

Ahsanullah University of Science and Technology



Department of Computer Science and Engineering

Program: Bachelor of Science in Computer Science and Engineering

Course No: CSE 4108

Course Title: Artificial Intelligence Lab

Assignment no:01

Date of Submission: 13.06.2022

Submitted to:

Mr. Md. Siam Ansary
Lecturer, Department of CSE, AUST.

Ms. Tamanna Tabassum
Lecturer, Department of CSE, AUST.

Submitted By,

Name: S. M. Tasnimul Hasan

Student ID: 18.02.04.142

Question 1: Write Python and Prolog codes to find the grandparent(s) of somebody.

Solution:

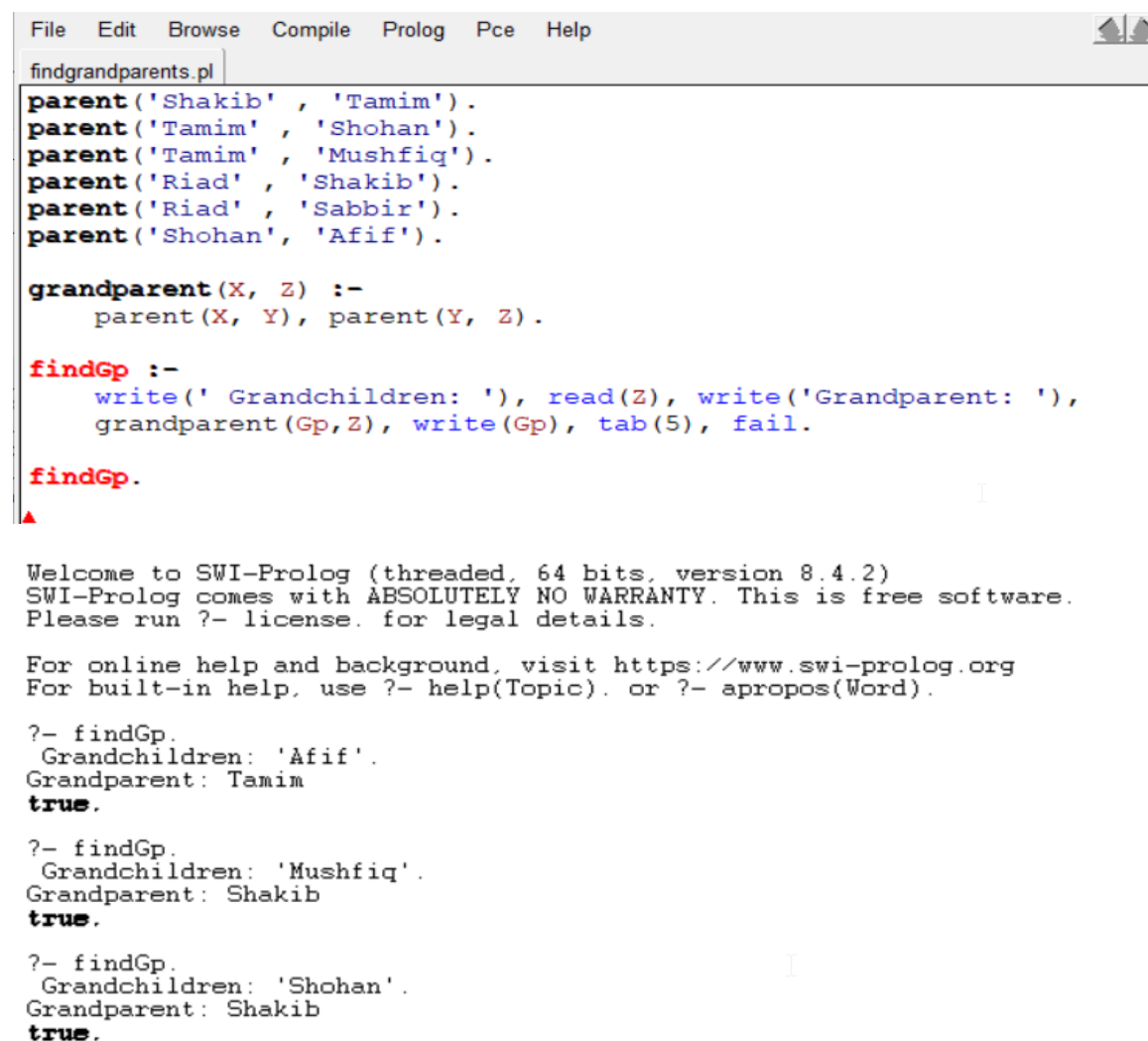
Prolog Code:

```
parent('Shakib' , 'Tamim').  
parent('Tamim' , 'Shohan').  
parent('Tamim' , 'Mushfiq').  
parent('Riad' , 'Shakib').  
parent('Riad' , 'Sabbir').  
parent('Shohan' , 'Afif').
```

```
grandparent(X, Z) :-  
    parent(X, Y), parent(Y, Z).
```

```
findGp :-  
    write(' Grandchildren: '), read(Z), write('Grandparent: '),  
    grandparent(Gp,Z), write(Gp), tab(5), fail.
```

findGp.



