Artificial Intelligence Lab Assignment 2

tree = {'S': [['A', 1], ['B', 2]],

'A': [['S', 1]],

'B': [['S', 2], ['C', 3], ['D', 4]],

'C': [['B', 2], ['E', 5], ['F', 6]],

'D': [['B', 4], ['G', 7]],

'E': [['C', 5]],

'F': [['C', 6]]

}

heuristic = {'S': 0, 'A': 5000, 'B': 2, 'C': 3, 'D': 4, 'E': 5000, 'F': 5000, 'G': 0}

def aStar():

g = 0

neighbour = []

current = 'S'

closed = []

leaf = []

while current!='G':

neighbour = []

closed.append(current)

for i in tree[current]:

n = i[0]

if n not in closed:

h = heuristic[n]

g\_for\_n = g + i[1]

f = g\_for\_n + h

neighbour.append([n,f])

if neighbour == []:

leaf.append(current)

removed = current

current = [i[0] for i in tree[current] if i[0] not in leaf][0]

g = g - [i[1] for i in tree[current] if i[0] == removed][0]

else:

smallest\_f = neighbour[0][1]

smallest\_f\_node = neighbour[0][0]

for i in neighbour:

if smallest\_f > i[1]:

smallest\_f = i[1]

smallest\_f\_node = i[0]

g = g + [i[1] for i in tree[current] if i[0]==smallest\_f\_node][0]

current = smallest\_f\_node

closed.append('G')

print("OPTIMUM PATH TO TRAVEL IS AS FOLLOWED ==> ",end='')

[print('--> ',i,end=" ")for i in closed]

print("\nREQUIRED COST WILL BE ==> ", g)

aStar()

Output

PS C:\Users\hp\Desktop\AI> python a.py

OPTIMUM PATH TO TRAVEL IS AS FOLLOWED ==> --> S --> B --> C --> E --> C --> F --> C --> B --> D --> G

REQUIRED COST WILL BE ==> 13

PS C:\Users\hp\Desktop\AI>