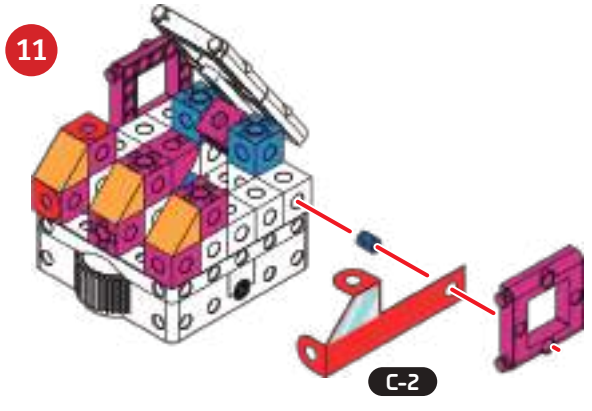
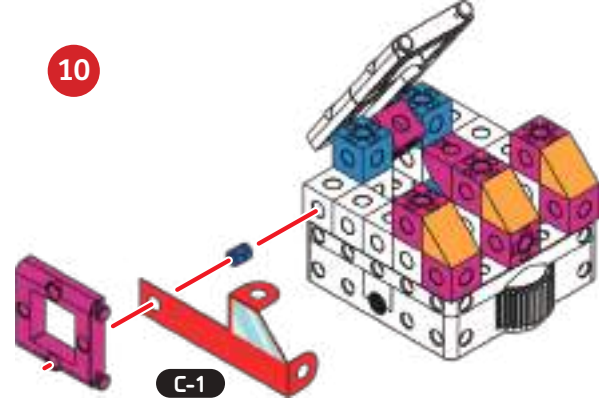
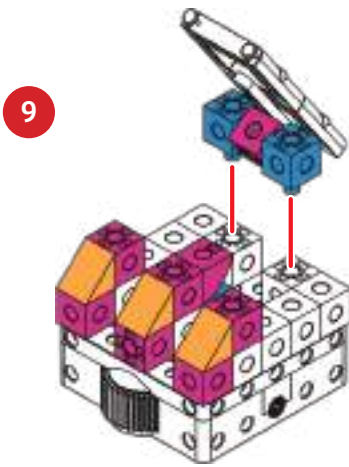
2  x12	12  x1	13  x2	14  x2	15  x11	16  x3	17  x7	25  x1	28  x3	32  x3
33  x2	36  x2	37  x1	50  x1	56  x1					

ПОЖАРНАЯ МАШИНА



11

Спаси кошку от пожара на кухне

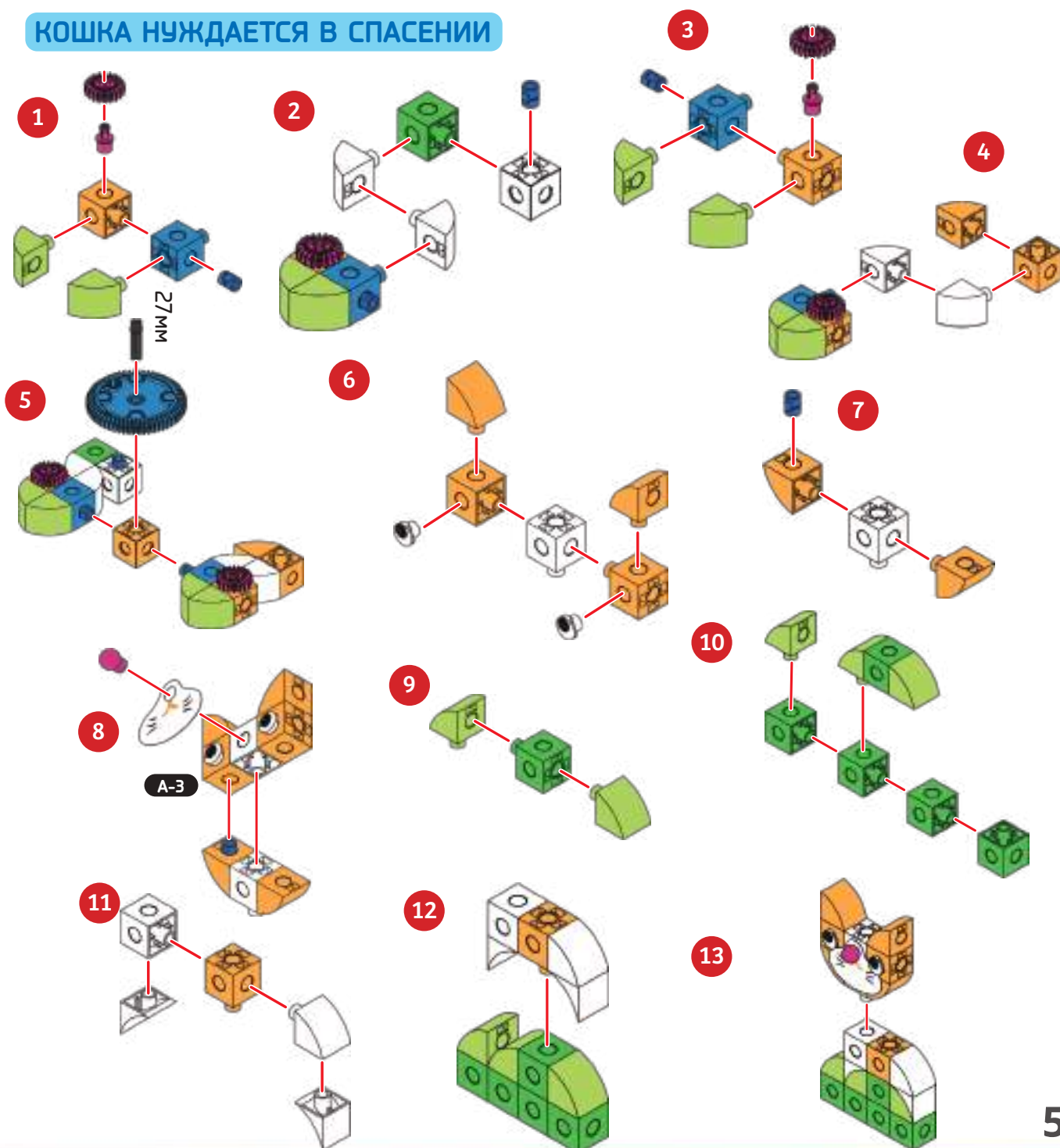


Умное руководство
Веб-сервис

Список деталей

1	2	5	6	7	15	16	18	20	21	22	23
x2	x4	x2	x1	x1	x3	x4	x6	x6	x1	x1	x5
26	28	33	36	38	40	50	51	53			
x7	x5	x2	x4	x1	x4	x1	x1	x1			

