

# 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: 25/12/2021

Submitted to:

Mr. Faisal Muhammad Shah  
Associate Professor, Department of CSE, AUST.

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

Submitted By:

Name: Mahin Opu

Student ID: 17.02.04.006

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

Solution:

Prolog Code:

%Someone's Grandparent

parent('Kofil', 'Mahin').

parent('Kofil', 'Karin').

parent('Kofil', 'Ornob').

parent('Kadir', 'Kofil').

grandparent(G,Z) :- parent(X,Z), parent(G,X).

searchGrandparent :- write('Enter Grandchild Name: '), read(INPUT), write('grandparent:'),  
grandparent(GP,INPUT), write(GP).

searchGrandparent.

A screenshot of a Prolog IDE window titled '170204006\_Q\_01.pl'. The window has a menu bar with 'File', 'Edit', 'Browse', 'Compile', 'Prolog', 'Pce', and 'Help'. Below the menu bar, there are two tabs: '170204006\_Q\_02.pl' (active) and '170204006\_Q\_01.pl'. The main text area contains the following Prolog code:

```
%Someone's Grandparent
parent('Kofil', 'Mahin').
parent('Kofil', 'Karin').
parent('Kofil', 'Ornob').
parent('Kadir', 'Kofil').

grandparent(G,Z) :- parent(X,Z), parent(G,X).

searchGrandparent :- write('Enter Grandchild Name: '), read(INPUT), write('grandparent:'),
grandparent(GP,INPUT), write(GP).
searchGrandparent.
```

File Edit Settings Run Debug Help

Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)  
 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).`

`?-``% e:/ai assignments/170204006_q_01 compiled 0.00 sec, -2 clauses``?- searchGrandparent.`

Enter Grandchild Name: 'Mahin'.

grandparent:Kadir

`true .``?- searchGrandparent.`

Enter Grandchild Name: 'Karin'.

grandparent:Kadir

`true .``?- searchGrandparent.`

Enter Grandchild Name: 'Ornob'.

grandparent:Kadir

`true .``?- █`

### Python Code:

```
parentlist=[
    ('parent','Kofil','Mahin'),
    ('parent','Kofil','Karin'),
    ('parent','Kofil','Ornob'),
    ('parent','Kadir','Kofil')
]
X=str(input("Enter grandchildren: "))
print("Grandparent name: ")

i,j=0,0
while(i<=3):
    if((parentlist[i][0]=='parent')&(parentlist[i][2]==X)):
        for j in range(4):
            if((parentlist[j][0]=='parent')&(parentlist[i][1]==parentlist[j][2]]):
                print(parentlist[j][1]," ")
            i=i+1
```

170204006\_Python\_01.py - E:\AI Assignments\170204006\_Python\_01.py (3.8.8)

File Edit Format Run Options Window Help

```
1 parentlist=[
2     ('parent','Kofil','Mahin'),
3     ('parent','Kofil','Karin'),
4     ('parent','Kofil','Ornob'),
5     ('parent','Kadir','Kofil')
6 ]
7 X=str(input("Enter grandchildren: "))
8 print("Grandparent name: ")
9
10 i,j=0,0
11 while(i<=3):
12     if((parentlist[i][0]=='parent') & (parentlist[i][2]==X)):
13         for j in range(4):
14             if((parentlist[j][0]=='parent') & (parentlist[i][1]==parentlist[j][2])):
15                 print(parentlist[j][1]," ")
16         i=i+1
17
```

IDLE Shell 3.8.8

File Edit Shell Debug Options Window Help

Python 3.8.8 (default, Apr 13 2021, 15:08:03) [MSC v.1916 64 bit (AMD64)] on win32  
Type "help", "copyright", "credits" or "license()" for more information.

>>>

===== RESTART: E:\AI Assignments\170204006\_Python\_01.py =====

Enter grandchildren: Mahin

Grandparent name:

Kadir

>>>

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

Prolog.

