

Name: Brijesh Rameshbhai Rohit

Admission number: U19CS009

PPL-ASSIGNMENT-08

1. Write a prolog program to implement a Menu Driven Calculator.

```
%U19CS009
%BRIJESH ROHIT

add(A,B,Sum):-
    Sum is A+B.
sub(A,B,Diffs):-
    Diffs is A-B.
mul(A,B,Prod):-
    Prod is A*B.
div(A,B,Quo):-
    Quo is A/B.

start:-
    write("Please enter first number : "),
    read(A),
    write("Please enter second number : "),
    read(B),
    write("Please enter\n"),
    write("1 : Add two numbers\n"),
    write("2 : Subtract two numbers\n"),
    write("3 : Multiply two numbers\n"),
    write("4 : Divide two numbers\n"),
    write("Enter your choice: "),
    read(C),
    (C:=1 ->
        (add(A,B,Sum),
         write("Addition : "),
         write(Sum),nl)
    ; C:=2 ->
        (sub(A,B,Diff),
         write("Difference : "),
         write(Diff),nl)
    ; C:=3 ->
        (mul(A,B,Prod),
```

```

        write("Product : "),
        write(Prod),nl)
;    C:=4 ->
        (B:=0 ->
            write("Division not possible!!\n")
        ;    div(A,B,Quo),
            write("Quotient : "),
            write(Quo),nl))
;    C:=5 ->
        (
            halt
        )
;    start.

:-
    start.

```

OUTPUT :

```

?- consult('calculator.pl').
Warning: /home/brijesh/Documents/ppl/ppl-assign08/calculator.pl:13:
Warning: Singleton variable in branch: C
Please enter first number : 12.
Please enter second number : |: 13.
Please enter
1 : Add two numbers
2 : Subtract two numbers
3 : Multiply two numbers
4 : Divide two numbers
Enter your choice: |: 1.
Addition : 25
true.

```

```

?- consult('calculator.pl').
Warning: /home/brijesh/Documents/ppl/ppl-assign08/calculator.pl:13:
Warning: Singleton variable in branch: C
Please enter first number : 12.
Please enter second number : |: 0.
Please enter
1 : Add two numbers
2 : Subtract two numbers
3 : Multiply two numbers
4 : Divide two numbers
Enter your choice: |: 4.
Division not possible!!
true.

```

2. Write a prolog program to find maximum and minimum salaries of given 3 employees.

```

%U19CS009
%BRIJESH ROHIT

max(A,B,C):-
    A>B,
    A>C,
    write(A).
max(A,B,C):-
    A>B,
    write(C).
max(_,B,C):-
    B>C,
    write(B).
max(_,_,C):-
    write(C).

min(A,B,C):-
    A<B,
    A<C,
    write(A).
min(A,B,C):-
    A<B,
    write(C).
min(_,B,C):-
    B<C,
    write(B).

```

```

min(_,_ ,C):-
    write(C).

maxmin(A,B,C):-
    write("Max salary : "),
    max(A,B,C),nl,
    write("Min salary : "),
    min(A,B,C).

:-
    write("Please enter first employee's salary : "),
    read(A),
    write("Please enter second employee's salary : "),
    read(B),
    write("Please enter third employee's salary : "),
    read(C),
    maxmin(A,B,C).

```

OUTPUT :

```

?- consult('max.pl').
Please enter first employee's salary : 12.
Please enter second employee's salary : |: 13.
Please enter third employee's salary : |: 14.
Max salary : 14
Min salary : 12
true.

?- consult('max.pl').
Please enter first employee's salary : 12.
Please enter second employee's salary : |: 12.
Please enter third employee's salary : |: 12.
Max salary : 12
Min salary : 12
true.

```

3. Write a prolog program to check whether a given number is odd or even.

```
%U19CS009
%BRIJESH ROHIT

check_even(N):-
    Y is N//2,Y*2==N
    ->format('~w is even.\n',[N]);
    format('~w is odd.\n',[N]).

:-
    write('Please enter a number : '),
    read(A),
    check_even(A).
```

OUTPUT:

```
?- consult('odd.pl').
Please enter a number : 0.
0 is even.
true.

?- consult('odd.pl').
Please enter a number : 11.
11 is odd.
true.

?- consult('odd.pl').
Please enter a number : 12.
12 is even.
true.
```

4. Write a prolog program to check whether a given year is a leap year or not.

```
%U19CS009
%BRIJESH RHOIT

leap(Y):-
    Y mod 400 == 0 , write("It is a Leap Year!!\n");
    Y mod 100 == 0 , write("It is not a Leap Year!!\n");
    Y mod 4    == 0 , write("It is a Leap Year!!\n");
    write("It is not a Leap Year!!\n").

:-
    write("Please enter a year : "),
    read(Y),
    leap(Y).
```

OUTPUT :

```
?- consult('leap.pl').
Please enter a year : 2000.
It is a Leap Year!!
true.

?- consult('leap.pl').
Please enter a year : 1700.
It is not a Leap Year!!
true.

?- consult('leap.pl').
Please enter a year : 2004.
It is a Leap Year!!
true.
```

