

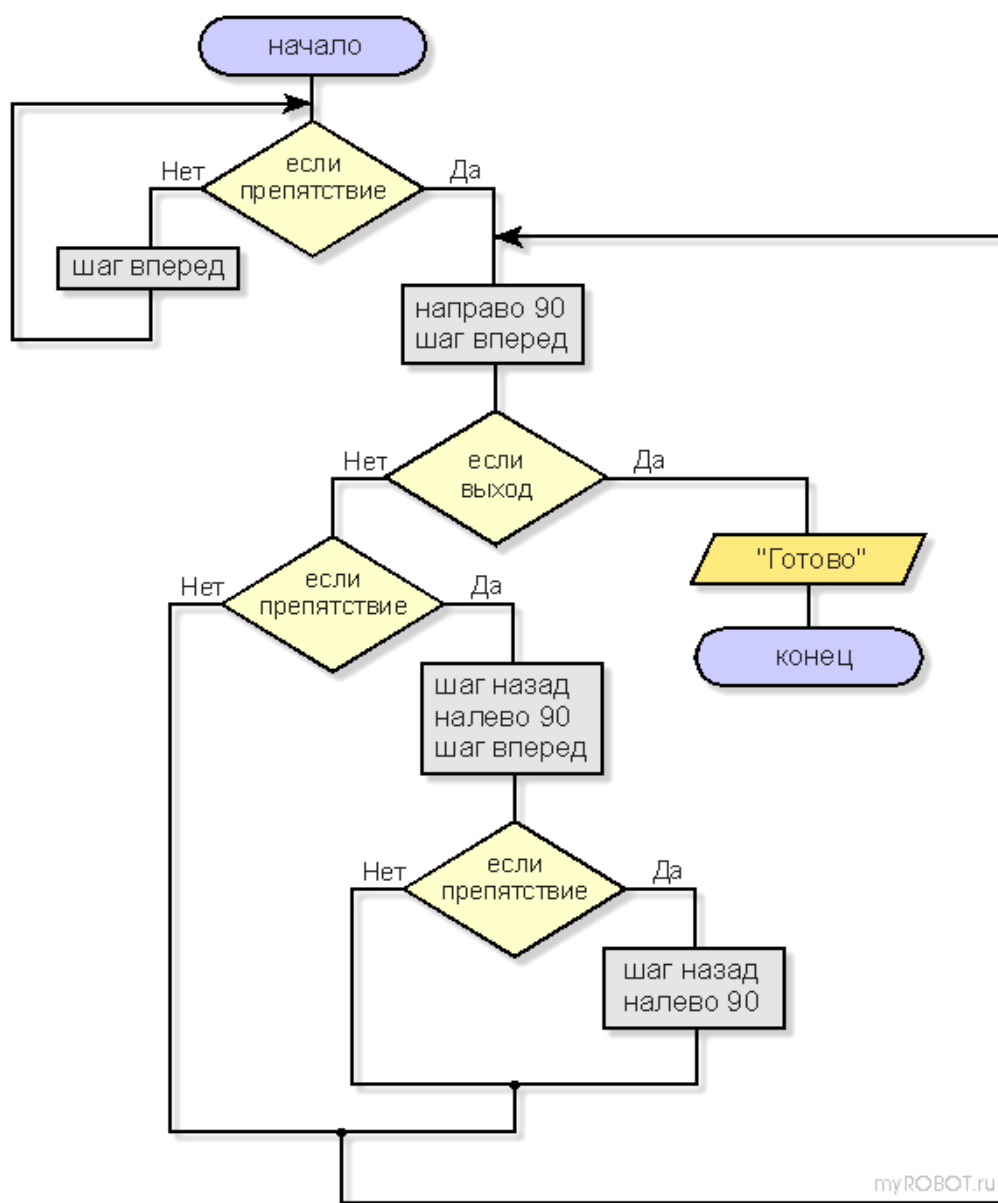
В начале своей работы робот должен найти стену, по которой он будет следовать. Для этого он может просто двигаться вперед, пока не упрется в преграду.

После того как робот наткнулся на препятствие, он начинает передвигаться в соответствии с правилом "правой руки".

Двигаясь вдоль стены, робот следит, есть ли проход справа. Если проход есть, робот должен идти по нему, чтобы не оторваться от стены справа.

Если прохода нет - впереди стена - робот поворачивает налево. Если прохода снова нет, он еще раз поворачивает налево, таким образом разворачиваясь на 180 градусов, и идет в обратном направлении.