The screenshot shows a Prolog IDE window titled 'findgrandparents.pl'. The code in the editor is as follows:

```
parent('Shakib' , 'Tamim').  
parent('Tamim' , 'Shohan').  
parent('Tamim' , 'Mushfiq').  
parent('Riad' , 'Shakib').  
parent('Riad' , 'Sabbir').  
parent('Shohan' , 'Afif').  
  
grandparent(X, Z) :-  
    parent(X, Y), parent(Y, Z).  
  
findGp :-  
    write(' Grandchildren: '), read(Z), write('Grandparent: '),  
    grandparent(Gp,Z), write(Gp), tab(5), fail.  
  
findGp.
```

The output window shows the following text:

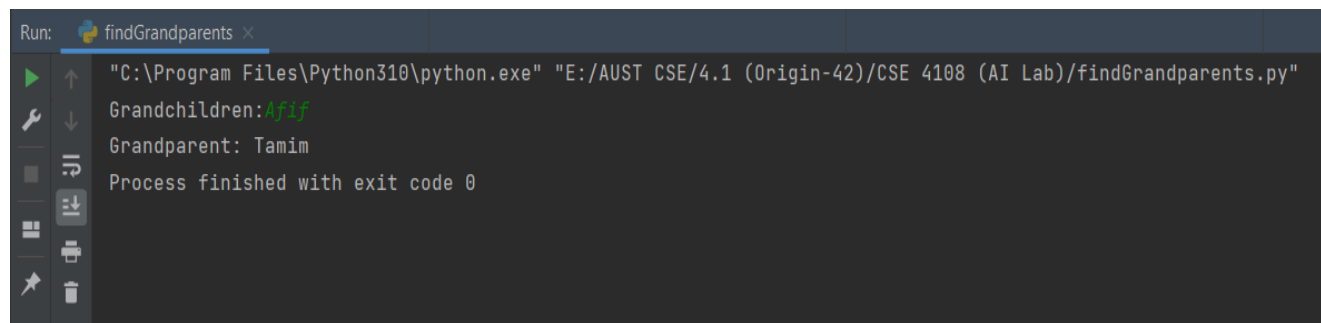
```
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.2)  
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.  
Please run ?- license. for legal details.  
  
For online help and background, visit https://www.swi-prolog.org  
For built-in help, use ?- help(Topic). or ?- apropos(Word).  
  
?- findGp.  
Grandchildren: 'Afif'.  
Grandparent: Tamim  
true.  
  
?- findGp.  
Grandchildren: 'Mushfiq'.  
Grandparent: Shakib  
true.  
  
?- findGp.  
Grandchildren: 'Shohan'.  
Grandparent: Shakib  
true.
```

Python Code:

```
tupleList1=[('parent', 'Shakib' , 'Tamim'),
             ('parent', 'Tamim' , 'Shohan'),
             ('parent', 'Tamim' , 'Mushfiq'),
             ('parent', 'Riad' , 'Shakib'),
             ('parent', 'Riad' , 'Sabbir'),
             ('parent', 'Shohan', 'Afif')]
X=str(input("Grandchildren:"))
print('Grandparent:', end=' ')
i,j=0,0
for i in range(6):
    if ((tupleList1[i][0] == 'parent')&( tupleList1[i][2] == X)):
        for j in range(6):
            if ((tupleList1[j][0] == 'parent') & ( tupleList1[i][1] == tupleList1[j][2]]):
                print(tupleList1[j][1], end=' ')
```



```
findGrandparents.py x
1 tupleList1 = [('parent', 'Shakib', 'Tamim'),
2               ('parent', 'Tamim', 'Shohan'),
3               ('parent', 'Tamim', 'Mushfiq'),
4               ('parent', 'Riad', 'Shakib'),
5               ('parent', 'Riad', 'Sabbir'),
6               ('parent', 'Shohan', 'Afif')]
7 X = str(input("Grandchildren:"))
8 print('Grandparent:', end=' ')
9 i, j = 0, 0
10 for i in range(6):
11     if (tupleList1[i][0] == 'parent') & (tupleList1[i][2] == X):
12         for j in range(6):
13             if (tupleList1[j][0] == 'parent') & (tupleList1[i][1] == tupleList1[j][2]):
14                 print(tupleList1[j][1], end=' ')
15
```



```
Run: findGrandParents x
"C:\Program Files\Python310\python.exe" "E:/AUST CSE/4.1 (Origin-42)/CSE 4108 (AI Lab)/findGrandparents.py"
Grandchildren: Afif
Grandparent: Tamim
Process finished with exit code 0
```

Question 2: Enrich the KB with ‘brother’, ‘sister’, ‘uncle’ and ‘aunt’ rules in Python and Prolog.