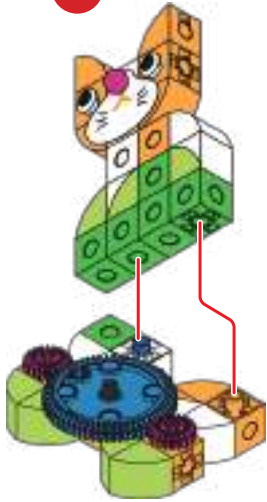
КОШКА НУЖДАЕТСЯ В СПАСЕНИИ



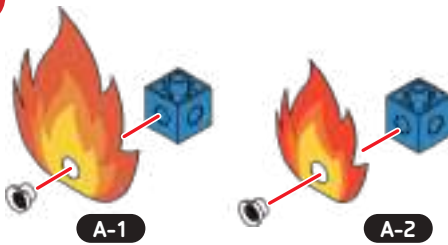
11

Спаси кошку от пожара на кухне

14



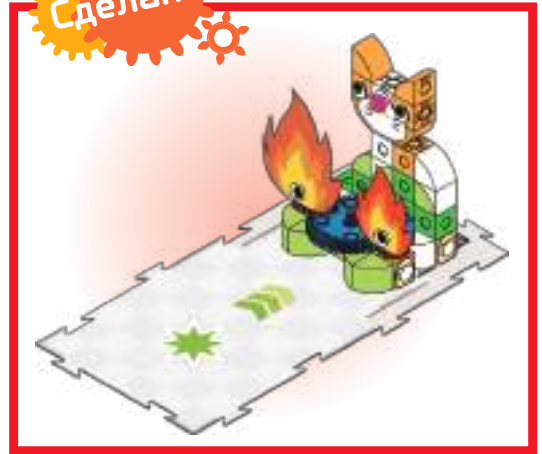
15



16



Сделано



Теперь прикрепите машину к этой базовой карте пластиковыми полосками.



Умное руководство
Веб-сервис

Список деталей

11



x2

16



x4

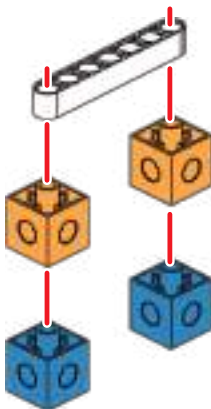
20



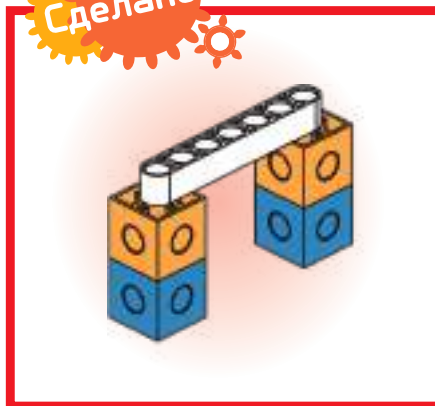
x4

БАРЬЕРЫ

1 x2



Сделано



Умное руководство
Веб-сервис

[код]



Что происходит?

Основная программа перемещает пожарную машину на базовую карту с зеленой звездой. Зеленая функция активирует выходной механизм, чтобы «потушить огонь» и воспроизводится звук пожарного шланга. Условное утверждение говорит робот должен мигать красным и синим светом, пока робот не отсканирует карту Событие (Event) 4. Затем, если робот сканирует Событие (Event) 4, он воспроизводит звук сирены и мигает красными и синими аварийными огнями.

Попробуйте это: когда пожарная машина покидает стартовую карту, она медленно мигает красным и синим светом. Как мы можем настроить программу, если мы хотим, чтобы медленные вспышки были заменены вспышками средней скорости?



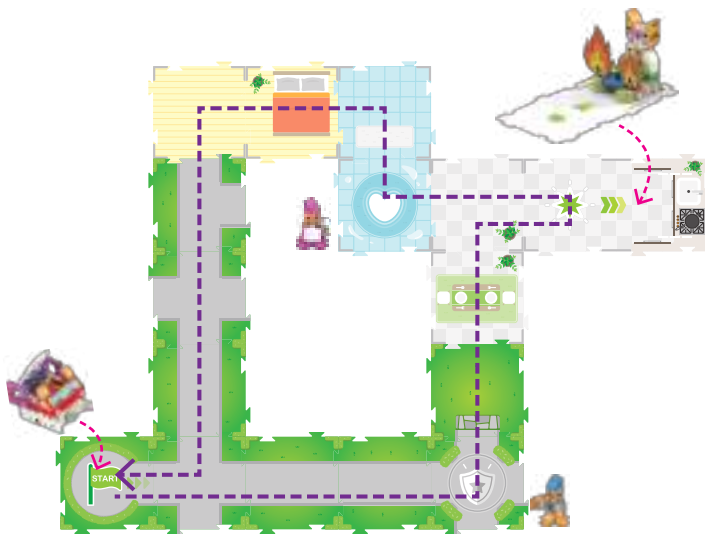
.....

.....



