
# Ahsanullah University of Science & Technology

**Department of Computer Science & Engineering**

**Course No.** : CSE 4108
**Course Name** : Artificial Intelligence Lab
**Assignment No.** : 01

**Submitted To :**

Md. Siam Ansary
Department of CSE, AUST

Tonmoy Hossain
Department of CSE, AUST

**Submitted By:**

Name : Sanjida Akter Ishita
ID No. : 170204089
Session : Fall 2020
Section : B
Lab group : B2

**Date of Submission:** July 7, 2021

## QUESTION 3:

**Modify the Python and Prolog codes demonstrated above to find the grandparents of somebody.**

## ANSWER TO THE QUESTION NO 3:

Here is the modification of code in python and prolog to find the grandparents of somebody.

### Code in prolog:

```
parent('Dipty','Ikra').
parent('Ikra','Imtiaz').
parent('Ikra','Imtiaz').
parent('Karim','Dipty').
grandchild(Z ,X):-
parent(Y, X), parent(Z, Y).
findGp:-
write('Grandchild: '), read(Gp), write('Grandparent: '),
grandchild(X, Gp), write(X), tab(5), fail.
findGp.
```

### Input and output:

```
% g:/4.1/lab/ai lab/new compiled 0.00 sec, 0 clauses
?-
|    findGp.
Grandchild: 'Imtiaz'.
Grandparent: Dipty      Dipty
true.
```

## Code in Python:

```
tupplelist1 = [
    ('parent','Dipty','Ikra'),
    ('parent','Ikra','Imtiaz'),
    ('parent','Ikra','Ratan'),
    ('parent','Karim','Dipty')
    ]
X = str(input("Grandchild:"))
print("Grandparent:",end='')
i,j=0,0
while(i<3):
    if((tupplelist1[i][0]=='parent') & (tupplelist1[i][2]==X)):
        for j in range(4):
            if((tupplelist1[j][0]=='parent') & (tupplelist1[i][1]==tupplelist1[j][2])):
                print(tupplelist1[j][1], end="")

    i=i+1
```

## Input And Output:

```
================== RESTART: G:/4.1/LAB/AI Lab/Ques3inPython.py ==================
Grandchild:Imtiaz
Grandparent:Dipty
>>>
```

## QUESTION 4:

**Enrich the KB demonstrated above with 'brother', 'sister', 'uncle' and 'aunt' rules in Python and Prolog.**

## ANSWER TO THE QUESTION NO 4:

Here are our prolog and python code with brother, sister, uncle and aunt rules and show how to find these relations.

### Prolog Code:

```
male('Asif'). male('Karim'). male('Joy'). male('Jahid').
male('Rahim'). male('Uday'). male('Anik'). male('Ujjal').
female('Maisha'). female('Dipty'). female('Lipy').
parent('Hasib' ,'Rakib').
parent('Rakib' ,'Sohel').
parent('Rakib' ,'Rebeka').
parent('Rashid' ,'Hasib').
parent('Maisha' ,'Dipty').
parent('Maisha' ,'Asif').
parent('Karim' ,'Maisha').
parent('Lipi' ,'Joy').
parent('Lipi' ,'Jahid').
parent('Karim' ,'Lipi').
parent('Karim' ,'Ujjal').
parent('Rahim' ,'Jahid').
parent('Rahim' ,'Joy').
parent('Uday' ,'Anik').
parent('Uday' ,'Anik').
brother(X, Z):-
parent(Y ,X), parent(Y, Z), male(Z), not(X=Z).
findBro:-
write('Name: '), read(Bro), write('Brother: '),
brother(Bro, X), write(X), tab(5), fail.
findBro.
sister(X, Z):-
parent(Y ,X), parent(Y, Z), female(Z), not(X=Z).
findSis:-
write('Name: '), read(Sis), write('Sister: '),
sister(Sis, X), write(X), tab(5), fail.
findSis.
uncle(X, Z):-
parent(Y ,X), parent(D ,Y), parent(D ,Z),
male(Z), not(Y=Z).
```

```
findUnc:-
write('Name: '), read(Unc), write('Uncle: '),
uncle(Unc, X), write(X), tab(5), fail.
findUnc.
aunt(X, Z):-
parent(Y ,X), parent(N ,Y), parent(N ,Z),
female(Z), not(Y=Z).
findAunt:-
write('Name: '), read(Aun), write('Aunt: '),
aunt(Aun, X), write(X), tab(5), fail.
findAunt.
```

## Input And Output

```
% g:/4.1/lab/ai lab/codeques4 compiled 0.00 sec, 0 clauses
?-
|    findBro.
Name: 'Jahid'.
Brother: Joy     Joy
true.

?- findSis.
Name: 'Asif'.
Sister: Dipty
true.

?- findUnc.
Name: 'Jahid'.
Uncle: Ujjal
true.

?- findAunt.
Name: 'Jahid'.
Aunt: Maisha
true.

?-
```