Solution:

Prolog Code:

```
male('Shakib').
male('Tamim').
male('Liton').
male('Shohan').
male('Afif').
female('Zeniah').
female('Tanjin').
female('Ayesha').
```

```
parent('Shakib' , 'Tamim').
parent('Tamim' , 'Liton').
parent('Tamim' , 'Shohan').
parent('Tamim' , 'Ayesha').
parent('Afif' , 'Shakib').
parent('Shohan' , 'Zeniah').
parent('Shohan' , 'Riad').
parent('Shohan' , 'Tanjin').
parent('Riad' , 'Mustafiz').
```

brother(Y, Z) :-

```
    parent(X, Y), parent(X, Z), male(Y), not(Y=Z).
```

sister(Y, Z) :-

```
    parent(X, Y), parent(X, Z), female(Y), not(Y=Z).
```

uncle(Y, U) :-

```
    parent(X, Y), parent(X, Z), parent(Z,U), male(Y), not(Y=U), not(Y=Z).
```

aunt(Y, U) :-

```
    parent(X, Y), parent(X, Z), parent(Z,U), female(Y), not(Y=U), not(Y=Z).
```

findBrother :-

```
    write(' Siblings: '), read(Z), write('Brother: '),
    brother(Brother,Z), write(Brother), tab(5),fail.
```

findBrother.

findSister :-

```
    write(' Siblings: '), read(Z), write('Sister: '),
    sister(Sister,Z), write(Sister), tab(5),fail.
```

findSister.

findUncle:-

```
    write(' Niece/Nephew: '), read(Z), write('Uncle: '),
```

```

uncle(Uncle,Z), write(Uncle), tab(5),fail.
findUncle.
findAunt:-
    write(' Niece/Nephew: '), read(Z), write('Aunt: '),
    aunt(Aunt,Z), write(Aunt), tab(5),fail.
findAunt.

```

```

File Edit Browse Compile Prolog Pce Help
assignment-q2.pl
male('Shakib').
male('Tamim').
male('Liton').
male('Shohan').
male('Afif').
female('Zenja').
female('Tanjin').
female('Ayesha').

parent('Shakib' , 'Tamim').
parent('Tamim' , 'Liton').
parent('Tamim' , 'Shohan').
parent('Tamim' , 'Ayesha').
parent('Afif' , 'Shakib').
parent('Shohan' , 'Zenja').
parent('Shohan' , 'Riad').
parent('Shohan' , 'Tanjin').
parent('Riad' , 'Mustafiz').

brother(Y, Z) :-
    parent(X, Y), parent(X, Z), male(Y), not(Y=Z).
sister(Y, Z) :-
    parent(X, Y), parent(X, Z), female(Y), not(Y=Z).
uncle(Y, U) :-
    parent(X, Y), parent(X, Z), parent(Z,U), male(Y), not(Y=U), not(Y=Z).
aunt(Y, U) :-
    parent(X, Y), parent(X, Z), parent(Z,U), female(Y), not(Y=U), not(Y=Z).

findBrother :-
    write(' Siblings: '), read(Z), write('Brother: '),
    brother(Brother,Z), write(Brother), tab(5),fail.
findBrother.

findSister :-
    write(' Siblings: '), read(Z), write('Sister: '),
    sister(Sister,Z), write(Sister), tab(5),fail.
findSister.

findUncle:-
    write(' Niece/Nephew: '), read(Z), write('Uncle: '),
    uncle(Uncle,Z), write(Uncle), tab(5),fail.
findUncle.

findAunt:-
    write(' Niece/Nephew: '), read(Z), write('Aunt: '),
    aunt(Aunt,Z), write(Aunt), tab(5),fail.
findAunt.

```

File Edit Settings Run Debug Help

Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.2)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run `?- license.` for legal details.