Видео работы модели





Нам нужно отправить пожарную машину на помощь и спасение кошки из костра. Но пожарная машина не знает где огонь, водитель должен остановиться и спрашивать людей по пути для верного направления. Когда миссия завершена и кот спасен, пожарная машина должна вернуться на пожарную станцию.

Подумайте об этом! Можете ли вы спланировать маршрут для

пожарной машины и спасти кота успешно, а затем вернуться на пожарную станцию, чтобы написать отчет? Какую программу можно использовать, чтобы остановить пожарную машину и спросить дорогу? Написать программу, чтобы заставить пожарную машину воспроизводить звук сирены и мигать красным и синим сигнальными огнями на пути к кухне.

Концепция кодирования

Операторы if-else включают операцию Or. В этом случае, либо первое условие, либо второе должно быть верно для всего утверждения, чтобы быть правдой. Если все утверждение правда, робот запускает коды после карточки **Do**. Если утверждение не соответствует действительности, робот запускает коды после карточки **Else**.

Мозговой штурм

Вы видели пожарные машины, тушившие огонь? Каковы различия между сиренами пожарных машин, полицейских машин и машины скорой помощи в месте, где ты живешь?

Пожарная машина и кот, нуждающийся в спасении, список деталей и этапы сборки:

Пожалуйста, обратитесь к Уроку 11.

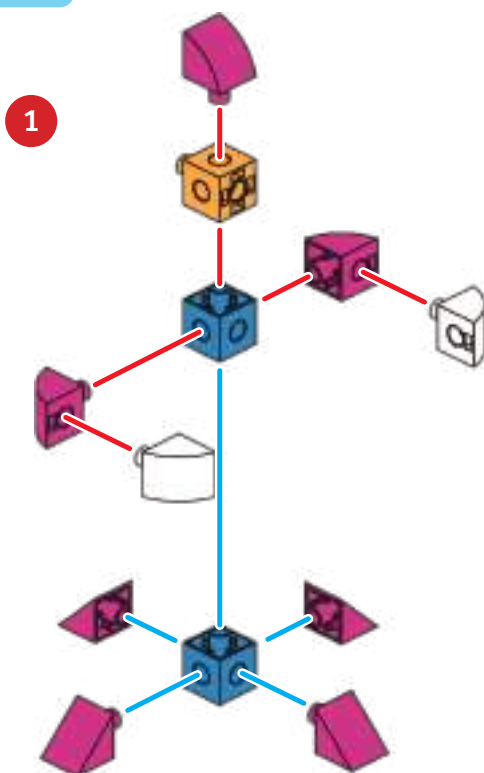


Список деталей

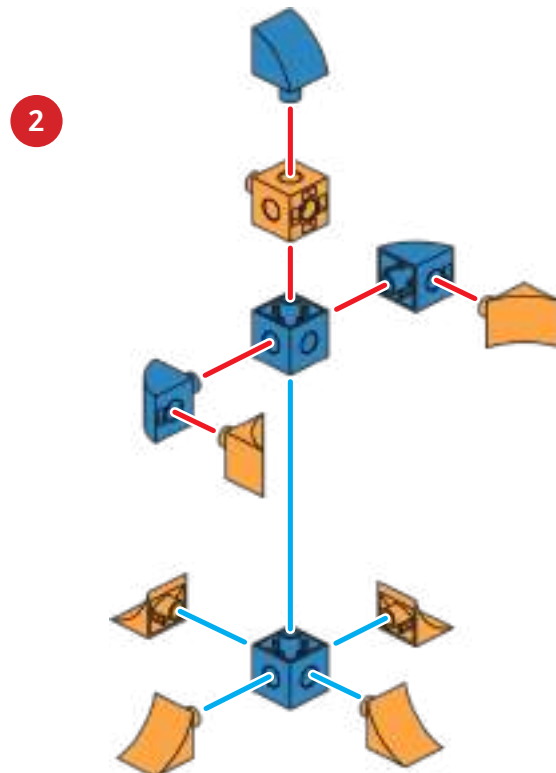
16	20	23	24	25	31	35
x4	x2	x2	x3	x3	x4	x6

ЛЮДИ

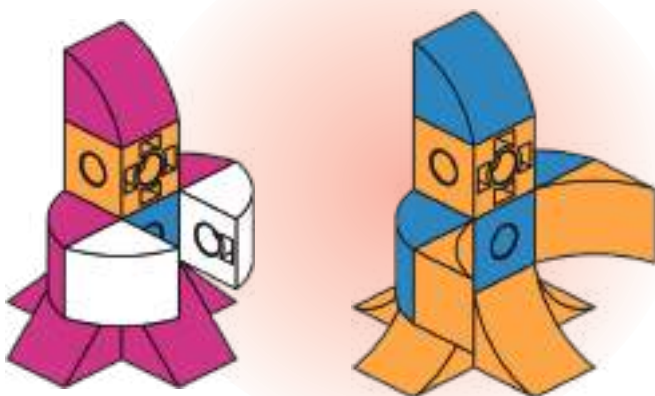
А.



В.



Сделано



А



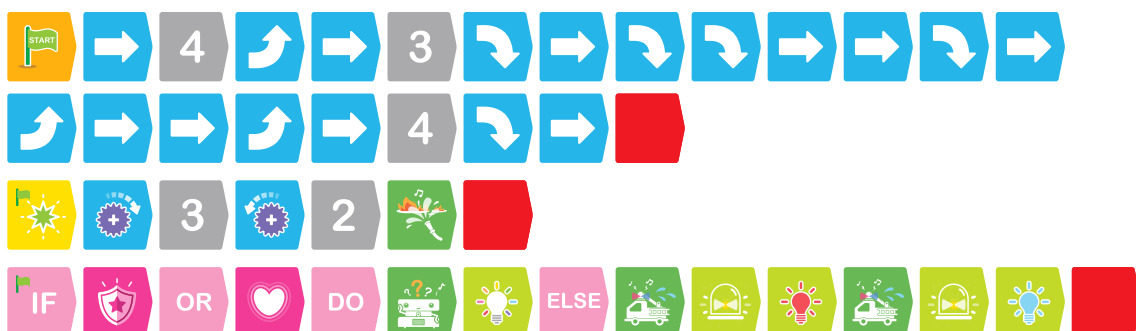
Умное руководство
Веб-сервис

В



Умное руководство
Веб-сервис

[код]



Что происходит?

