

Programming Languages Assignment: Prolog

CODE:

```
male(jake).  
male(john).  
male(steve).  
male(alex).
```

```
female(mary).  
female(martha).  
female(rachel).  
female(mia).
```

```
parent(jake,rachel).  
parent(jake,mia).  
parent(jake,steve).  
parent(mary,rachel).  
parent(mary,mia).  
parent(mary,steve).  
parent(john,alex).  
parent(martha,alex).
```

```
sibling(X,Y) :- parent(M,X),parent(M,Y),not(X = Y).  
sister(X,Y) :- female(X),female(Y),sibling(X,Y),not(X = Y).  
mother(X,Y):- parent(X,Y),female(X),not(X = Y).
```

```
maxlist([Max],Max).  
maxlist([Head|Tail],Max) :- maxlist(Tail,TailMax), (Head > TailMax -> Max = Head ; Max =  
TailMax).
```

```
lastelement([Y],Y).  
lastelement([X|Xs],Y):- lastelement(Xs,Z), (Xs==[] -> Y = X; Y is Z).
```

```
append([A | B], C, [A | D]) :- append(B, C, D).  
append([], A, A).
```

```
member(A, [A | _]).  
member(A, [_ | B]) :- member(A, B).
```

```
union([],Z,Z).  
union([A|B], C, D) :- ((member(A,C)) -> union(B,C,D)).  
union([A|B], C, [A|D]) :- union(B,C,D).
```

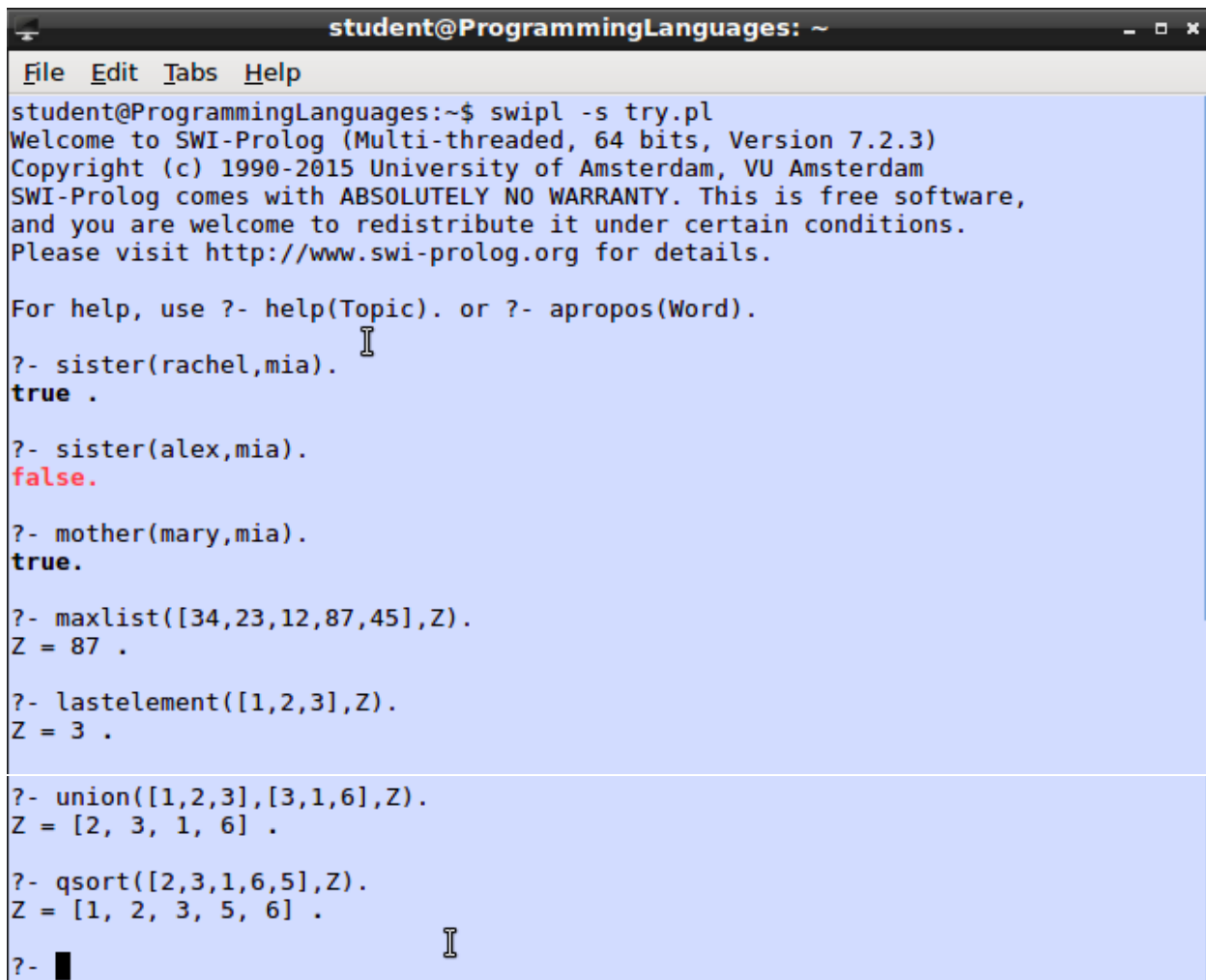
```
intersect([],_,[]).
intersect([X|Xs],Y,[X|Z]):- (member(X,Y)-> intersect(Xs,Y,Z)).
intersect([X|Xs],Y,Z):-not(member(X,Y)),intersect(Xs,Y,Z).
```

```
intersect_empty([X],[Y]).
intersect_empty(X,Y):- intersect(X,Y,[]).
```

```
qsort([],[]).
qsort([H|T],S):- split(H,T,L,R), qsort(L,LS), qsort(R,Rs), append(LS,[H|Rs],S).
```

```
split(_,[],[],[]).
split(H,[A|X],[A|Y],Z):- A <= H,!,split(H,X,Y,Z).
split(H,[A|X],Y,[A|Z]):- A > H,!,split(H,X,Y,Z).
```