Solution:

Prolog Code:

```
parent('Kofil', 'Mahin').
parent('Kofil', 'Karin').
parent('Kofil', 'Ornob').
parent('Kadir', 'Kofil').
parent('Kadir', 'Bachhu').
parent('Kadir', 'Roshena').
```

```
boy('Kofil').
boy('Mahin').
boy('Ornob').
boy('Bachhu').
girl('Karin').
girl('Nova').
girl('Roshena').
```

```
uncle(U,M):-parent(Y,M),parent(Z,Y),parent(Z,U),boy(U), not(Y=U).
aunty(A,M):-parent(Y,M),parent(Z,Y),parent(Z,A),girl(A),not(Y=A).
brother(B,X):- parent(Y,X),parent(Y,B),boy(B),not(X=B).
sister(S,X):- parent(Y,X),parent(Y,S),girl(S),not(X=S).
```

```
searchUncle :- write(' Enter Nephew Name: '), read(INPUT), write('Uncle:'),
               uncle(UNCLE,INPUT ), write(UNCLE), tab(5), fail.
searchUncle.
```

```
searchAunty :- write(' Enter Nephew name: '), read(INPUT), write('AUNTY is : '),
               aunty(AUNTY,INPUT ), write(AUNTY), tab(5), fail.
searchAunty.
```

```
searchBrother:- write(' Enter sibling name: '), read(INPUT), write('brother: '),
                brother(BROTHER,INPUT ), write(BROTHER), tab(5), fail.
searchBrother.
searchSister:- write(' Enter sibling name: '), read(INPUT), write('sister: '),
```

sister(SISTER,INPUT ), write(SISTER), tab(5), fail.  
searchSister.

```
170204006_Q_02.pl
File Edit Browse Compile Prolog Pce Help
170204006_Q_02.pl
parent('Kofil', 'Mahin').
parent('Kofil', 'Karin').
parent('Kofil', 'Ornob').
parent('Kadir', 'Kofil').
parent('Kadir', 'Bachhu').
parent('Kadir', 'Roshena').

boy('Kofil').
boy('Mahin').
boy('Ornob').
boy('Bachhu').
girl('Karin').
girl('Nova').
girl('Roshena').

uncle(U,M):-parent(Y,M),parent(Z,Y),parent(Z,U),boy(U), not(Y=U).
aunty(A,M):-parent(Y,M),parent(Z,Y),parent(Z,A),girl(A),not(Y=A).
brother(B,X):- parent(Y,X),parent(Y,B),boy(B),not(X=B).
sister(S,X):- parent(Y,X),parent(Y,S),girl(S),not(X=S).

searchUncle :- write(' Enter Nephew Name: '), read(INPUT), write('Uncle:'),
               uncle(UNCLE,INPUT ), write(UNCLE), tab(5), fail.
searchUncle.

searchAunty :- write(' Enter Nephew name: '), read(INPUT), write('AUNTY is : '),
               aunty(AUNTY,INPUT ), write(AUNTY), tab(5), fail.
searchAunty.

searchBrother:- write(' Enter sibling name: '), read(INPUT), write('brother: '),
                brother(BROTHER,INPUT ), write(BROTHER), tab(5), fail.
searchBrother.
searchSister:- write(' Enter sibling name: '), read(INPUT), write('sister: '),
               sister(SISTER,INPUT ), write(SISTER), tab(5), fail.
searchSister.
```

```
File Edit Settings Run Debug Help
Welcome to SWI-Prolog (threaded, 64 bits, version 8.4.1)
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).

?-
% e:/ai assignments/170204006_q_02 compiled 0.00 sec, -2 clauses
?- searchBrother.
   Enter sibling name: 'Ornob'.
  brother: Mahin
true.

?- searchSister.
   Enter sibling name: 'Mahin'.
  sister: Karin
true.

?- searchAunty.
   Enter Nephew name: 'Mahin'.
  AUNTY is : Roshena
true.

?- searchUncle.
   Enter Nephew Name: 'Mahin'.
  Uncle: Bachhu
true.

?- ■
```

