Output

1. Is Pete Mark’s parent?

A computer screen shot of a computer screen

Description automatically generated

1. Is Anne Jenny’s parent?

A screenshot of a computer

Description automatically generated

1. Who is Todd’s father?

A screenshot of a computer

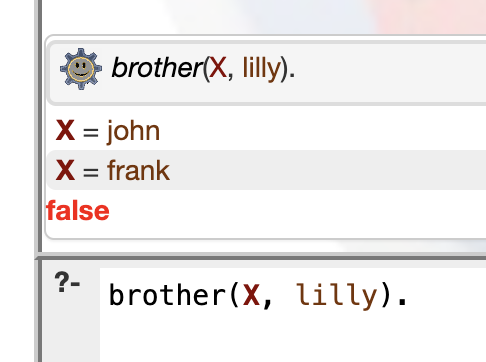
Description automatically generated

1. Who is Tom’s sibling?

A screenshot of a computer

Description automatically generated

1. Who is Lilly’s brother?



1. Who is Henry grandparent?

A screenshot of a computer

Description automatically generated

1. Who is Alice’s sister?

A screenshot of a computer

Description automatically generated

1. Is Frank Kate’s brother?

A screenshot of a computer

Description automatically generated

1. Who is Matt’s mother?

A screenshot of a computer

Description automatically generated

1. Is Mark Anne’s brother?

A screenshot of a computer

Description automatically generated

Code for Part 1.

male(pete).

male(mark).

male(tom).

male(john).

male(frank).

male(matt).

male(henry).

male(todd).

female(anne).

female(lilly).

female(kate).

female(alice).

female(jenny).

parent(pete, mark).

parent(pete, tom).

parent(pete, anne).

parent(mark, lilly).

parent(mark, john).

parent(mark, frank).

parent(tom, kate).

parent(anne, alice).

parent(anne, matt).

parent(alice, henry).

parent(matt, jenny).

parent(matt, todd).

sibling(X, Y) :- parent(Parent, X), parent(Parent, Y), not(X=Y).

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

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

father(X, Y) :- parent(X, Y), male(X).

mother(X, Y) :- parent(X, Y), female(X).

grandparent(X, Y) :- parent(X, Child), parent(Child, Y).

A screenshot of a computer

Description automatically generated

Without tracing list of 8 elements

A screenshot of a computer

Description automatically generated

Code for part 2.

max([], 0).

max([Head | Tail], Max) :- max(Tail, TailMax), Head > TailMax, Max is Head.

max([Head | Tail], Max) :- max(Tail, TailMax), Head =< TailMax, Max is TailMax.