AHSANULLAH UNIVERSITY OF SCIENCE AND TECHNOLOGY

Department of Computer Science and Engineering

Program: Bachelor of Science in Computer Science and Engineering
Spring 2019

Course No: CSE 4108

Course Title: Artificial Intelligence Lab

Term Project No: 01

Topic: Reasoning of Propositional Logic

Date of Submission: 22 September, 2019

Submitted To: Dr. S.M. Abdullah Al-Mamun
Professor, Department of CSE, AUST.

Md. Siam Ansary
Part time Faculty, Department of CSE, AUST.

Submitted By,

Name: Iffatun Nessa
Student ID: 160104031
Group: A2

Propositional logic, also known as sentential logic and statement logic, is the branch of logic that studies ways of joining and/or modifying entire propositions, statements or sentences to form more complicated propositions, statements or sentences, as well as the logical relationships and properties that are derived from these methods of combining or altering statements. In propositional logic, the simplest statements are considered as indivisible units, and hence, propositional logic does not study those logical properties and relations that depend upon parts of statements that are not themselves statements on their own, such as the subject and predicate of a statement. According to mental model theory, propositional reasoning is a complex cognitive activity that requires the processing, activation, and maintenance of relevant information, and the moment-to-moment control and monitoring of the process of finding a solution.

So, for reasoning of propositional logic I implemented a goal based logical agent for simplified 'Treasure Hunt: A risky Business' game.

Input:

1. Name of the Player (to save in the database)

2. Name of the current move (up, down, right, left)

Major Steps of Processing:

First of all, the game board is defined in python code as the following picture:

Here,

M = Monster is here and the game is over

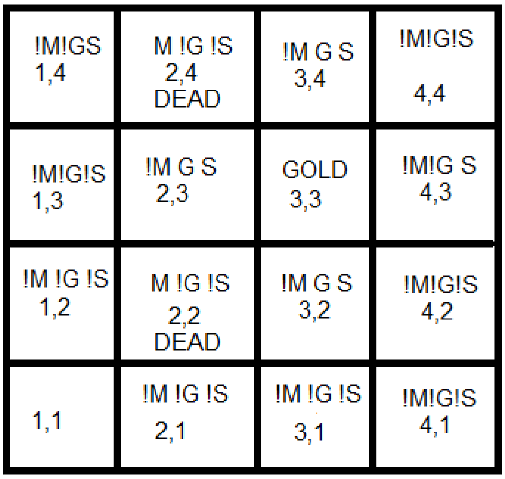
S = Smell of the monster (monster is in up/down/left/right)

G = Gold is glittering here (gold is in up/down/left/right)

!M = no monster

!S = no smell

!G = no glitter



!M!GS 1,4	M !G !S 2,4 DEAD	!M G S 3,4	!M!G!S 4,4
!M!G!S 1,3	!M G S 2,3	GOLD 3,3	!M!G S 4,3
!M !G !S 1,2	M !G !S 2,2 DEAD	!M G S 3,2	!M!G!S 4,2
1,1	!M !G !S 2,1	!M !G !S 3,1	!M!G!S 4,1

If the player goes to 3,3 = position 11 then he will win and get the points (+10), or if he found any of the monster then he will get -5 and it will be saved into the database.

Assumption:

If, i =1 and j=1

A i,j – the agent is in [i,j]

!M i,j – no moster in [i,j]

!G i,j – no glitter in [I,j]

!S i,j – no smell in [I,j]

Actions: MoveUp, MoveDown, MoveLeft, MoveRight, GoldFound, Killed, Clear etc.

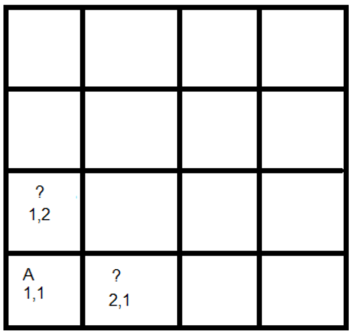
Initial KB:

1. $S_{i,j} \Leftrightarrow M_{i-1,j} \lor M_{i+1,j} \lor M_{i,j-1} \lor M_{i,j+1}$

2.$(i \geq 1) \land (j \geq 1) \land (i \leq 4) \land (j \leq 4)$

3.$! M_{1,1} \land !\ S_{1,1} \land ! G_{1,1}$

4. $A_{1,1}$



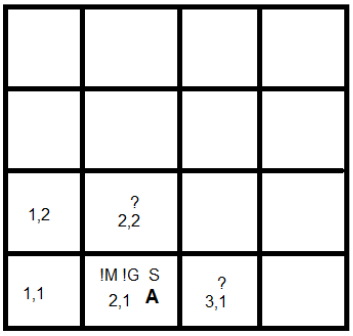
? 1,2			
A 1,1	? 2,1		

Reasoning:

Step 1. Move right.