5. Write a prolog program to give grade to a student based on total marks given:

- 00 - 80 Grade A
- 60 - 79 Grade B
- 35 - 59 Grade C
- 1 - 35 Grade D

```
%U19CS009
%BRIJESH ROHIT

check(P):-
    P=<100,P>=80, write("A"),nl;
    P<80,P>=60, write("B"),nl;
    P<60,P>=35, write("C"),nl;
    write("D"),nl.
:-
    write("Please enter percentage : "),
    read(P),
    write("Your Grade : "),
    check(P).
```

OUTPUT :

```
?- consult('grade.pl').
Please enter percentage : 90.
Your Grade : A
true.

?- consult('grade.pl').
Please enter percentage : 70.
Your Grade : B
true.

?- consult('grade.pl').
Please enter percentage : 50.
Your Grade : C
true.

?- consult('grade.pl').
Please enter percentage : 30.
Your Grade : D
true.
```

6. Write a prolog program to take values of length and breadth of a rectangle from the user and check if it is square or not.

```
%U19CS009
%BRIJESH ROHIT

check(L,B):-
    L==B
    -> write("It is a square!!\n")
    ; write("It is not a square!!\n").

:-
    write("Enter Length : "),
    read(L),
    write("Enter Breadth : "),
    read(B),
    check(L,B).
```

OUTPUT :

```
?- consult('sq.pl').
Enter Length : 12.
Enter Breadth : |: 13.
It is not a square!!
true.

?- consult('sq.pl').
Enter Length : 12.
Enter Breadth : |: 12.
It is a square!!
true.
```


7. Write a PROLOG program to calculate the roots of quadratic equation. Consider all possibilities real, equal, imaginary.

```
%U19CS009
%BRIJESH ROHIT
roots(A,B,C):-
    (B*B-4*A*C) > 0 ->
        write("Roots are REAL and UNIQUE!!\n"),
        A1 is 2*A,
        R1 is ((-B+sqrt(B*B-4*A*C))/(A1)),
        R2 is ((-B-sqrt(B*B-4*A*C))/(A1)),
        write("Roots : "), write(R1),write(", "),write(R2),nl
;    (B*B-4*A*C) == 0 ->
        write("Roots are REAL and Equal!!"),nl,
        R1 is -B/(2*A),
        write("Roots : "), write(R1),write(", "),write(R1),nl
;    (B*B-4*A*C) < 0 ->
        write("Roots are Imaginary!!"),nl,
        A1 is 2*A,
        D1 is (-1)*(B*B-4*A*C),
        R1 is ((-B)/A1),
        I1 is ((sqrt(D1)/A1)),
        write("Roots : "), write(R1), write(" + "),
write(I1),write("i and "),
        write(R1), write(" - "), write(I1), write("i"),nl.
check(A):-
    A==0 -> write("A can't be 0\n"),halt;
    !.
:-
    write("Please enter coefficients of equation 'ax^2 + bx +
c':\n"),
    write("Please enter value of a : "),
    read(A),
    check(A),
    write("Please enter value of b : "),
    read(B),
    write("Please enter value of c : "),
    read(C),
    roots(A,B,C).
```

OUTPUT :

```
?- consult('root.pl').
Please enter coefficients of equation 'ax^2 + bx + c':
Please enter value of a : 0.
A can't be 0
```

```
?- consult('root.pl').
Please enter coefficients of equation 'ax^2 + bx + c':
Please enter value of a : 4.
Please enter value of b : |: -16.
Please enter value of c : |: 12.
Roots are REAL and UNIQUE!!
Roots : 3.0, 1.0
true.
```

```
?- consult('root.pl').
Please enter coefficients of equation 'ax^2 + bx + c':
Please enter value of a : 1.
Please enter value of b : |: 2.
Please enter value of c : |: 1.
Roots are REAL and Equal!!
Roots : -1, -1
true.
```

```
?- consult('root.pl').
Please enter coefficients of equation 'ax^2 + bx + c':
Please enter value of a : 10.
Please enter value of b : |: 8.
Please enter value of c : |: 4.
Roots are Imaginary!!
Roots : -0.4 + 0.4898979485566356i and -0.4 - 0.4898979485566356i
true.
```

8. Write a PROLOG program to find the number whether the number is positive, negative or Zero.

```
%U19CS009
%BRIJESH ROHIT

check(N):-
    N:=0 -> write("You entered 0.\n");
    N>0 -> write("You entered Positive number.\n");
    N<0 -> write("You entered Negative number.\n").

:-
    write("Please enter a number : "),
    read(N),
    check(N).
```

OUTPUT :

```
?- consult('sign.pl').
Please enter a number : 12.
You entered Positive number.
true.

?- consult('sign.pl').
Please enter a number : 0.
You entered 0.
true.

?- consult('sign.pl').
Please enter a number : -10.
You entered Negative number.
true.
```