For online help and background, visit <https://www.swi-prolog.org>
For built-in help, use `?- help(Topic).` or `?- apropos(Word).`

```
?- findBrother.  
  Siblings: 'Liton'.  
Brother: Shohan  
true.
```

```
?- findSister.  
  Siblings: 'Riad'.  
Sister: Zenia      Tanjin  
true.
```

```
?- findUncle.  
  Niece/Nephew: 'Zenias'.  
Uncle: Liton  
true.
```

```
?- findAunt.  
  Niece/Nephew: 'Mustafiz'.  
Aunt: Zenia      Tanjin  
true.
```

Python Code:

```
tupleList1=[('parent', 'Shakib', 'Tamim'),  
            ('parent', 'Tamim', 'Liton'),  
            ('parent', 'Tamim', 'Shohan'),  
            ('parent', 'Tamim', 'Ayesha'),  
            ('parent', 'Afif', 'Shakib'),  
            ('parent', 'Shohan', 'Zenja'),  
            ('parent', 'Shohan', 'Riad'),  
            ('parent', 'Shohan', 'Tanjin'),  
            ('parent', 'Riad', 'Mustafiz')]
```

```
tupleList2=[('gender', 'Shakib', 'male'),  
            ('gender', 'Tamim', 'male'),  
            ('gender', 'Liton', 'male'),  
            ('gender', 'Shohan', 'male'),  
            ('gender', 'Ayesha', 'female'),  
            ('gender', 'Afif', 'male'),  
            ('gender', 'Zenja', 'female'),  
            ('gender', 'Riad', 'male'),  
            ('gender', 'Tanjin', 'female'),  
            ('gender', 'Mustafiz', 'male')]
```

```
X=str(input("Sibling:"))  
print('Brother:', end=' ' )  
i,j,k=0,0,0  
for i in range(9):  
    if ((tupleList1[i][0] == 'parent') & ( tupleList1[i][2] == X)):  
        for j in range(9):  
            if ((tupleList1[j][0] == 'parent') & ( tupleList1[i][1] == tupleList1[j][1]) & ( tupleList1[j][2]  
!= tupleList1[i][2])) :  
                for k in range (10) :  
                    if((tupleList1[j][2] == tupleList2[k][1]) & (tupleList2[k][2] == 'male')) :  
                        print(tupleList1[j][2], end=' ' )  
print("\n")  
X=str(input("Sibling:"))  
print('Sister:', end=' ' )  
i,j,k=0,0,0  
for i in range(9):  
    if ((tupleList1[i][0] == 'parent') & ( tupleList1[i][2] == X)):  
        for j in range(9):  
            if ((tupleList1[j][0] == 'parent') & ( tupleList1[i][1] == tupleList1[j][1]) & ( tupleList1[j][2]  
!= tupleList1[i][2])) :  
                for k in range (10) :  
                    if((tupleList1[j][2]== tupleList2[k][1]) & (tupleList2[k][2] == 'female')) :  
                        print(tupleList1[j][2], end=' ' )  
print("\n")  
X=str(input("Niece/Nephew:"))  
print('Uncle:', end=' ' )
```

```

i,j,k,l=0,0,0,0
for i in range(9):
    if ((tupleList1[i][0] == 'parent') & ( tupleList1[i][2] == X)):
        for j in range(9):
            if ((tupleList1[j][0] == 'parent') & ( tupleList1[i][1] == tupleList1[j][2]) & ( tupleList1[j][2]
!= tupleList1[i][2])) :
                for k in range (9) :
                    if((tupleList1[j][0] == 'parent') & (tupleList1[j][1] == tupleList1[k][1]) &
(tupleList1[i][1] != tupleList1[k][2])):
                        for l in range (10) :
                            if((tupleList1[k][2] == tupleList2[l][1]) & (tupleList2[l][2] == 'male')) :
                                print(tupleList1[k][2], end=' ')
print("\n")
X=str(input("Niece/Nephew:"))
print(Aunt:', end=' ')
i,j,k,l=0,0,0,0
for i in range(9):
    if ((tupleList1[i][0] == 'parent') & ( tupleList1[i][2] == X)):
        for j in range(9):
            if ((tupleList1[j][0] == 'parent') & ( tupleList1[i][1] == tupleList1[j][2]) & ( tupleList1[j][2]
!= tupleList1[i][2])) :
                for k in range (9) :
                    if((tupleList1[j][0] == 'parent') & (tupleList1[j][1] == tupleList1[k][1]) &
(tupleList1[i][1] != tupleList1[k][2])):
                        for l in range (10) :
                            if((tupleList1[k][2] == tupleList2[l][1]) & (tupleList2[l][2] == 'female')) :
                                print(tupleList1[k][2], end=' ')
print("\n")

