**SOFTWARE LABORATORY 1**

**GROUP B – EXPERIMENT 2**

**TITLE:**

Implement A star (A\*) Algorithm for any game search problem.

**CODE:**

import heapq

def a\_star\_search(start, goal, grid):

"""

start, goal: (row, col) tuples grid: 2D list where 0 = free cell, 1 = obstacle

"""

rows, cols = len(grid), len(grid[0])

def heuristic(a, b):

# Manhattan distance (works well for grids without diagonal movement)

return abs(a[0] - b[0]) + abs(a[1] - b[1])

open\_list = []

heapq.heappush(open\_list, (0, start))

came\_from = {}

g\_score = {start: 0}

f\_score = {start: heuristic(start, goal)}

closed\_set = set()

while open\_list:

current = heapq.heappop(open\_list)[1]

if current == goal:

return reconstruct\_path(came\_from, current)

closed\_set.add(current)

# 4 possible moves (up, down, left, right)

for dr, dc in [(0, 1), (1, 0), (0, -1), (-1, 0)]:

neighbor = (current[0] + dr, current[1] + dc)

if not (0 <= neighbor[0] < rows and 0 <= neighbor[1] < cols):

continue # out of bounds

if grid[neighbor[0]][neighbor[1]] == 1:

continue # obstacle

if neighbor in closed\_set:

continue

tentative\_g = g\_score[current] + 1

if tentative\_g < g\_score.get(neighbor, float('inf')):

came\_from[neighbor] = current

g\_score[neighbor] = tentative\_g

f\_score[neighbor] = tentative\_g + heuristic(neighbor, goal)

heapq.heappush(open\_list, (f\_score[neighbor], neighbor))

return None # No path found

def reconstruct\_path(came\_from, current):

path = [current]

while current in came\_from:

current = came\_from[current]

path.append(current)

path.reverse()

return path

# Example usage:

grid = [

[0, 0, 0, 0, 0],

[0, 1, 1, 1, 0],

[0, 0, 0, 1, 0],

[1, 1, 0, 0, 0],

[0, 0, 0, 0, 0]

]

start = (0, 0) goal = (4, 4)

path = a\_star\_search(start, goal, grid)

if path:

print("Path found:", path)

else:

print("No path found")

**OUTPUT:**

/home/student/PycharmProjects/PythonProject/.venv/bin/python /home/student/PycharmProjects/PythonProject/ass2.py

Path found: [(0, 0), (0, 1), (0, 2), (0, 3), (0, 4), (1, 4), (2, 4), (3, 4), (4, 4)]

Process finished with exit code 0