For this step if the player wants to move right (2,1), he will get some hints for both up and right.

If he does not want to go right or up he can back in (1,1) and move to next step, but the limited moves will be reduced.

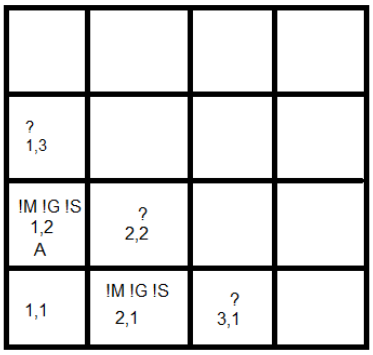


1,2	? 2,2		
1,1	!M !G S 2,1 A	? 3,1	

Step 2. Move up.

For this step if the player wants to move right (1,2), he will get some hints for both up and left.

This is how Step 3,4… up to 12 can be counted for every move.



? 1,3			
!M !G !S 1,2 A	? 2,2		
1,1	!M !G !S 2,1	? 3,1	

Implementation in python:

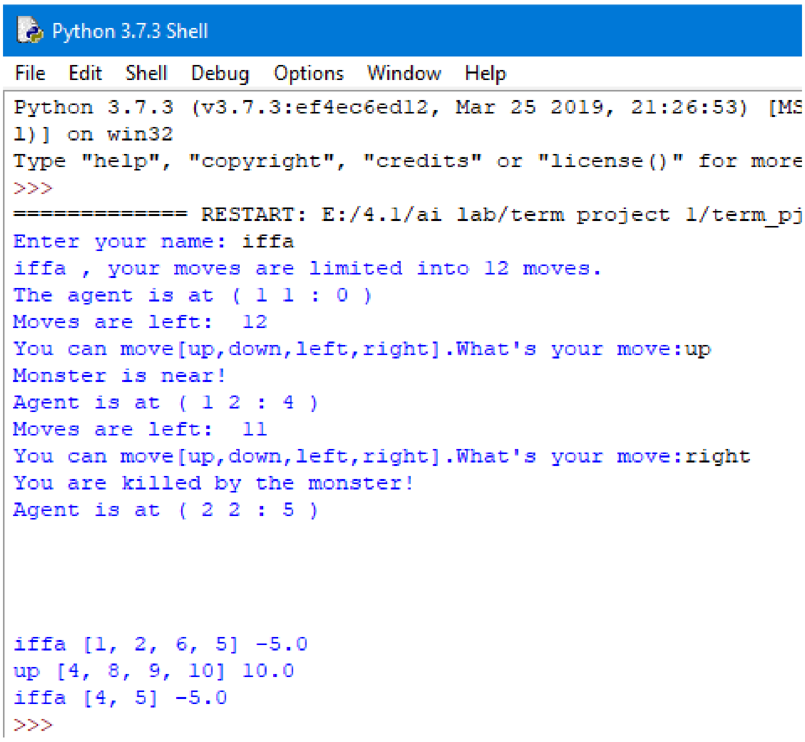
```
def position(ti,tj):
    if(ti == 1 and tj == 1):
        posi = 0
    elif(ti == 2 and tj == 1):
        posi = 1
    elif(ti == 3 and tj == 1):
        posi = 2
    elif(ti == 4 and tj == 1):
        posi = 3
    elif(ti == 1 and tj == 2):
        posi = 4
    elif(ti == 2 and tj == 2):
        posi = 5
    elif(ti == 3 and tj == 2):
        posi = 6
    elif(ti == 4 and tj == 2):
        posi = 7
    elif(ti == 1 and tj == 3):
        posi = 8
    elif(ti == 2 and tj == 3):
        posi = 9
    elif(ti == 3 and tj == 3):
        posi = 10
    elif(ti == 4 and tj == 3):
        posi = 11
    elif(ti == 1 and tj == 4):
        posi = 12
    elif(ti == 2 and tj == 4):
        posi = 13
    elif(ti == 3 and tj == 4):
        posi = 14
    elif(ti == 4 and tj == 4):
        posi = 15
    return posi

board = [('Start','1','1'),('smell','2','1'),('clear','3','1'),('smell','4','1'),
         ('smell','1','2'),('monster','2','2'),('smell & glitter','3','2'),('clear','4','2'),
         ('clear','1','3'),('smell & glitter','2','3'),('gold','3','3'),('glitter','4','3'),
        ('smell','1','4'),('monster','2','4'),('smell & glitter','3','4'),('clear','4','4')]

i,j=1,1
ti,tj=1,1
pos=0
visited=[]

name= str(input("Enter your name: "))
print(name,", your moves are limited into 12 moves.\nThe agent is at (",i,j,":",pos,")")

k=12
while(k>0):
    print("Moves are left: ",--k)
    move = str(input("You can move[up,down,left,right].What's your move:"))
    if(move == 'up'):
        ti = i
        tj = tj+1
        if(tj>4):
            print("Can't go more up")
            tj = tj-1
            pos=position(ti,tj)
            continue
        pos=position(ti,tj)
```

```
elif(move == 'down'):
    ti = i
    tj = tj-1
    if(tj<1):
        print("Can't go more down")
        tj = tj+1
        pos=position(ti,tj)
        continue
    pos=position(ti,tj)

elif(move == 'right'):
    ti = ti+1
    tj = j
    if(ti>4):
        print("Can't go more right")
        ti = ti-1
        pos=position(ti,tj)
        continue
    pos=position(ti,tj)

elif(move == 'left'):
    ti = ti-1
    tj = j
    if(ti<1):
        print("Can't go more left")
        ti = ti+1
        pos=position(ti,tj)
        continue
    pos=position(ti,tj)
```

