Lab-4 A* algo

```
def astar_search(map, start, end):
    open = []
    closed = []
    start_node = Node(start, None)
    goal_node = Node(end, None)
    open.append(start_node)

    while len(open) > 0:
        open.sort()
        current_node = open.pop(0)
        closed.append(current_node)

    if current_node != start_node:
        path.append(current_node.position)
        current_node = current_node.parent
        return path[::-1]
    (x, y) = current_node.position
    neighbours = [(x-1, y), (x+1, y), (x, y-1),
                  (x, y+1)]

for next in neighbours:
    map_value = map.get(next)
    if (map_value == '#'):
        continue
    neighbour = Node(next, current_node)
    if (neighbour in closed):
        continue
```

... continued

```
neighbour.g = abs(neighbor.position[0] - start_node.
position[0]) + abs(neighbor.position[1] -
start_node.position[1])
neighbor.h = abs(neighbor.postion[0] - goal_
node.position[0]) + abs(neighbor.position[1] -
goal_node.position[1])
neighbor.f = neighbor.g + neighbor.h

if (add_to_open(open, neighbor) == true):
        open.append(neighbor)
return None
```