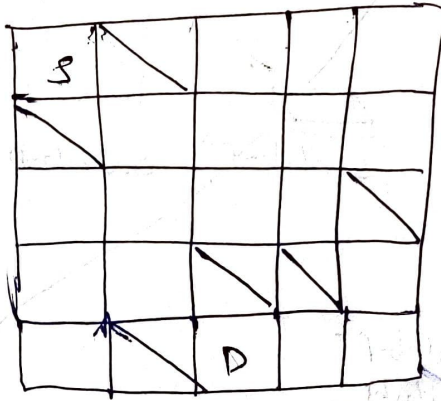AI - Lab Test 1

Program 4 :-

A* Search with euclidean distance :-

Pseudocode :-

```
def f (curr-state, goal-state):
    h = sqrt (((curr-state.x - goal-state.x)**2 + (curr-state.y
                                 - goal-state.y)**2)
    return h

def a-star-search:
    def astarsearch (grid, src, dest):
        if isValid (src.first, src.second) == false:
            return
        if isValid (dest.first, dest.second) == false:
            return
        if isDestination (src.first, src.second, dest):
            return grid    # Already at destination

        x = grid.x ;
        y = grid.y.
        eclid = f(grid, dest)
```

(1)

```
if grid[x][y] == 2:
        print " found at %d, %d", (x, y)
        return True

elif grid[x][y] == 1:
        print "wall at %d, %d", (x, y)
        return false

elif grid[x][y] == 3:
        print "visited %d, %d", (x, y)
        return False.
```

#1 = wall #2 = available block 3 = visited block.

```
grid[x][y] = 3

if(( x < len(grid)-1) and astarsearch(x-1, y)) or
        (y>0 and astarsearch(x, y-1)) or x> and actarsearch(x-1,
                                                            y)) or

        (y < len(grid)-1 and astarsearch(x, y-1)):

                return True

    return false.
```

```
astarsearch(0, 0).
```

Hence, ~~both~~ A* search ~~with~~ ~~or~~ using euclidean function

is included.