## Python Code:

```
tupplelist1 = [
    ('parent', 'Hasib', 'Rakib'),
    ('parent', 'Rakib', 'Sohel'),
    ('parent', 'Rakib', 'Rebeka'),
    ('parent', 'Rashid', 'Hasib'),
    ('parent', 'Maisha', 'Dipty'),
    ('parent', 'Maisha', 'Asif'),
    ('parent', 'Karim', 'Maisha'),
    ('parent', 'Lipi', 'Joy'),
    ('parent', 'Lipi', 'Jahid'),
    ('parent', 'Karim', 'Lipi'),
    ('parent', 'Karim', 'Ujjal'),
    ('parent', 'Rahim', 'Jahid'),
    ('parent', 'Rahim', 'Joy'),
    ('parent', 'Uday', 'Rahim'),
    ('parent', 'Uday', 'Anik')
]

male = ['Asif', 'Karim', 'Joy', 'Jahid', 'Rahim', 'Uday', 'Anik', 'Ujjal']
female = ['Maisha', 'Dipty', 'Lipy']

def findBro(B):
    i,j,k = 0,0,0
    flag = 2
    while (i <= len(tupplelist1) - 1):
        if ((tupplelist1[i][0] == 'parent') & (tupplelist1[i][2] == B)):
            for j in range(len(tupplelist1)):
                if ((tupplelist1[j][0] == 'parent') & (tupplelist1[i][1] == tupplelist1[j][1]) & (tupplelist1[i][2] != tupple
                    for k in range(len(male)):
                        if(male[k] == tupplelist1[j][2]):
                            print("Brother: " + tupplelist1[j][2])
                            flag = 0
                        else:
                            if(flag != 0):
                                flag = 1
        i = i + 1
    return flag
```

```
B = str(input("Enter Name: "))
flag = findBro(B)
if(flag == 1):
    print("No Brother! ")


def findSis(S):
    i,j,k = 0,0,0
    flag = 2
    while (i <= len(tupplelist1) - 1):
        if ((tupplelist1[i][0] == 'parent') & (tupplelist1[i][2] == S)):
            for j in range(len(tupplelist1)):
                if ((tupplelist1[j][0] == 'parent') & (tupplelist1[i][1] == tupplelist1[j][1]) & (tupplelist1[i][2] != tupplelist1[j][2])):
                    for k in range(len(female)):
                        if(female[k] == tupplelist1[j][2]):
                            print("Sister: " + tupplelist1[j][2])
                            flag = 0
                        else:
                            if(flag != 0):
                                flag = 1
        i = i + 1
    return flag

S = str(input("Enter Name: "))
flag = findSis(S)
if(flag == 1):
   print("No Sister! ")
```

```
def findUnc(U):
    i,j,k,l = 0,0,0,0
    flag = 2
    while (i <= len(tupplelist1) - 1):
        if ((tupplelist1[i][0] == 'parent') & (tupplelist1[i][2] == U)):
            for j in range(len(tupplelist1)):
                if ((tupplelist1[j][0] == 'parent') & (tupplelist1[i][1] == tupplelist1[j][2])):
                    for l in range(len(tupplelist1)):
                        if((tupplelist1[l][0] == 'parent') & (tupplelist1[j][1] == tupplelist1[l][1]) & (tupplelist1[i][1] != tupplelist1[l][2])):
                            for k in range(len(male)):
                                if(male[k] == tupplelist1[l][2]):
                                    print("Uncle: " + tupplelist1[l][2])
                                    flag = 0
                                else:
                                    if(flag != 0):
                                        flag = 1
        i = i + 1
    return flag

U = str(input("Enter Name: "))
flag = findUnc(U)
if(flag == 1):
   print("No Uncle! ")
```

```
def findAun(A):
    i,j,k,l = 0,0,0,0
    flag = 2
    while (i <= len(tupplelist1) - 1):
        if ((tupplelist1[i][0] == 'parent') & (tupplelist1[i][2] == A)):
            for j in range(len(tupplelist1)):
                if ((tupplelist1[j][0] == 'parent') & (tupplelist1[i][1] == tupplelist1[j][2])):
                    for l in range(len(tupplelist1)):
                        if((tupplelist1[l][0] == 'parent') & (tupplelist1[j][1] == tupplelist1[l][1]) & (tupplelist1[i][1] != tupplelist1[l][2])):
                            for k in range(len(female)):
                                if(female[k] == tupplelist1[l][2]):
                                    print("Aunt: " + tupplelist1[l][2])
                                    flag = 0
                                else:
                                    if(flag != 0):
                                        flag = 1
        i = i + 1
    return flag

A = str(input("Enter Name: "))
flag = findAun(A)
if(flag == 1):
   print("No Aunt! ")
```

## Input And Output:

```
================== RESTART: H:/Python/Python39/Ques4python.py ==================
Enter Name: Jahid
Brother: Joy
Brother: Joy
Enter Name: Asif
Sister: Dipty
Enter Name: Jahid
Uncle: Ujjal
Uncle: Anik
Enter Name: Jahid
Aunt: Maisha
>>>
```