

**SUBJECT: AI LAB**

**ASSIGNMENT # 5 & 6**

**SUBMITTED TO: SIR QAZI WAQAS KHAN**

**SUBMITTED BY: RABIA ZEB**

**REGISTRATION NO: SP20-BCS-037**

**DATED: 27TH APRIL 2022**

**ASSIGNMENT # 5**

d={'o':[('z',71),('s',151)],

'z':[('a',75),('o',71)],

'a':[('z',75),('s',140),('t',118)],

't':[('a',118),('l',111)],

'l':[('t',111),('m',70)],

'm':[('l',70),('d',75)],

'd':[('m',75),('c',120)],

'c':[('d',120),('r',146),('p',138)],

'r':[('c',146),('p',97),('s',80)],

's':[('o',151),('a',140),('r',80),('f',99)],

'f':[('s',99),('b',211)],

'p':[('r',97),('c',138),('b',101)],

'b':[('g',90),('p',101),('u',85),('f',211)],

'g':[('b',90)],

'u':[('b',85),('h',98),('v',142)],

'h':[('e',86),('u',98)],

'v':[('u',142),('i',92)],

'i':[('n',87),('v',92)],

'n':[('i',87)],

'e':[('h',86)]

}

lis = [('z',71),('s',99),('g',90)]

def my(x):

return x[1]

lis = sorted(lis,key=my)

lis

lis

new = ('b',90)

def build\_dcit(lis):

di = {}

for i in lis:

name = i[0]

cost = i[1]

di[name] = cost

return di

new[0] in build\_dcit(lis)

def build\_dcitt(lis):

di = {}

for i in lis:

di[i[0]] = i[1]

return di

new[0] in build\_dcitt(lis)

def build\_dcitt(lis):

return {i[0]:i[1] for i in lis}

new[0] in build\_dcitt(lis)

for i in lis:

if(i[0] == new[0]):

print(i)

if i[1]<new[1]:

pass

else:

ind = lis.index(i)

lis[ind] = new

lis

s =[]

s.append(('o',0))

s

def UCS\_new(d,goal):

q=[]

q.append(('o',0))

explored =[]

while len(q)>0:

print('Queue :',q)

node = q.pop(0)

if node[0] not in explored:

explored.append(node[0])

if node[0] == goal:

return explored

child = d.get(node[0])

for i in child:

cost = i[1] + node[1]

name = i[0]

new = (name,cost)

if new[0] not in explored and new[0] not in build\_dcit(q):

q.append(new)

elif new[0] in build\_dcit(q):

old\_data = build\_dcit(q)

old\_cost = old\_data[new[0]]

old\_name = new[0]

old = (old\_name,old\_cost)

if new[1]<old[1]:

inde = q.index(old)

q[inde] = new

else:

pass

q = sorted(q,key=my)

print(50\*'\*')

UCS\_new(d,'g')

D

ASSIGNMENT # 6

d={'o':[('z',71),('s',151)],

'z':[('a',75),('o',71)],

'a':[('z',75),('s',140),('t',118)],

't':[('a',118),('l',111)],

'l':[('t',111),('m',70)],

'm':[('l',70),('d',75)],

'd':[('m',75),('c',120)],

'c':[('d',120),('r',146),('p',138)],

'r':[('c',146),('p',97),('s',80)],

's':[('o',151),('a',140),('r',80),('f',99)],

'f':[('s',99),('b',211)],

'p':[('r',97),('c',138),('b',101)],

'b':[('g',90),('p',101),('u',85),('f',211)],

'g':[('b',90)],

'u':[('b',85),('h',98),('v',142)],

'h':[('e',86),('u',98)],

'v':[('u',142),('i',92)],

'i':[('n',87),('v',92)],

'n':[('i',87)],

'e':[('h',86)]

}

h\_n\_ = {}

total\_nodes = list(d.keys())

total\_nodes

import random

for i in total\_nodes:

rand = random.randint(0,15)

h\_n\_[i] = rand

h\_n\_

def my(x):

return x[1]

def build\_dcit(lis):

di = {}

for i in lis:

name = i[0]

cost = i[1]

di[name] = cost

return di

def GBFS(start,goal):

q = []

start\_val = h\_n\_.get(start)

q.append((start,start\_val))

explored = []

expanded = []

while len(q)>0:

node = q.pop(0)

if node[0] not in explored:

explored.append(node[0])

if node[0] == goal:

print('Result :',explored,expanded,len(expanded))

return

child = d.get(node[0])

for i in child:

n\_key = i[0]

n\_val = h\_n\_.get(n\_key)

n\_tuple = n\_key,n\_val

if i[0] not in explored and i[0] not in build\_dcit(q):

q.append(n\_tuple)

expanded.append(node[0])

q = sorted(q,key= my)

return explored,expanded,len(expanded)

GBFS('o','g')