FAMILY TREE

```
parent(jayaramreddy,bhanuprasadreddy). parent(jyothi,bhanuprasadreddy).
parent(bhanuprasadreddy,yamini). male(jayaramreddy). male(bhanuprasadreddy). female(jyothi).
female(yamini). mother(X,Y):-parent(X,Y),female(X). sister(X,Y):-parent(Z,X),parent(Z,Y),female(X).
grandparent(X,Z):-parent(X,Y),parent(Y,Z).
```

FACTORIAL

```
fact(0, 1). fact(N, F) :- (% The below is for +ve factorialN > 0-> (N1 is N - 1, fact(N1,F1), F is N * F1); %
The below is for -ve factorial N(N1 is N+1,fact(N1, F1),F is N * F1))
```

GCD OF NUM

```
gcd(X,Y):-X=Y,write('GCD of two numbers is '),write(X); X=0,write('GCD of two numbers is '),write(Y);
Y=0,write('GCD of two numbers is '),write(X); Y>X,Y1 is Y-X,gcd(X,Y1); X>Y,Y1 is X-Y,gcd(Y1,Y).
```

USER INPUT

```
reference("yamini", "9493272585"). reference("radhika", "8919666297"). reference("hemanth",
"9642499090"). reference("jayaram reddy", "9490013093").
```

OUTPUT USER

```
type(ungulate,animal). type(fish,animal). is_a(zebra,ungulate). is_a(herring,fish). is_a(shark,fish).
lives(zebra,on_land). lives(frog,on_land). lives(frog,in_water). lives(shark,in_water). can_swim(Y):-
type(X, animal), is_a(Y,X), lives(Y, in_water).
```

MONKEY

```
move(state(middle,onbox,middle,hasnot), grasp,state(middle,onbox,middle,has)).
move(state(P,onfloor,P,hasnot),climb, state(P,onbox,P,hasnot)). move(state(P,onfloor,P,hasnot),push,
state(P1,onfloor,P1,hasnot)). move(state(P1,onfloor,B,hasnot),walk, state(P2,onfloor,B,hasnot)).
canget(state(_,_,_,has)) :- write("get"). canget(State1) :- move(State1,Move,State2), canget(State2),
write(State2),nl.
```

LIST

```
11 A) PRINT LIST: printlist([]). printlist([X|List]) :- write(X),nl, printlist(List).
```

MEMBERS IN LIST

```
member(X,List):- delete(X,List,_). delete(X,[X|Tail],Tail). delete(X,[Y|Tail1],[Y|Tail2]):-
delete(X,Tail1,Tail2).
```