



Ahsanullah University of Science and Technology (AUST)
Department of Computer Science and Engineering

Assignment 1

Course No.: CSE4108

Course Title: Artificial Intelligence Lab

Date of Submission-16.05.2023

Submitted To- Dr. S.M.A. Al-Mamun & Mr. Raihan Tanvir.

Submitted By-

MD Shihabul Islam Shovo

190204075

B

Year- 4th

Semester-1st

Department-CSE

- 1) **Question: 1.** Modify the Python and Prolog codes demonstrated above to find the grandparents of somebody.
- 2) **Question: 1.** Enrich the KB demonstrated above with 'brother', 'sister', 'uncle' and 'aunt' rules in Python and Prolog.

Answer:

Python Code:

while True:

```
# Assignment01
```

```
tupleList1 = [('parent', 'john', 'mary'),  
              ('parent', 'john', 'peter'),  
              ('parent', 'sue', 'mary'),  
              ('parent', 'sue', 'peter'),  
              ('parent', 'mary', 'tom'),  
              ('parent', 'peter', 'lisa'),  
              ('parent', 'peter', 'bob'),  
              ('parent', 'john', 'tyler'),  
              ('parent', 'sue', 'tyler')]
```

```
genderList = [('male', 'john'),  
              ('male', 'peter'),  
              ('male', 'tom'),  
              ('male', 'bob'),  
              ('male', 'tyler'),  
              ('female', 'sue'),  
              ('female', 'mary'),  
              ('female', 'lisa')]
```

```
ch = int(input("Enter your choice to find the relation of:\n1. Brother\n2. Sister\n3. Uncle\n4. Aunt\n5. Grandparents\nChoice: "))
```

```

if ch == 1 or ch == 2:
    X = input("Enter the name to find the siblings: ")
    if ch == 1:
        print("Brother: ", end=' ')
    else:
        print("Sister: ", end=' ')

    for i in range(len(tupleList1)):
        if tupleList1[i][0] == 'parent' and tupleList1[i][2] == X:
            for j in range(len(tupleList1)):
                if tupleList1[j][0] == 'parent' and tupleList1[i][1] == tupleList1[j][1] and
tupleList1[j][2] != X:
                    for k in range(len(genderList)):
                        if ch == 1:
                            if genderList[k][0] == 'male' and genderList[k][1] == tupleList1[j][2]:
                                print(tupleList1[j][2], end=' ')
                        else:
                            if genderList[k][0] == 'female' and genderList[k][1] == tupleList1[j][2]:
                                print(tupleList1[j][2], end=' ')
                    print()

elif ch == 3 or ch == 4:
    X = input("Enter the name to find someone's uncle/aunt: ")
    if ch == 3:
        print("Uncle: ", end=' ')
    else:
        print("Aunt: ", end=' ')

```

```

for l in range(len(tupleList1)):
    if tupleList1[l][0] == 'parent' and tupleList1[l][2] == X:
        for i in range(len(tupleList1)):
            if tupleList1[i][0] == 'parent' and tupleList1[i][2] == tupleList1[l][1]:
                for j in range(len(tupleList1)):
                    if tupleList1[j][0] == 'parent' and tupleList1[i][1] == tupleList1[j][1] and
tupleList1[j][2] != tupleList1[l][1]:
                        for k in range(len(genderList)):
                            if ch == 3:
                                if genderList[k][0] == 'male' and genderList[k][1] == tupleList1[j][2]:
                                    print(tupleList1[j][2], end=' ')
                            else:
                                if genderList[k][0] == 'female' and genderList[k][1] == tupleList1[j][2]:
                                    print(tupleList1[j][2], end=' ')

print()

elif ch == 5:
    X = input("Enter the name to find someone's grandparent: ")
    print('Grandparent:', end=' ')

    for i in range(len(tupleList1)):
        if tupleList1[i][0] == 'parent' and tupleList1[i][2] == X:
            for j in range(len(tupleList1)):
                if tupleList1[j][0] == 'parent' and tupleList1[i][1] == tupleList1[j][2]:
                    print(tupleList1[j][1], end=' ')

    print()

else:
    print("Invalid choice. Please try again.")

```

Prolog Code:

% Facts about family relationships

parent('john', 'mary').

parent('john', 'peter').

parent('sue', 'mary').

parent('sue', 'peter').

parent('mary', 'tom').

parent('peter', 'lisa').

parent('peter', 'bob').

parent('john', 'tyler').

parent('sue', 'tyler').

male('john').

male('peter').

male('tom').

male('bob').

male('tyler').

female('sue').

female('mary').

female('lisa').

% Rules to define family relationships

sibling(X, Y) :-

parent(Z, X),

```
parent(Z, Y),  
X \= Y.
```

```
brother(X, Y) :-  
    sibling(X, Y),  
    male(X).
```

```
sister(X, Y) :-  
    sibling(X, Y),  
    female(X).
```

```
uncle(X, Y) :-  
    parent(Z, Y),  
    brother(X, Z).
```

```
aunt(X, Y) :-  
    parent(Z, Y),  
    sister(X, Z).
```

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

```
findGrandparent :- write('Enter a name to find their grandparents: '),  
    read(X),  
    grandparent(Gp, X),  
    write('Grandparent is: '), write(Gp), tab(5),  
    fail.
```