Основная программа перемещает пожарную машину к базовой карте. Зеленая функция работает так же, как и в предыдущем уроке. На этот раз условное утверждение говорит роботу воспроизводить звук "А?", чтобы спросить горожан, где огонь, если робот сканирует событие 1(Event 1) или событие 4(Event 4) . В противном случае он воспроизводит звук сирены и мигает светом.

Попробуйте это: измените местоположение горожан, и перепишите программу так, чтобы завершить задачу по спасению кот и вернуться к пожарной станции.



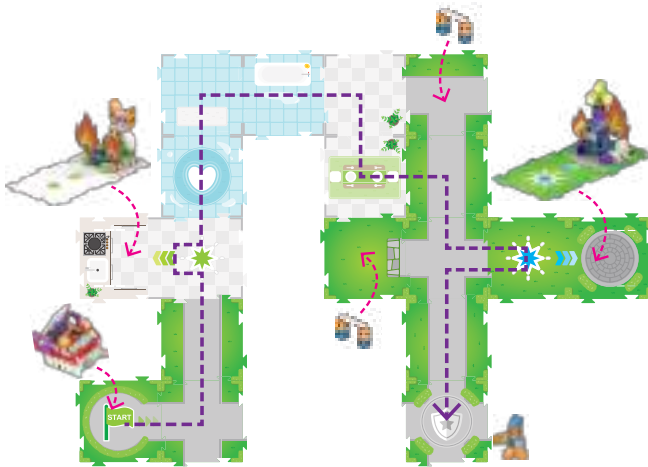
.....

.....



Видео работы модели





На этот раз пожарная машина должна сначала спасти кота, а затем птицу, избегая барьеров вдоль пути. После того, как робот тушит каждый огонь, он воспроизводит звук аплодисментов.

Напишите программу для пожарной машины так, чтобы звучала ее сирена и мигали огни на пути к пожару. Заставьте пожарную машину воспроизвести аплодисменты, а , после того как он спасёт кошку и птицу, загорится фиолетовым светом.

Концепция кодирования

Операторы if-else включают операцию And. В этом случае, первое условие и второе должны быть верными для всего утверждения, чтобы быть правдой. Если все утверждение правда, робот запускает коды после карточки **Do**. Если утверждение неправда, робот запускает коды после карточки **Else**.

Мозговой штурм

Можете ли вы выяснить, какая часть программы в этом уроке использует операцию And ?

Пожарная машина, кошка и этапы сборки:

Пожалуйста, обратитесь к Уроку 11 и 12.



Список деталей

1	2	5	6	10	15	16	19	21	22	23	24
x2	x4	x1	x1	x2	x2	x3	x8	x2	x1	x4	x1
26	27	28	36	37	40	42	50	51	53		
x5	x4	x1	x4	x1	x4	x1	x1	x2	x1		

ПТИЦА НУЖДАЕТСЯ В СПАСЕНИИ

1

2

3

4

5

6

7

8

9

В-1

В-2

В-3

Передний план

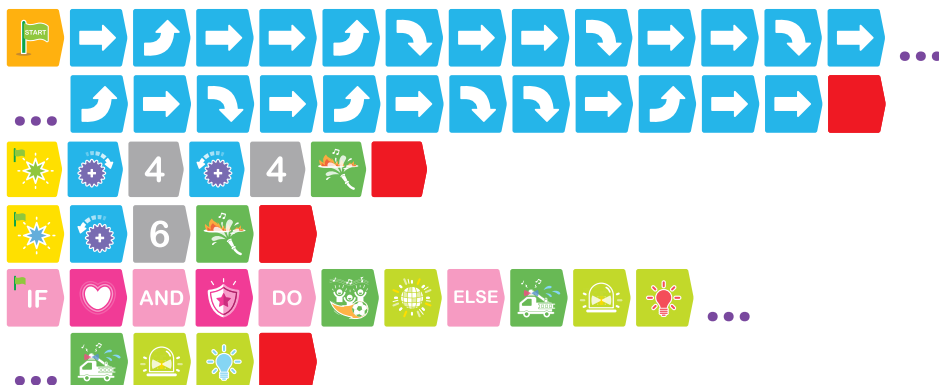
Сделано

Поместите на передачу

Умное руководство Веб-сервис

Теперь прикрепите машину к базовой карте пластиковыми полосками.

[код]



Что происходит?

Основная программа перемещает пожарную машину к базовой карте с зеленой звездой, затем с синей звездой и, наконец, к карточке Event 1 (условный элемент 1). Зеленые и синие функции работают так же, как зеленые функции работают в предыдущих уроках. Условное утверждение воспроизводит подбадривающий звук, если робот сканирует карточки Event 4(условный элемент 4) и Event 1(условный элемент 1). В противном случае пожарная машина воспроизводит звук сирены и мигает путь к огню.

Попробуйте это: поставьте кодовую карточку Event 1(условный элемент 1) к началу маршрута. Напишите программу, которая заставляет пожарную машину вернуться на стартовую карту после завершения двух миссий (спасений).



Видео работы модели



1



Собрал модель

2



Участвовал в эксперименте

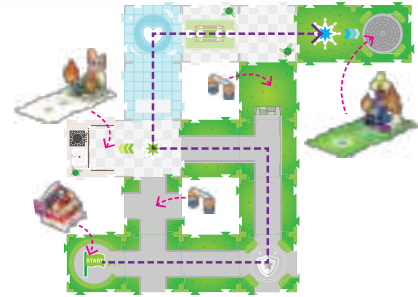
3



Выполнил творческое задание



Есть барьеры по всему городу. Пожарной машине сначала нужно спасти кошку, а затем спасти птицу, избегая барьеров на этом пути. После этого робот тушит каждый огонь, он воспроизводит одбадривающий звук.



