

Experiment No. 8

1) Program to perform Arithmetic Operations –

```
add(X, Y, Z) :- Z is X + Y.  
subtract(X, Y, Z) :- Z is X - Y.  
multiply(X, Y, Z) :- Z is X * Y.  
divide(X, Y, Z) :- Z is X / Y.
```

Output –



The screenshot displays a Prolog interpreter window with four separate execution blocks. Each block contains a query followed by its result. The queries are: `add(5, 5, Result).`, `subtract(10, 5, Result).`, `multiply(10, 5, Result).`, and `divide(10, 5, Result).`. The corresponding results are `Result = 10`, `Result = 5`, `Result = 50`, and `Result = 2`.

 <code>add(5, 5, Result).</code>	  
Result = 10	
 <code>subtract(10, 5, Result).</code>	  
Result = 5	
 <code>multiply(10, 5, Result).</code>	  
Result = 50	
 <code>divide(10, 5, Result).</code>	  
Result = 2	

2) Program to check the number is even, odd or prime –

```
is_even(X) :-  
    X mod 2 == 0.  
is_odd(X) :-  
    X mod 2 == 1.  
  
is_prime(X) :-  
    X > 1,  
    Upper is floor(sqrt(X)),  
    \+ (between(2, Upper, Y), X mod Y == 0).
```

Output –



The screenshot shows a Prolog interpreter window with the following queries and results:

- Query: `is_even(7).` Result: **false**
- Query: `is_even(8).` Result: **true** (with a small '1' in the bottom right corner)
- Query: `is_odd(8).` Result: **false**
- Query: `is_odd(9).` Result: **true** (with a small '1' in the bottom right corner)
- Query: `is_prime(10).` Result: **false**
- Query: `is_prime(13).` Result: **true** (with a small '1' in the bottom right corner)

3) Program to calculate factorial of a number –

```
factorial(0, 1).  
factorial(N, Result) :-  
    N > 0,  
    Prev is N - 1,  
    factorial(Prev, PrevResult),  
    Result is N * PrevResult.
```

Output –



The screenshot shows a Prolog interpreter window with the following queries and results:

- Query: `factorial(5, Result).` Result: **Result = 120**
- Query: `factorial(10, Result).` Result: **Result = 3628800**
- Query: `factorial(3, Result).` Result: **Result = 6**
- Query: `factorial(7, Result).` Result: **Result = 5040**

4) Program to find the greatest among three numbers –

```
greatest(X, Y, Z, G) :-  
    X >= Y, X >= Z,  
    G is X.  
greatest(X, Y, Z, G) :-  
    Y >= X, Y >= Z,  
    G is Y.  
greatest(_, _, Z, G) :-
```

Output –



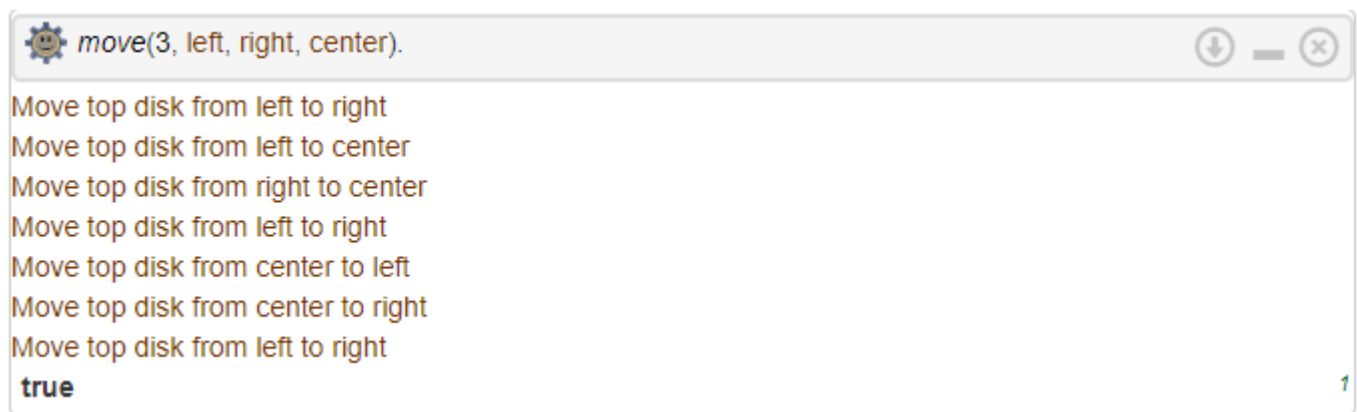
The screenshot shows a Prolog interpreter window with three queries and their results. Each query is in a separate box with a gear icon on the left and window controls on the right.

```
greatest(10, 5, 20, Result).  
Result = 20  
greatest(3, 5, 1, Result).  
Result = 5  
greatest(13, 5, 1, Result).  
Result = 13
```

5) Program to solve the problem of Tower Of Hanoi (TOH) –

```
move(1,X,Y,_) :-  
    write('Move top disk from '), write(X), write(' to '), write(Y), nl.  
move(N,X,Y,Z) :-  
    N>1,  
    M is N-1,  
    move(M,X,Z,Y),  
    move(1,X,Y,_),  
    move(M,Z,Y,X).
```

Output –



```
move(3, left, right, center).  
Move top disk from left to right  
Move top disk from left to center  
Move top disk from right to center  
Move top disk from left to right  
Move top disk from center to left  
Move top disk from center to right  
Move top disk from left to right  
true
```

The image shows a Python REPL window with a title bar containing a gear icon, a download icon, a close icon, and a maximize icon. The code `move(3, left, right, center).` is entered. The output consists of seven lines of text describing disk moves, followed by the boolean value `true`. A small number '1' is visible in the bottom right corner of the window.