## Python Code:

```
ParentList = [('parent', 'Kofil', 'Mahin'),
              ('parent', 'Kofil', 'Karin'),
              ('parent', 'Kofil', 'Ornob'),
              ('parent', 'Kadir', 'Kofil'),
              ('parent', 'Kadir', 'Bachhu'),
              ('parent', 'Kadir', 'Roshena'),

              ]
boy = ['Kadir', 'Kofil', 'Mahin', 'Bachhu']
girl = ['Karin', 'Roshena']
grandparent = []
parent = []
brother = []
sister = []
uncle = []
aunt = []
```

```

grandchildren = input("ENTER Name:")
for i in range(6):
    if (ParentList[i][0] == 'parent') and (ParentList[i][2] == grandchildren):
        parent.append(ParentList[i][1])
        for j in range(6):
            if (ParentList[j][0] == 'parent') and (ParentList[i][1] == ParentList[j][2]):
                if (ParentList[j][1] not in grandparent and (ParentList[j][1] in boy)):
                    grandparent.append(ParentList[j][1])
                    for k in range(6):
                        if (ParentList[k][0] == 'parent') and (j != k) and (ParentList[j][1]
== ParentList[k][1]):
                            if (ParentList[k][2] not in uncle and (ParentList[k][2] in boy)):
                                uncle.append(ParentList[k][2])
                                for l in range(6):
                                    if (ParentList[l][0] == 'parent') and (j != l) and
(ParentList[k][2] == ParentList[l][2]):
                                        if (ParentList[l][2] not in aunt and (ParentList[l][2] in girl)):
                                            aunt.append(ParentList[l][2])

                                elif (ParentList[j][0] == 'parent') and (i != j) and (ParentList[i][1] ==
ParentList[j][1]):
                                    if (ParentList[j][2] not in brother and (ParentList[j][2] in boy)):
                                        brother.append(ParentList[j][2])
                                    elif (ParentList[j][2] not in sister and (ParentList[j][2] in girl)):
                                        sister.append(ParentList[j][2])
print("BROTHER :", end=' ')
print(*brother, sep=', ')
print("SISTER :", end=' ')
print(*sister, sep=', ')
print("UNCLE :", end=' ')
print(*uncle, sep=', ')
print("AUNTY :", end=' ')
print(*aunt, sep=', ')

```



```
170204006_Task_02.py - E:\AI Assignments\170204006_Task_02.py (3.8.8)
File Edit Format Run Options Window Help
ParentList = [(['parent', 'Kofil', 'Mahin'),
                ('parent', 'Kofil', 'Karin'),
                ('parent', 'Kofil', 'Ornob'),
                ('parent', 'Kadir', 'Kofil'),
                ('parent', 'Kadir', 'Bachhu'),
                ('parent', 'Kadir', 'Roshena'),
                ]
boy = ['Kadir', 'Kofil', 'Mahin', 'Bachhu']
girl = ['Karin', 'Roshena']
grandparent = []
parent = []
brother = []
sister = []
uncle = []
aunt = []
grandchildren = input("ENTER Name:")
for i in range(6):
    if (ParentList[i][0] == 'parent') and (ParentList[i][2] == grandchildren):
        parent.append(ParentList[i][1])
        for j in range(6):
            if (ParentList[j][0] == 'parent') and (ParentList[i][1] == ParentList[j][2]):
                if (ParentList[j][1] not in grandparent and (ParentList[j][1] in boy)):
                    grandparent.append(ParentList[j][1])
                    for k in range(6):
                        if (ParentList[k][0] == 'parent') and (j != k) and (ParentList[j][1] == ParentList[k][1]):
                            if (ParentList[k][2] not in uncle and (ParentList[k][2] in boy)):
                                uncle.append(ParentList[k][2])
                                for l in range(6):
                                    if (ParentList[l][0] == 'parent') and (j != l) and (ParentList[k][2] == ParentList[l][2]):
                                        if (ParentList[l][2] not in aunt and (ParentList[l][2] in girl)):
                                            aunt.append(ParentList[l][2])
                    elif (ParentList[j][0] == 'parent') and (i != j) and (ParentList[i][1] == ParentList[j][1]):
                        if (ParentList[j][2] not in brother and (ParentList[j][2] in boy)):
                            brother.append(ParentList[j][2])
                        elif (ParentList[j][2] not in sister and (ParentList[j][2] in girl)):
                            sister.append(ParentList[j][2])
print("BROTHER :", end=' ')
print(*brother, sep=', ')
print("SISTER :", end=' ')
print(*sister, sep=', ')
print("UNCLE :", end=' ')
print(*uncle, sep=', ')
print("AUNTY :", end=' ')
print(*aunt, sep=', ')
170204006_Task_02.py (3.8.8)
```

IDLE Shell 3.8.8

— □ ×

File Edit Shell Debug Options Window Help

Python 3.8.8 (default, Apr 13 2021, 15:08:03) [MSC v.1916 64 bit (AMD64)] on win32

Type "help", "copyright", "credits" or "license()" for more information.

>>>

===== RESTART: E:\AI Assignments\170204006\_Task\_02.py =====

ENTER Name:Mahin

BROTHER : Ornob

SISTER : Karin

UNCLE : Bachhu

AUNTY : Roshena

>>> |