Концепция кодирования

Разделение программы на подпрограммы известно как модульное программирование. Модуль является частью нечто большего. Если вещь может быть легко разделена на части, или ее можно разбить на разные части, мы говорим, что это модульное. Гиги блоки могут быть лучшим примером модульной системы. Вы можете взять кучу разных блоков и использовать их для создания разных моделей. Целью модульных решений проблемы является то, что большие проблемы могут быть решены путем решения множества мелких проблем одна за другой. Этот метод решения проблем может быть применен рекурсивно. После проблема была разложена до такой степени, что она больше не может быть разложена. Люди решают проблемы, которые у них есть, а затем объединяют ответы. Таким образом, большие проблемы решаются!

Мозговой штурм

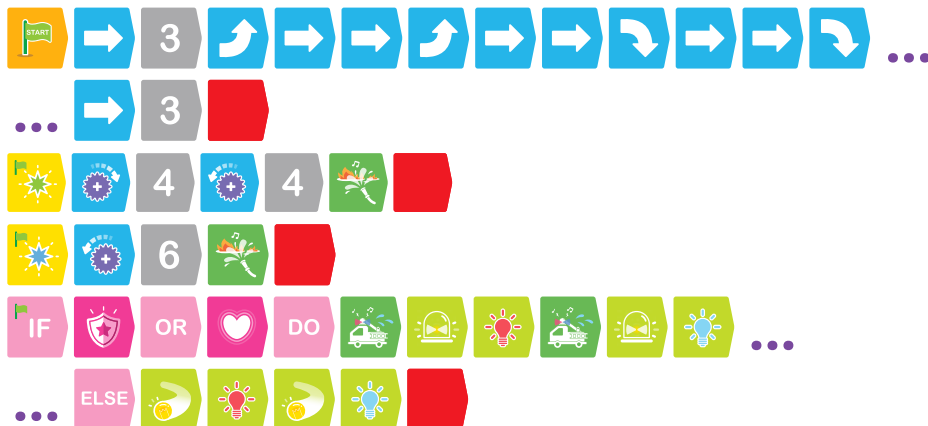
Применяет ли программа в этом уроке модульное программирование? Если да, сколько подпрограмм в программе этого урока?

Пожарная машина и Кошка, нуждающаяся в спасении, барьеры и птица, нуждающаяся в спасении, список запасных частей и этапы сборки:

Пожалуйста, обратитесь к Урокам 11 и 13.



[КОД]



Что происходит?

Основная программа перемещает пожарную машину к базовой маршрутной карте с помощью зеленой звезды, а затем синей звезды. Зеленые и синие функции работают так же, как в предыдущем уроке. Условное утверждение воспроизводит звук сирены и мигает красным и синим, если робот сканирует кодовые карточки Event 1 (условный элемент 1) или Event 4 (условный элемент 4). В противном случае, воспроизводится эффект красно-синего света.

Попробуйте это: принять вызов программирования маршрута для пожарной машины, но на этот раз вы не можете использовать какой-либо левый поворот карт. Как пожарная машина может завершить миссию?



.....

.....



Видео работы модели



1



Собрал модель

2



Участвовал в эксперименте

3



Выполнил творческое задание

15 Конкурсная работа 3

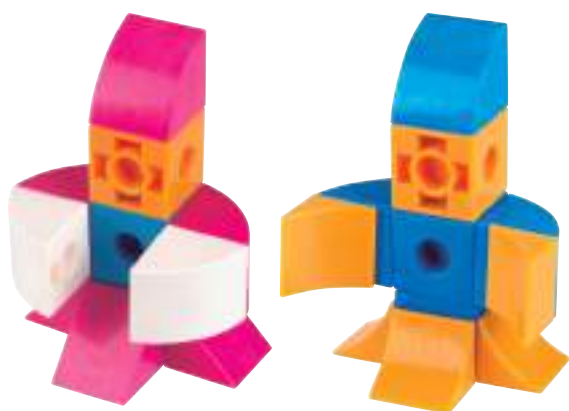
Сделайте модель светофора и поместите ее рядом с трассой, где находится пожарная машина по возвращении к пожарной части (стартовая зона). Пусть пожарная машина остановится, чтобы спросить указания от горожан, говоря «А?», а затем посещает два пожара, чтобы их потушить. Когда пожарная машина возвращается на старт, она остановится на три секунды, когда увидит красный светофор.



