***EXP 04: A\* Search Algorithm***

***Program:***

graph = [

['A', 'H', 7, 0],

['A', 'B', 1, 3],

['A', 'C', 2, 4],

['B', 'D', 4, 2],

['B', 'E', 6, 6],

['C', 'F', 3, 3],

['C', 'G', 2, 1],

['D', 'E', 7, 6],

['D', 'H', 5, 0],

['F', 'H', 1, 0],

['G', 'H', 2, 0],

]

nodes = set([]) # Set so that it has unique values only

for edges in graph:

nodes.add(edges[0])

nodes.add(edges[1])

# Init the requirements

cost = dict()

path = dict()

openl = set()

closel = set()

for node in nodes:

cost[node] = 10000

path[node] = ''

start, goal = input("Enter the start and goal node: ").split(" ") # Input

cost[start] = 0 # Init wrt to start inputed

path[start] = start

openl.add(start)

def AStar(start, goal, openl, closel, cost):

if start in openl:

openl.remove(start)

closel.add(start) # Explore the start as added to close with cost

for edges in graph:

if edges[0] == start: # Find edges from start to a node

from\_node = edges[0]

to\_node = edges[1]

#print("found edge : ", from\_node, " -> ", to\_node)

# If current cost to reach to\_node is lesser then existing cost

NEW\_COST = cost[from\_node]+edges[2]+edges[3]

OLD\_COST = cost[to\_node]

#print(NEW\_COST, OLD\_COST)

if NEW\_COST < OLD\_COST:

print("found that path to: ", to\_node, " via ", from\_node, " is shorter with cost: ",

cost[from\_node]+edges[2]+edges[3], " which is < ", cost[to\_node])

cost[to\_node] = NEW\_COST

path[edges[1]] = path[from\_node]+" - > "+to\_node

cost[start] = 10000 # Prevent start from getting selected again

smallest = min(cost, key=cost.get)

openl.add(smallest)

if smallest not in closel:

AStar(smallest, goal, openl, closel, cost)

AStar(start, goal, openl, closel, cost)

print("Path is: ", path[goal])

***Output:***

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

Enter the start and goal node: A H

found that path to: H via A is shorter with cost: 7 which is < 10000

found that path to: B via A is shorter with cost: 4 which is < 10000

found that path to: C via A is shorter with cost: 6 which is < 10000

found that path to: D via B is shorter with cost: 10 which is < 10000

found that path to: E via B is shorter with cost: 16 which is < 10000

found that path to: F via C is shorter with cost: 12 which is < 10000

found that path to: G via C is shorter with cost: 9 which is < 10000

found that path to: H via G is shorter with cost: 11 which is < 10000

Path is: A - > C - > G - > H

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***