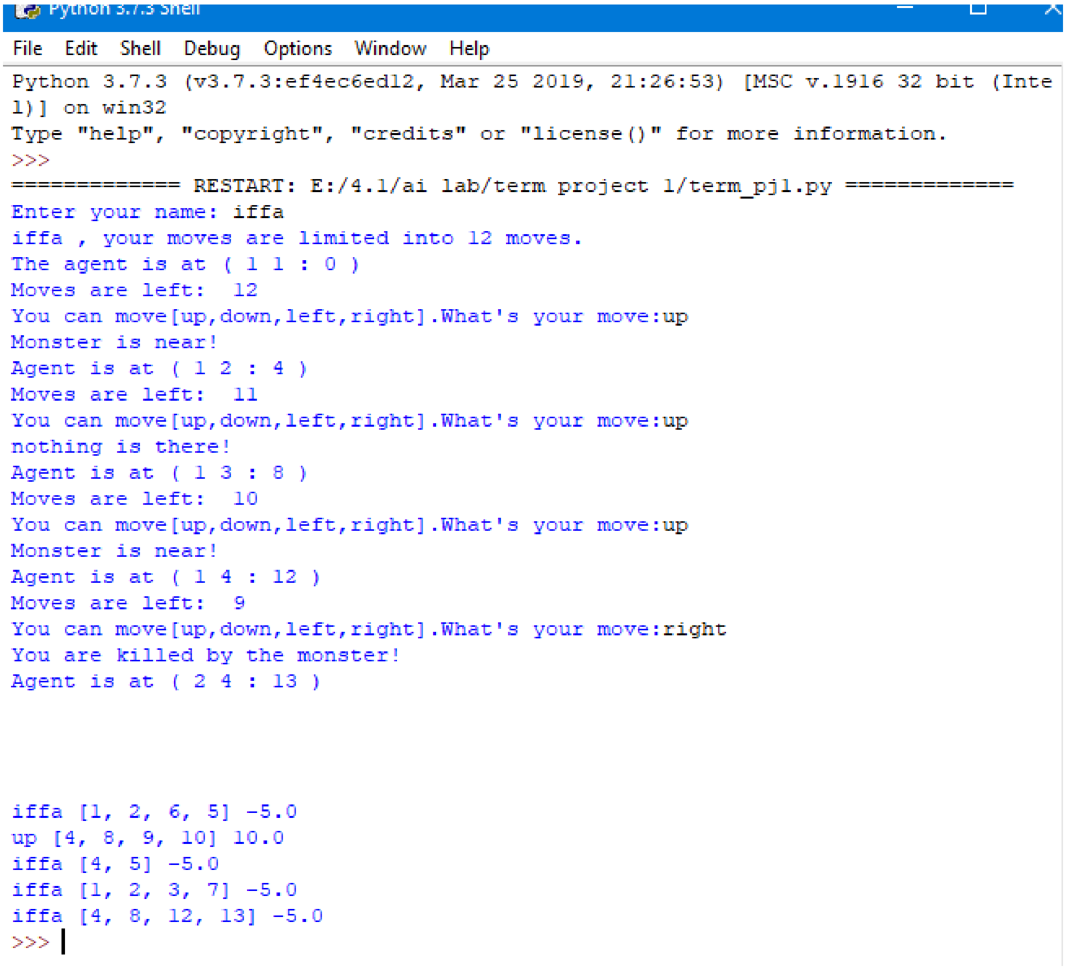
```
else:
    print("No moves.")

if(board[pos][0] == 'smell'):
    i,j=ti,tj
    visited.append(pos)
    print("Monster is near!")
    print("Agent is at (",i,j,":",pos,")")

elif(board[pos][0] == 'smell & glitter'):
    i,j=ti,tj
    visited.append(pos)
    print("Gold is gittering also Monster is near!")
    print("Agent is at (",i,j,":",pos,")")

elif(board[pos][0] == 'clear'):
    i,j=ti,tj
    visited.append(pos)
    print("nothing is there!")
    print("Agent is at (",i,j,":",pos,")")

elif(board[pos][0] == 'monster'):
    i,j=ti,tj
    visited.append(pos)
    score = -5
    print("You are killed by the monster!")
    print("Agent is at (",i,j,":",pos,")")
    break

elif(board[pos][0] == 'glitter'):
    i,j=ti,tj
```

```
        visited.append(pos)
        print("Gold is gittering,it's near!")
        print("Agent is at (",i,j,":",pos,")")

    elif(board[pos][0] == 'gold'):
        i,j=ti,tj
        visited.append(pos)
        print("You find the Gold!!")
        score = 10
        print("Agent is at (",i,j,":",pos,")")
        break

    k-=1

f1=open('score.txt', "a+")
print("\n")
for i in range(1):
        visit=str(visited)
        std=name+"\t"+visit+"\t"+str(score)
        print(std, end="\n", file=f1)
        print("\n")
f1.close

f1=open('score.txt', "r")
for l in f1:
        name, visited, score =l.split("\t")
        print(name, visited, float(score), end="\n")
f1.close
```