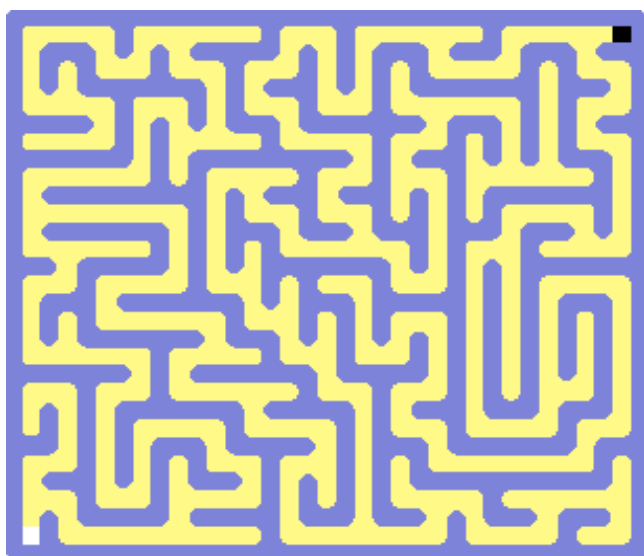
Блок-схема алгоритма для робота, работающего по правилу "правой руки", представлена на рисунке.



Попробуем проверить работу данного алгоритма и напишем для него программу. Для этой цели обратимся к среде программирования [GameLogo](#). Эта среда является удобным средством для моделирования различных алгоритмов, связанных с управлением роботами. В ней есть исполнитель черепаха, который по своей сути является не чем иным, как самым настоящим роботом. Черепаха располагает очень удобным набором команд - вперед, направо, налево, назад. Кроме того, в центре черепахи есть датчик, принимающий значение от 0 до 100, в зависимости от тона поверхности, на которой она находится.

Диалект языка Лого, который мы будем использовать, очень прост и похож на Basic. Познакомиться с командами языка можно [здесь](#). А бесплатно скачать среду программирования GameLogo - [здесь](#). Размер дистрибутива невелик - всего 1 Mb.

В архиве с GameLogo есть фоны с лабиринтами, одним из которых мы и воспользуемся.



Файл maze1.gif

В самом начале программы дадим команду черепахе, чтобы она подняла перо (по умолчанию черепаха оставляет после себя след).

Размер поля - 800 на 600 точек. Исходное положение для черепахи находится в месте с координатами 115, 545 (белый квадрат).

Цвет дорожек лабиринта - светлый, на них датчик будет принимать значения больше 50. Цвет стен лабиринта - темный, значение датчика будет меньше 50. Выход из лабиринта представлен черным квадратом, значение датчика над которым будет равно 0.

Объявим переменную флаг, с помощью которой будем контролировать, достигнут ли выход из лабиринта.

Напишем программу и запустим ее с помощью большой красной кнопки с надписью "Выполнить".

```
переменная флаг
флаг = 0

фон = maze1.gif
    поднять перо
        место 115, 545

' поиск первой стены

повторять пока датчик > 50 {
    вперед 12
}

' правило правой руки

повторять пока флаг = 0 {

    направо 90
    вперед 12

    если датчик = 0 то

        флаг = 1

    иначе

        если датчик < 50 то

            назад 12
            налево 90
            вперед 12

            если датчик < 50 то
                назад 12
                налево 90
            конец условия

        конец условия

    конец условия
}

пиши "цель достигнута"
```

Если известно, что у лабиринта нет отдельно стоящих стенок, то есть нет замкнутых маршрутов, по которым можно возвращаться в исходную точку, то такой лабиринт называют односвязным и его всегда можно обойти полностью, применив правило "одной руки".

Если же лабиринт содержит отдельно стоящие стенки, то, применяя правило "одной руки", не всегда можно пройти все коридоры и тупики. Лабиринты с отдельно стоящими стенками и с замкнутыми маршрутами называются многосвязными. При этом многосвязные лабиринты можно разделить на две группы: без "петли" вокруг цели (замкнутый маршрут не проходит

вокруг цели) и с замкнутой "петлей" вокруг цели (цель можно обойти по замкнутому маршруту).