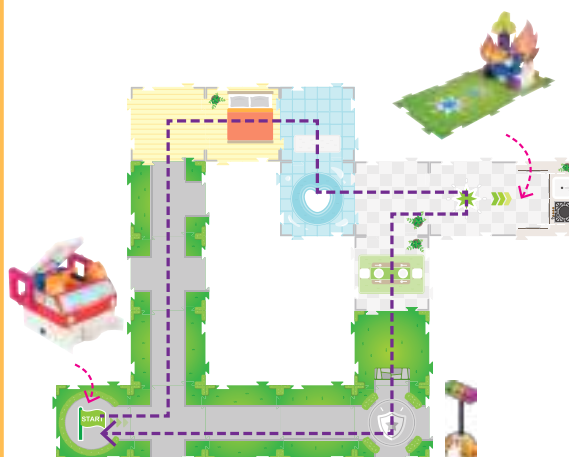
1. Пожарная машина



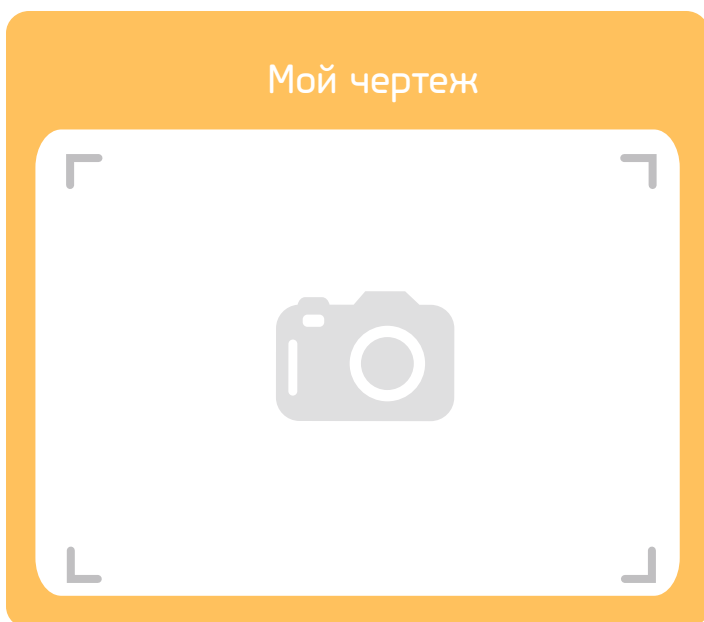
2. Птица, нуждающаяся в спасении



3. Городские жители

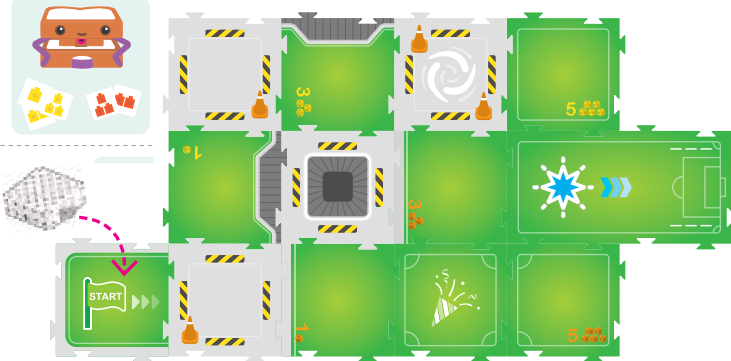


4. Маршрутная карта



урок 16

Найди кубики того же цвета



Сканируйте код «Урок16» рисунок на стр. 81. Программа водит робота только по маршрутным картам с номерами одинакового цвета, и в конечном итоге приводит на синюю звезду.

Вы заметили, что на картах напечатаны два разных по цвету строительных блока? Попробуйте найти два разные способы решения этой проблемы.

Концепция
кодирования

Слово «алгоритм» часто используется в компьютерном программировании. Алгоритм является частью компьютера. Программа используется для решения проблемы с конкретной последовательностью расчетов или шагов.

Алгоритм является пошаговым, поэтапным методом решения проблем. Алгоритмы не являются языком программирования, это мыслительный процесс.

Особенностями алгоритмов являются:

1. Ввод (но не обязательно)
2. Выход
3. Ограничение (результаты должны быть получены в ограниченном количестве шагов)
4. Ясность (без двусмысленности)
5. Эффективность (каждый шаг должен быть осуществимым).

Мозговой штурм

Теперь разработайте алгоритм или серию алгоритмов, которые могут производить крем и желе для бутерброда.

Список деталей

52



x1

53



x1

56



x1

[код]

Нахождение оранжевых чисел:



Нахождение желтых чисел:



Попробуйте это: используя только карты «Двигаться назад», но не переместить карты вперед, написать программу, чтобы робот следовал по пути на маршрутных картах с желтыми строительными блоками.



.....

.....



Видео работы модели



1



Собрал модель

2



Участвовал в эксперименте

3



Выполнил творческое задание



Отсканируйте рисунок кода «Урок 17» на стр. 81. Запрограммируйте робот на движение только на карточках с номерами одинакового значения (или количества), напечатанными на них, и в конечном итоге прийти на синюю звезду.



Концепция кодирования

Данные - это информация, которую можно хранить, читать и обновлять. Это может быть принято практически в любой форме, в том числе слова, цифры, голоса, изображения, видео. Данные и информация неразделима. Данные сами по себе не имеют смысла, но это становится информацией, когда это интерпретируется и вызывает поведение. Данные бывают двух основных типов: дискретные и непрерывные. Дискретный - это цифровой, а непрерывный - аналоговый.

Мозговой штурм

В компьютерной системе, как единица (двоичная цифра или бит) представлена в двоичной системе счисления?

Список деталей

52



x1

53



x1

56



x1

[код]

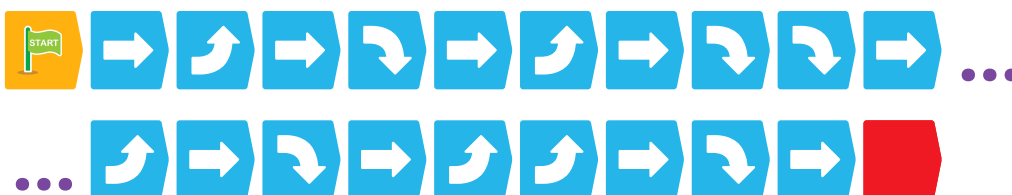
2.3 Нахождение обоих 3-х:



2.3 б Нахождение обоих 1:



2.3 с Нахождение обоих 2-х:



Попробуйте написать самую короткую программу, используя циклы.



Видео работы модели



1



Собрал модель

2



Участвовал в эксперименте

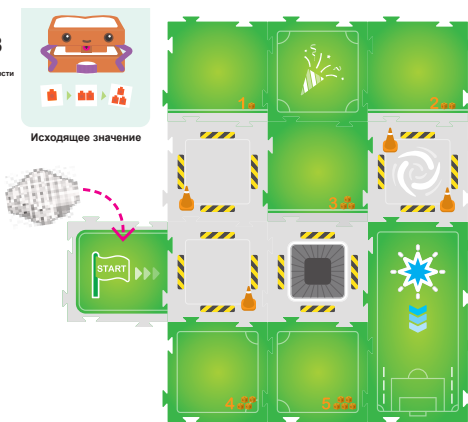
3



Выполнил творческое задание

Урок 18

Найди кубики в последовательности



Отсканируйте графический код «Увеличение значения» на стр. 81.