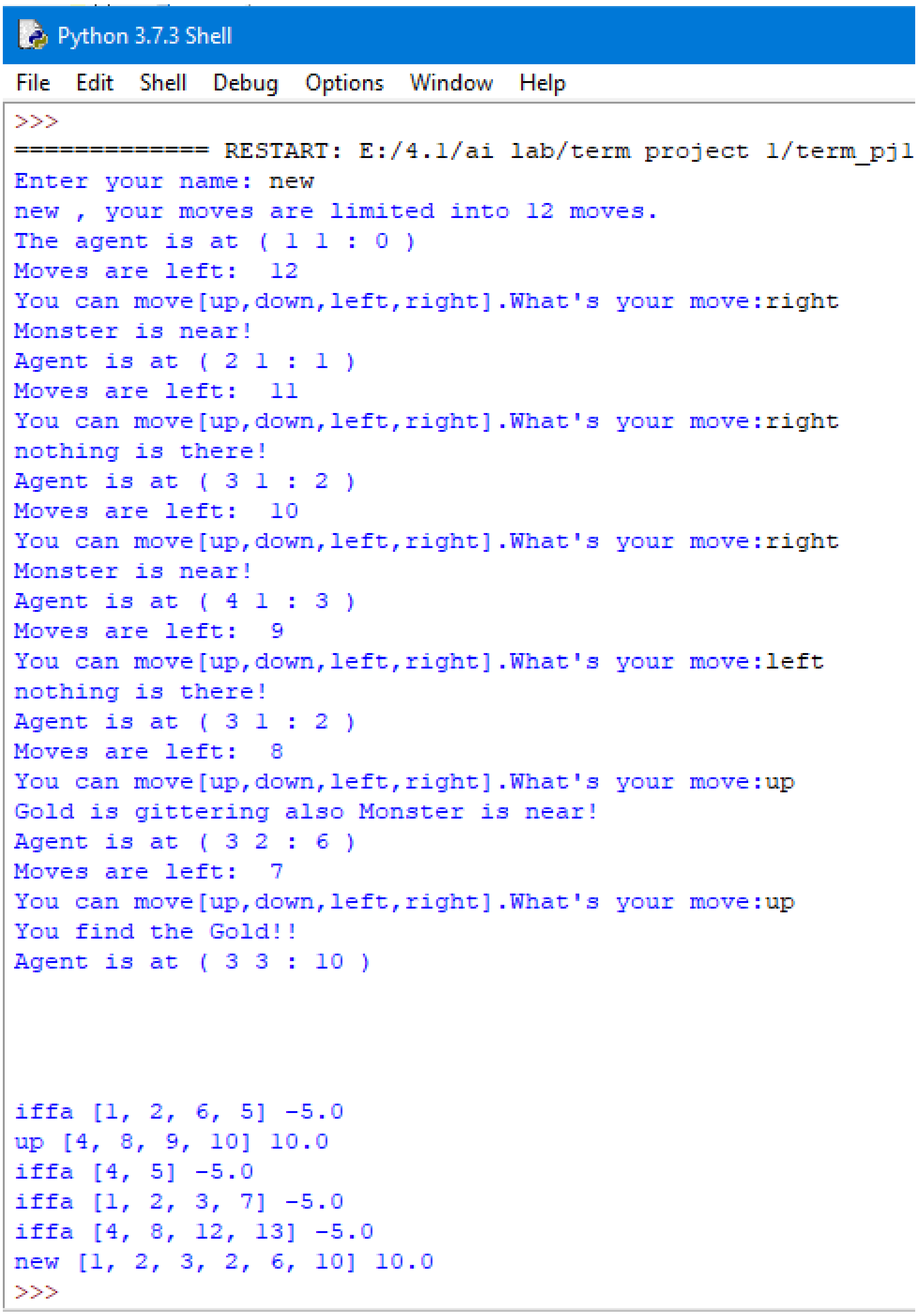
Output:



```
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 21:26:53) [MS
1)] on win32
Type "help", "copyright", "credits" or "license()" for more
>>>
============= RESTART: E:/4.1/ai lab/term project 1/term_pj
Enter your name: iffa
iffa , your moves are limited into 12 moves.
The agent is at ( 1 1 : 0 )
Moves are left:  12
You can move[up,down,left,right].What's your move:up
Monster is near!
Agent is at ( 1 2 : 4 )
Moves are left:  11
You can move[up,down,left,right].What's your move:right
You are killed by the monster!
Agent is at ( 2 2 : 5 )




iffa [1, 2, 6, 5] -5.0
up [4, 8, 9, 10] 10.0
iffa [4, 5] -5.0
>>>
```



```
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 21:26:53) [MSC v.1916 32 bit (Inte
1)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
============= RESTART: E:/4.1/ai lab/term project 1/term_pj1.py =============
Enter your name: iffa
iffa , your moves are limited into 12 moves.
The agent is at ( 1 1 : 0 )
Moves are left:  12
You can move[up,down,left,right].What's your move:up
Monster is near!
Agent is at ( 1 2 : 4 )
Moves are left:  11
You can move[up,down,left,right].What's your move:up
nothing is there!
Agent is at ( 1 3 : 8 )
Moves are left:  10
You can move[up,down,left,right].What's your move:up
Monster is near!