```



```

assignment-q2.py
1 tupleList1=[('parent', 'Shakib', 'Tamim'),
2             ('parent', 'Tamim', 'Liton'),
3             ('parent', 'Tamim', 'Shohan'),
4             ('parent', 'Tamim', 'Ayesha'),
5             ('parent', 'Afif', 'Shakib'),
6             ('parent', 'Shohan', 'Zeniah'),
7             ('parent', 'Shohan', 'Riad'),
8             ('parent', 'Shohan', 'Tanjin'),
9             ('parent', 'Riad', 'Mustafiz')]
10 tupleList2=[('gender', 'Shakib', 'male'),
11             ('gender', 'Tamim', 'male'),
12             ('gender', 'Liton', 'male'),
13             ('gender', 'Shohan', 'male'),
14             ('gender', 'Ayesha', 'female'),
15             ('gender', 'Afif', 'male'),
16             ('gender', 'Zeniah', 'female'),
17             ('gender', 'Riad', 'male'),
18             ('gender', 'Tanjin', 'female'),
19             ('gender', 'Mustafiz', 'male')]
20
21
22 X=str(input("Sibling:"))
23 print('Brother:', end=' ')
24 i,j,k=0,0,0
25 for i in range(9):
26     if ((tupleList1[i][0] == 'parent') & (tupleList1[i][2] == X)):
27         for j in range(9):
28             if ((tupleList1[j][0] == 'parent') & (tupleList1[i][1] == tupleList1[j][1] & (tupleList1[j][2] != tupleList1[i][2]))):
29                 for k in range_(10):
30                     if((tupleList1[j][2] == tupleList2[k][1]) & (tupleList2[k][2] == 'male')):
31                         print(tupleList1[j][2], end=' ')
32
33 print("\n")
34 X=str(input("Sibling:"))
35 print('Sister:', end=' ')
36 i,j,k=0,0,0
37 for i in range(9):
38     if ((tupleList1[i][0] == 'parent') & (tupleList1[i][2] == X)):
39         for j in range(9):
40             if ((tupleList1[j][0] == 'parent') & (tupleList1[i][1] == tupleList1[j][1] & (tupleList1[j][2] != tupleList1[i][2]))):
41                 for k in range_(10):
42                     if((tupleList1[j][2] == tupleList2[k][1]) & (tupleList2[k][2] == 'female')):
43                         print(tupleList1[j][2], end=' ')
44
45 print("\n")
46
47 X=str(input("Niece/Nephew:"))
48 print('Uncle:', end=' ')
49 i,j,k,l=0,0,0,0
50 for i in range(9):
51     if ((tupleList1[i][0] == 'parent') & (tupleList1[i][2] == X)):
52         for j in range(9):
53             if ((tupleList1[j][0] == 'parent') & (tupleList1[i][1] == tupleList1[j][2] & (tupleList1[j][2] != tupleList1[i][2]))):
54                 for k in range_(9):
55                     if((tupleList1[j][0] == 'parent') & (tupleList1[j][1] == tupleList1[k][1]) & (tupleList1[i][1] != tupleList1[k][2])):
56                         for l in range_(10):
57                             if((tupleList1[k][2] == tupleList2[l][1]) & (tupleList2[l][2] == 'male')):
58                                 print(tupleList1[k][2], end=' ')
59
60 print("\n")
61 X=str(input("Niece/Nephew:"))
62 print('Aunt:', end=' ')
63 i,j,k,l=0,0,0,0
64 for i in range(9):
65     if ((tupleList1[i][0] == 'parent') & (tupleList1[i][2] == X)):
66         for j in range(9):
67             if ((tupleList1[j][0] == 'parent') & (tupleList1[i][1] == tupleList1[j][2] & (tupleList1[j][2] != tupleList1[i][2]))):
68                 for k in range_(9):
69                     if((tupleList1[j][0] == 'parent') & (tupleList1[j][1] == tupleList1[k][1]) & (tupleList1[i][1] != tupleList1[k][2])):
70                         for l in range_(10):
71                             if((tupleList1[k][2] == tupleList2[l][1]) & (tupleList2[l][2] == 'female')):
72                                 print(tupleList1[k][2], end=' ')
73
74 print("\n")

```

Run: assignment-q2 | x

"C:\Program Files\Python310\python.exe" "E:/AUST CSE/4.1 (Origin-42)/CSE 4108 (AI Lab)/assignment-q2.py"

Sibling: *Liton*

Brother: Shohan

Sibling: *Lion*

Sister: Zenia Tanjin

Niece/Nephew: *Zenia*

Uncle: Liton

Niece/Nephew: *Mustafiz*

Aunt: Zenia Tanjin

Process finished with exit code 0