Запрограммируйте робота на движение на маршрутной карте с напечатанными на ней номерами возрастающего значения, заканчиваясь на синей звезде.

Концепция кодирования

Различные типы операторов числовых данных включают в себя: операторы присваивания (+, -, *, /, %), реляционные операторы (>, >=, ==, !=, <, <=), логические операторы. Этот Урок использует реляционные операторы. Реляционный оператор проверяет связь между двумя числами, и имеет только два условия (правда / ложь). Например, это выражение «3 < 5» "правда"; тогда как «3 = 5» означает «ложь».

На этом уроке робот оценит, является ли решение правильным, расположены ли в порядке увеличения или уменьшения значения, после чего робот заканчивает программу. Если робот следует программе, которая требует рационального оператора «1 < 2 < 3», когда он следует путь карты номер 1, затем 2, а затем 3, он будет воспроизводить Счастливое звучание «правильной» музыки и произойдет вспышка ее света.

Мозговой штурм

Теперь решите, являются ли следующие реляционные операторы истинными или нет. А. $9 > 3$ В. $5 < 2$ С. $8 > 4 > 3$.

Список деталей

52



x1

53



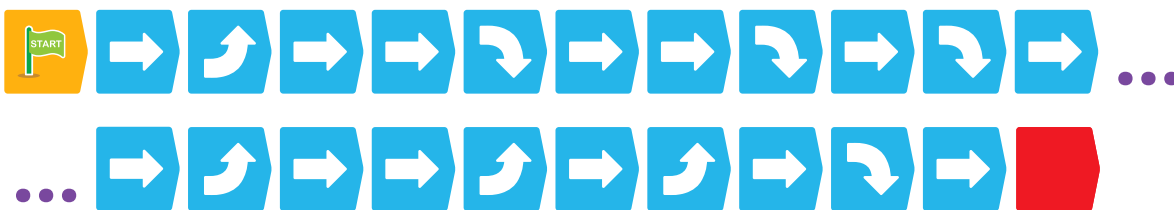
x1

56



x1

[код]



(Увеличение значения)

Попробуйте это: отсканируйте графический код «Уменьшающееся значение» на странице 81. Пожалуйста, напишите программу, чтобы робот проходил путь, который следует логике уменьшающегося значения.



.....

.....



Видео работы модели



1



Собрал модель

2

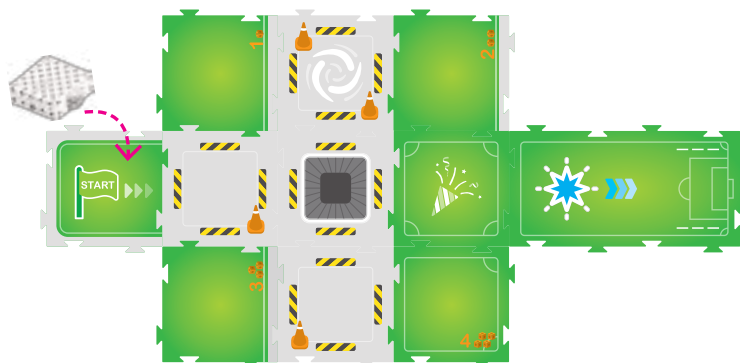


Участвовал в эксперименте

3



Выполнил творческое задание



Отсканируйте графические числа на странице 82. Напишите программу движения робота на маршрутной карте по картам с напечатанными номерами на тех, которые складываются в номер, который вы отсканировали, и

заканчивайте движение на синей звезде.

Подумайте об этом. Напишите программу для движения робота на маршрутной карте по картам с напечатанными числами, которые составляют в сумме 5.

Концепция кодирования

В программе переменная похожа на контейнер, который может быть использован для хранения и изменения данных. Содержание переменной можно изменить во время программирования. Одна переменная может хранить только одну часть данных, а две части данных требуют две переменные. Если вы хотите использовать и хранить много данных, вы можете использовать «массив структуры данных».

Переменная в этом уроке установлена в 5, что может быть сделано либо значения 1 и 4 на маршрутной карте; или значениями 2 и 3.

Мозговой штурм

Если вы установите количество раз, когда робот поднимает руки в качестве значения переменной, сколько раз робот поднимет руки, согласно следующей программе?

Установите переменную на 3
 Переменная = переменная + 5

Список деталей

52



x1

53



x1

56



x1

[код]

Добавление до 5:



Попробуйте это: установите накопленную переменную на 10. Теперь напишите программу для пути, которая добавляет до 10.



.....

.....



Видео работы модели



1



Собрал модель

2



Участвовал в эксперименте

3



Выполнил творческое задание

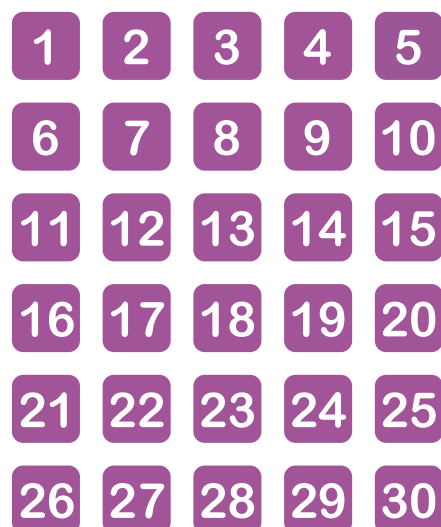
Основываясь на предыдущих уроках, напишите программу, которая перемещает робота на маршрутной карте по картам с напечатанными номерами, так что в сумме получалось 15. Обязательно используйте кодовые карточки циклов в вашей программе!