Agent is at ( 1 4 : 12 )
Moves are left:  9
You can move[up,down,left,right].What's your move:right
You are killed by the monster!
Agent is at ( 2 4 : 13 )




iffa [1, 2, 6, 5] -5.0
up [4, 8, 9, 10] 10.0
iffa [4, 5] -5.0
iffa [1, 2, 3, 7] -5.0
iffa [4, 8, 12, 13] -5.0
>>>
```



```
>>>
============= RESTART: E:/4.1/ai lab/term project 1/term_pj1
Enter your name: new
new , your moves are limited into 12 moves.
The agent is at ( 1 1 : 0 )
Moves are left:  12
You can move[up,down,left,right].What's your move:right
Monster is near!
Agent is at ( 2 1 : 1 )
Moves are left:  11
You can move[up,down,left,right].What's your move:right
nothing is there!
Agent is at ( 3 1 : 2 )
Moves are left:  10
You can move[up,down,left,right].What's your move:right
Monster is near!
Agent is at ( 4 1 : 3 )
Moves are left:  9
You can move[up,down,left,right].What's your move:left
nothing is there!
Agent is at ( 3 1 : 2 )
Moves are left:  8
You can move[up,down,left,right].What's your move:up
Gold is gittering also Monster is near!
Agent is at ( 3 2 : 6 )
Moves are left:  7
You can move[up,down,left,right].What's your move:up
You find the Gold!!
Agent is at ( 3 3 : 10 )




iffa [1, 2, 6, 5] -5.0
up [4, 8, 9, 10] 10.0
iffa [4, 5] -5.0
iffa [1, 2, 3, 7] -5.0
iffa [4, 8, 12, 13] -5.0
new [1, 2, 3, 2, 6, 10] 10.0
>>>
```