OUTPUT:

A screenshot of a terminal window titled "student@ProgrammingLanguages: ~". The window has a menu bar with "File", "Edit", "Tabs", and "Help". The terminal shows the execution of a Prolog program. It starts with a prompt "student@ProgrammingLanguages:~\$ swipl -s try.pl", followed by a welcome message for SWI-Prolog (Multi-threaded, 64 bits, Version 7.2.3) and copyright information. The user then enters several Prolog queries, and the terminal displays the results. The queries and their outputs are: 1. "?- sister(rachel,mia)." returns "true .". 2. "?- sister(alex,mia)." returns "false.". 3. "?- mother(mary,mia)." returns "true.". 4. "?- maxlist([34,23,12,87,45],Z)." returns "Z = 87 .". 5. "?- lastelement([1,2,3],Z)." returns "Z = 3 .". 6. "?- union([1,2,3],[3,1,6],Z)." returns "Z = [2, 3, 1, 6] .". 7. "?- qsort([2,3,1,6,5],Z)." returns "Z = [1, 2, 3, 5, 6] .". The terminal ends with a prompt "?- " and a cursor.

```
student@ProgrammingLanguages: ~
File Edit Tabs Help
student@ProgrammingLanguages:~$ swipl -s try.pl
Welcome to SWI-Prolog (Multi-threaded, 64 bits, Version 7.2.3)
Copyright (c) 1990-2015 University of Amsterdam, VU Amsterdam
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software,
and you are welcome to redistribute it under certain conditions.
Please visit http://www.swi-prolog.org for details.

For help, use ?- help(Topic). or ?- apropos(Word).
?- sister(rachel,mia).
true .

?- sister(alex,mia).
false.

?- mother(mary,mia).
true.

?- maxlist([34,23,12,87,45],Z).
Z = 87 .

?- lastelement([1,2,3],Z).
Z = 3 .

?- union([1,2,3],[3,1,6],Z).
Z = [2, 3, 1, 6] .

?- qsort([2,3,1,6,5],Z).
Z = [1, 2, 3, 5, 6] .

?- 
```

```
student@ProgrammingLanguages: ~  
File Edit Tabs Help  
student@ProgrammingLanguages:~$ swipl -s try.pl  
Warning: /home/student/try.pl:44:  
    Singleton variables: [X,Y]  
Welcome to SWI-Prolog (Multi-threaded, 64 bits, Version 7.2.3)  
Copyright (c) 1990-2015 University of Amsterdam, VU Amsterdam  
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software,  
and you are welcome to redistribute it under certain conditions.  
Please visit http://www.swi-prolog.org for details.  
  
For help, use ?- help(Topic). or ?- apropos(Word).  
  
?- intersect_empty([1,2,3],[4,5,6]).  
true.  
  
?- intersect_empty([1,2,3],[4,5,3]).  
false.  
  
?- █
```