Для расчета реализуем две функции:

```
     FUNCTION A*(start,end)
     FUNCTION RECONSTRUCT_PATH(fromset,start,end)

FUNCTION A*(start,end)

     closedset = the empty set
     openset = {start}
     fromset = the empty set
     G(start) = COST(start,start) = 0
     F(start) = G(start) + H(start,end)

     WHILE openset IS NOT EMPTY
       curr = MIN_F(openset)
       IF (curr = end) RETURN RECONSTRUCT_PATH(fromset,start,end)

       REMOVE curr FROM openset
       ADD curr TO closedset

       FOREACH neighbour OF curr neighbours

          IF neighbour IN closedset CONTINUE
          tentative_g_score = G(curr) + COST(curr,neighbour)

          IF neighbour NOT IN openset

              ADD neighbour TO openset
              tentative_is_better = TRUE

          ELSE

              IF tentative_g_score < G(neighbour)
                     tentative_is_better = TRUE
              ELSE
                     tentative_is_better = FALSE

           IF tentative_is_better = TRUE
               fromset(neighbour) = curr
               G(neighbour) = tentative_g_score
               F(neighbour) = G(neighbour) + H(neighbour, end)

RETURN FAILURE


FUNCTION RECONSTRUCT_PATH(fromset,start,end)
     pathset = the empty list
     curr = end
     ADD curr TO pathset

     WHILE curr <> start
          curr = fromset(curr)
          ADD curr TO pathset
RETURN REVERSE(pathset)
```