1. Базовый роботизированный блок

Урок 19

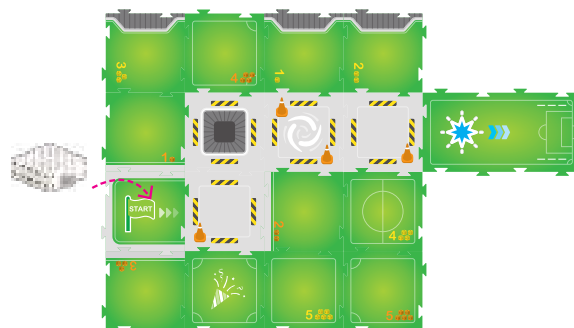
Найдите кубики со значениями, сумма которых равняется заданному числу



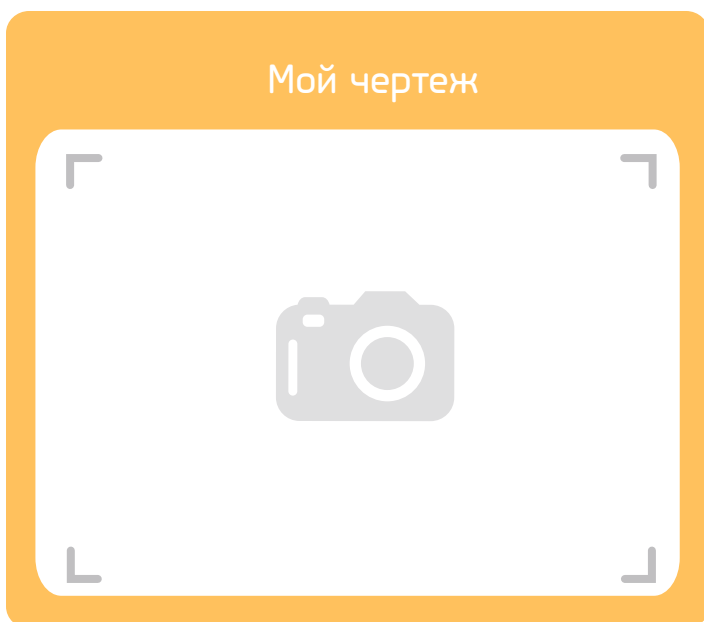
2. Найдите кубики, составляющие сумму данного числа



3. Простой зеленый цикл/Простой красный цикл



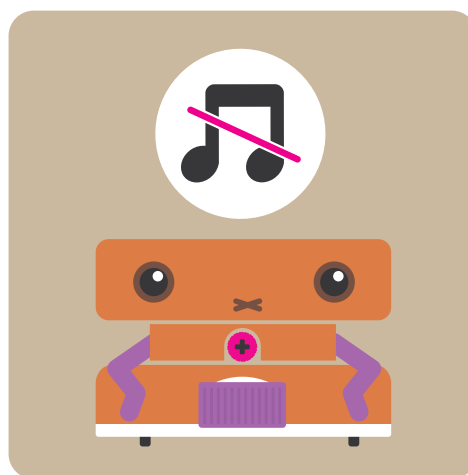
4. Маршрутная карта



Фоновая музыка



ВКЛЮЧЕНО

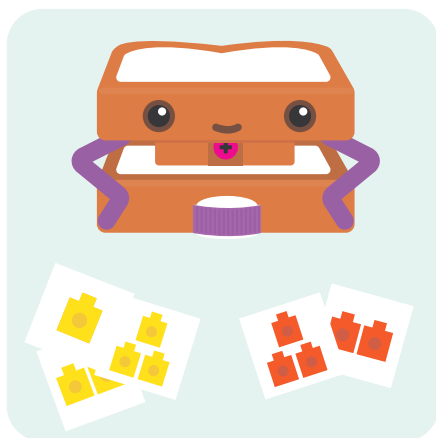


ВЫКЛЮЧЕНО

Математическая программа

Урок 16

Найди кубики того же цвета



Урок 17

Найди кубики равного значения

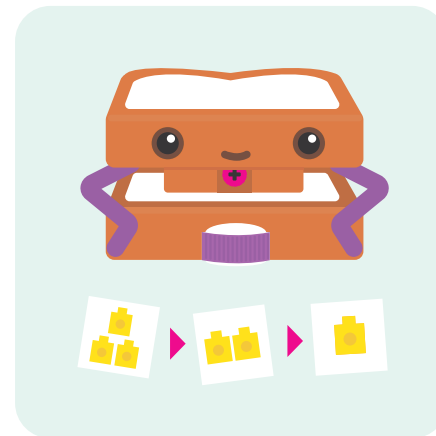


Урок 18

Найди кубики в последовательности



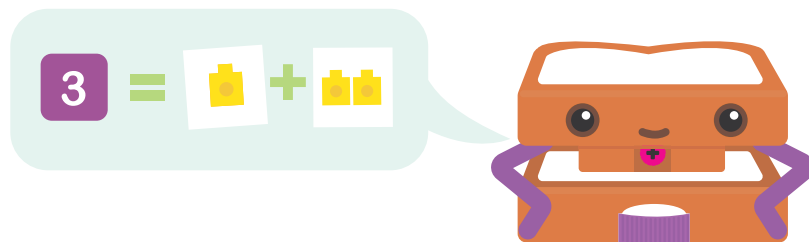
Исходящее значение



Нисходящее значение

Урок 19

Найдите кубики со значениями, сумма которых равняется заданному числу



1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25
26	27	28	29	30



Эксклюзивный дистрибьютор в России:
ООО "СИТИ"
241050, г. Брянск, ул. Трудовая, 1А
тел. +7 (4832)64-92-70
WWW.iqcamp.ru

Head office:
IQ CAMP OU
REG. NR. 12094213, EE101452816, Katusepapi 6-328,
Tallinn, Estonia, 11412