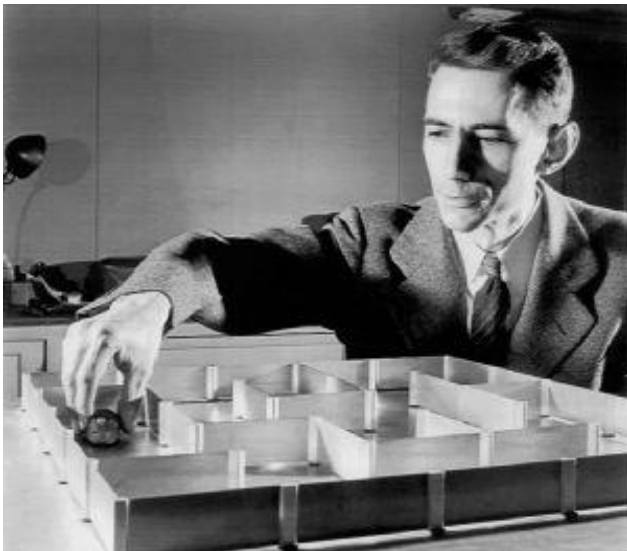
Робот для прохождения лабиринта на базе ATmega8 (соревнования Micromouse competition).

В многосвязных лабиринтах второй группы правило "одной руки" не работает и, применяя его, достичь цели невозможно. Но и эти лабиринты можно пройти, полагаясь на точный алгоритм.

Решение задачи о таких лабиринтах принадлежит сравнительно позднему времени, и начало ему положено Леонардом Эйлером. Эйлер не без оснований полагал, что выход из любого лабиринта может быть найден, и притом сравнительно простым путем.

Универсальный алгоритм прохождения любых лабиринтов был описан только через столетие в книге французского математика Э. Люка "Recreations matematicques", изданной в 1882 году. Интересно, что Люка при описании алгоритма указал на первенство другого французского математика М. Тремо. Таким образом, алгоритм стал известен как **алгоритм Люка-Тремо**.

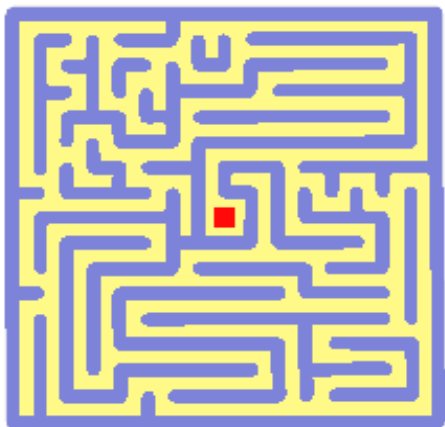
Тремо предлагает следующие правила: выйдя из любой точки лабиринта, надо сделать отметку на его стене (крест) и двигаться в произвольном направлении до тупика или перекрестка; в первом случае вернуться назад, поставить второй крест, свидетельствующий, что путь пройден дважды - туда и назад, и идти в направлении, не пройденном ни разу, или пройденном один раз; во втором - идти по произвольному направлению, отмечая каждый перекресток на входе и на выходе одним крестом; если на перекрестке один крест уже имеется, то следует идти новым путем, если нет - то пройденным путем, отметив его вторым крестом.



Клод Шеннон (Claude Elwood Shannon)

Зная алгоритм Тремо, можно скорректировать поведение легендарного Тесея. Вдохновленный подарком любимой Ариадны, он уверенно идет по лабиринту. Вдруг перед ним возникает ход, по которому уже протянута нить... Что делать? Ни в коем случае не пересекать ее, а вернуться по уже известному пути, сдваивая нить, пока не найдется еще один непройденный ход.

Применив вариант алгоритма Тремо, отец теории информации Клод Шеннон (Claude Elwood Shannon) построил одного из первых самообучающихся роботов. Шеннон дал ему звучное имя "Тесей", но в истории "Тесей" стал больше известен как "мышь" Шеннона. "Мышь" сначала обследовала весь лабиринт, а затем (во второй раз) проходила весь путь значительно быстрее, избегая участков, пройденных дважды.



Лабиринт на соревнованиях
Micromouse competition.

В наши дни роботы, проходящие лабиринт, являются участниками одного из самых интересных состязаний думающих машинок, которое проходит в нескольких странах мира. Эти соревнования носят общее название [Micromouse competition](#) и по своим техническим новациям относятся к лидерам робототехнического спорта.

На первой Российской Олимпиаде Роботов проводились соревнования, целью которых было прохождение своеобразного лабиринта: за наиболее короткое время, двигаясь через "открытые двери" в стенках, робот должен был добраться от места старта до места финиша. Контролировать свое движение робот мог по черным линиям, нанесенным на пол лабиринта.