

Сказка-описание маршрута-3.12-Челышев

Добрый вечер, Михаил Иванович!

Изначально пользователь через командный файл задает координаты начальной и конечной точек. Координаты через контроллер передаются в процессор. Процессор начинает строить маршрут и вести по нему ровер. Как он это делает?

У нас стоит задача, как можно быстрее добраться из точки А в точку В, поэтому чтобы минимизировать время затраченное на путь, процессор будет вести ровер по диагонали, с объездом препятствий, если же у точек совпадает какая-то из координат, то будет вести по этой координате (по прямой) с объездом препятствий.

Реализация:

Процессор смотрит разность координат точек А и В по х и по у и оценивает ее: больше нуля, равна нулю, меньше нуля. После оценки у нас есть несколько вариантов развития событий ($A(x_0, y_0)$, $B(x, y)$):

1. Если $g_x = (x - x_0) > 0$ и $g_y = (y - y_0) > 0$, значит ровер должен будет поехать по диагонали вверх вправо.
2. Если $g_x > 0$ и $g_y = 0$, то есть совпадение по координате у \Rightarrow ровер поедет по прямой (вправо).
3. Если $g_x > 0$ и $g_y < 0$, то ровер поедет по диагонали вниз вправо.
4. Если $g_x = 0$ и $g_y > 0$, то ровер поедет вверх по прямой.
5. Если $g_x = 0$ и $g_y = 0$, то координаты точек А и В совпадают, а значит мы приехали, победа!
6. Если $g_x = 0$ и $g_y < 0$, то едем вниз по прямой.
7. Если $g_x < 0$ и $g_y > 0$, то едем по диагонали вверх влево.
8. Если $g_x < 0$ и $g_y = 0$, то едем влево по прямой.
9. Если $g_x < 0$ и $g_y < 0$, то едем по диагонали вниз влево.

Такую проверку мы проделываем, каждый такт. В зависимости от изменения координат ровера, будет и меняться направление его движения. Осталось разобраться с препятствиями, нам необходимо не просто объехать препятствие стоящее на пути, а сделать это быстро. Как же мы будем это делать?

Когда мы сталкиваемся с препятствием, процессор будет пытаться объехать препятствие, стараясь вернуть ровер назад на маршрут, то есть мы будем клонировать ровер и ехать в разные стороны (объезжать препятствие), в каждый такт, мы будем смотреть на наши разности координат, и пытаться вернуться на по оптимальной маршрут. Если вернуться туда не получается, то мы продолжаем объезжать препятствие. Так мы делаем до тех пор, пока не сможем вновь поехать по оптимальному маршруту. Таким образом ровер объезжает препятствие, координаты точек итогового маршрута ровера

записываются в отдельный файл, который потом в Gnuplot и показывает нам финальный, оптимальный маршрут.