

Experiment no. 9

Aim: To find max of two number using Prolog.

Requirements: Compatible version of SWI-Prolog.

Theory:

There is a built-in predicate construction in Prolog which allows you to express exactly such conditions: the if-then-else construct. In Prolog, *if A then B else C* is written as (A -> B ; C). To Prolog this means: try A. If you can prove it, go on to prove B and ignore C. If A fails, however, go on to prove C ignoring B. The max predicate using the if-then-else construct looks as follows:

max(X,Y,Z) :-

```
( X =< Y
-> Z = Y
; Z = X
).
```

Prolog's Persistence

- When a subgoal fails, Prolog will backtrack to the most recent successful goal and try to find another solution.
- Once there are no more solutions for this subgoal it will backtrack again; retrying every subgoal before failing the parent goal.
- A call can match any clause head.
- A redo ignores old matches.

Cut !

The cut, in Prolog, is a goal, written as `!`, which always succeeds, but cannot be backtracked past. It is used to prevent unwanted backtracking, for example, to prevent extra solutions being found by Prolog.

Code:

```
/*Without Cut*/
```

```
maximum(X,Y,Z):- (X>=Y-> Z=X; Z=Y).
```

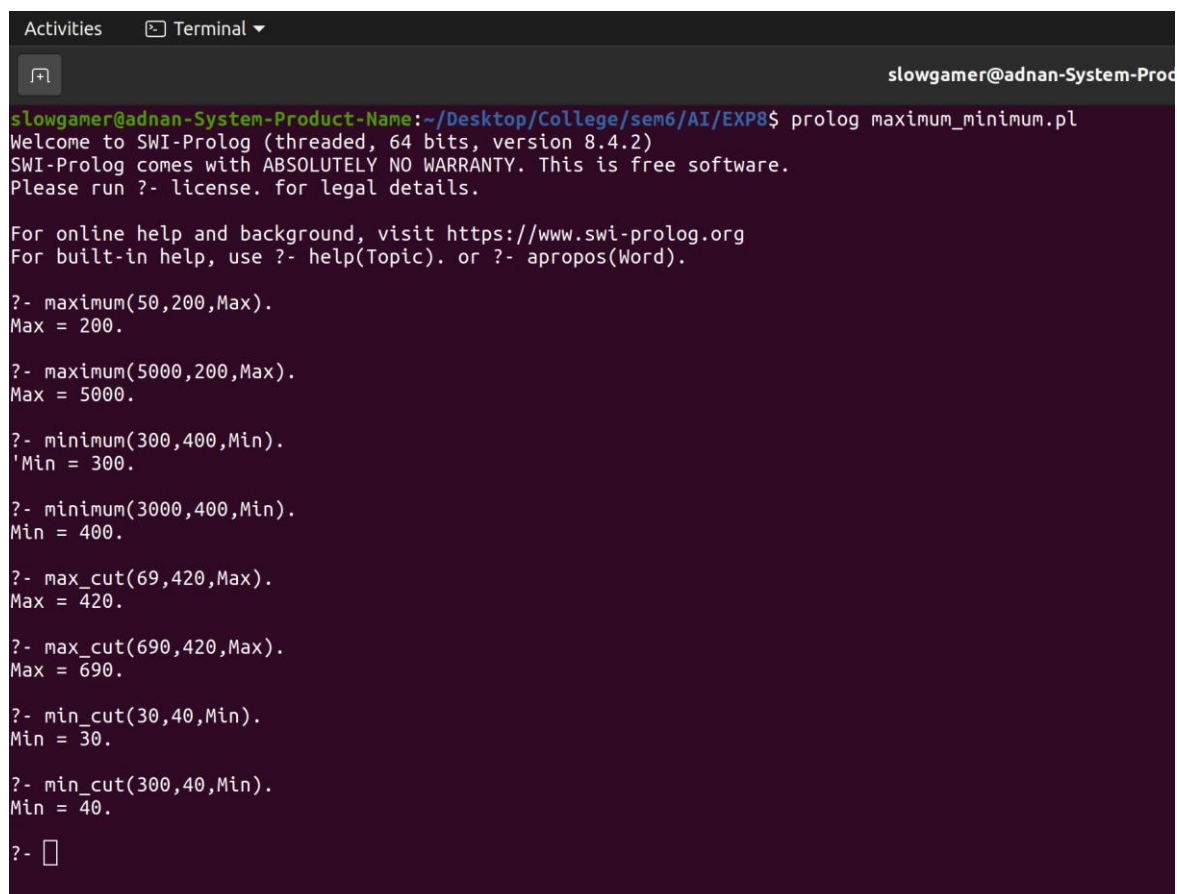
```
minimum(X,Y,Z):- (X>=Y-> Z=Y; Z=X).
```

```
/*With Cut*/
```

```
max_cut(X,Y,Max):- X>=Y,!,Max=X; Max=Y.
```

```
min_cut(X,Y,Min):- X>=Y,!,Min=Y; Min=X.
```

Output:

A screenshot of a terminal window titled 'Terminal' with a dark background. The user 'slowgamer@adnan-System-Product-Name' is in the directory '~/Desktop/College/sem6/AI/EXP8'. They have run the command 'prolog maximum_minimum.pl'. The terminal shows the Prolog welcome message and several queries. The queries and their outputs are: 1. '?- maximum(50,200,Max).' followed by 'Max = 200.'; 2. '?- maximum(5000,200,Max).' followed by 'Max = 5000.'; 3. '?- minimum(300,400,Min).' followed by 'Min = 300.'; 4. '?- minimum(3000,400,Min).' followed by 'Min = 400.'; 5. '?- max_cut(69,420,Max).' followed by 'Max = 420.'; 6. '?- max_cut(690,420,Max).' followed by 'Max = 690.'; 7. '?- min_cut(30,40,Min).' followed by 'Min = 30.'; 8. '?- min_cut(300,40,Min).' followed by 'Min = 40.'; and finally '?- ' followed by a cursor. The terminal window also shows a top bar with 'Activities' and 'Terminal' tabs, and a title bar with 'slowgamer@adnan-System-Product-Name'.

Conclusion: We have successfully implemented Maximum number finding code in Prolog.