

AI LAB-5

AIM: To study about controlling execution in prolog using cut and fail predicate

- Failure can be forced in any rule by using the built-in fail predicate.
- The fail forces backtracking in an attempt to unify with other clause.
- The primary purpose of cut is to prevent or block backtracking based on a specified condition. The cut predicate is specified as an exclamation point(!).

Sample Program: I

```
predicates
    s(integer,integer)
    q(integer,integer)
    i(integer)
    j(integer)
    k(integer)

clauses
    s(X,Y) :- q(X,Y).
    s(0,0).
    q(X,Y) :- i(X),j(Y).
    q(X,Y) :- k(X).
    i(1).
    i(2).
    j(1).
    j(2).
    j(3).
    k(5).
Goal: S(P,Q)
????
```

Sample Program: II

```
predicates
    s(integer,integer)
    q(integer,integer)
    i(integer)
    j(integer)
    k(integer)

clauses
    s(X,Y) :- q(X,Y).
    s(0,0).
    q(X,Y) :- i(X),!,j(Y).
    q(X,Y) :- k(X).
    i(1).
    i(2).
    j(1).
    j(2).
    j(3).
    k(5).

Goal: S(P,Q)
      ?????
```

Sample Program: III

```
predicates
    location(String,String)
    go

clauses
    go:-
        writef("%-10%5\n","CITY","STATE"),fail.

    go:-location(X,Y), writef("%-10%5\n",X,Y),fail.
    go.
    location("Jackson","MS").
    location("Washington","DC").
    location("Raleigh","NC").
```

Sample Program: IV

```
predicates
    location(String,String)
    go
    chkstate(String)
clauses
    go:-writef("%-10%5\n","CITY","STATE"),!.
    go:-location(X,Y),chkstate(Y),writef("%-10%5\n",X,Y),fail.
    location("Jackson","MS").
    location("Washington","DC").
    location("Raleigh","NC").
    chkstate("DC"):-write("IN DC"),fail.
    chkstate(_).
```

Exercises:

1. Implement a prolog program to find minimum and maximum of two integers using cut and/ or fail predicate. Program must have three arguments and it must handle all cases.

```
max(100,101,X).
X = 101
Yes
```

```
max(3,2,X).
X = 3
Yes
```

```
max(2,3,2).
No
```

2. Write a prolog program to verify that given year is leap year or not using cut and/ or fail predicate.
Note: A year is a leap year if it is divisible by 4, but century years are not leap years unless they are divisible by 400. So, the years 1700, 1800, and 1900 were not leap years, but the year 2000 was.
3. Write a prolog program to verify that given number is prime or not using cut and/ or fail predicate.