

Write a prolog program that represent the following database about population تعداد السكان and area of various countries:

country	Population (in million people)	Area (in millions of square miles)
Egypt	90	1
India	600	2
China	900	4
Brazil	120	3

- a- **Facts**; use the predicate pop to represent the relationship between a country and its population, and the predicate area to represent the relationship between a country and its area
- b- **Rules**; use the predicate density: the population density of country X is Y, if the population of X is P, and the area of X is A, and Y is calculated by dividing P by A.
- c- Write the following questions using prolog syntax:
- 1) Find the population of India.
 - 2) Find the area of China.
 - 3) Find the population density of Brazil.

Sol:

- a- **Facts**:
- ```
pop(egypt,90).
pop(india,600).
pop(china,900).
pop(brazil,120).
area(egypt,1).
area(india,2).
area(china,4).
area(brazil,3).
```
- b- **Rules**:
- ```
density(X,Y):- pop(X,P) , area(X,A), Y is P / A.
```
- c- **Questions**:
- 1) `pop(india,X).`
X = 600.
 - 2) `area(china, X).`
X= 4.
 - 3) `density(brazil,X).`
X= 40.

Write a prolog program that represents the following database as:

a-**Facts**: using the predicates: male, female and age.

- Aly is 42 years old. - Hany is 15 years old. - saly is 20 years old.

b-**Rules**: using the predicate older: Person1 is older than Person2 if age of Person1 is greater than age of Person2.

C-Write the following questions using prolog syntax

- 1) What is Aly's age?
- 2) Who is male?
- 3) List names with age<50?
- 4) List names with age<40, and they are female?
- 5) Find persons older than hany.
- 6) Find persons with ages less than aly.

Sol:

a-facts:

- 1- male (aly).
- 2- male (hany).
- 3- female (saly).
- 4- age (aly, 42).
- 5- age (hany, 15).
- 6- age (saly, 20).

b-Rules:

older (P1, P2):- age (P1, A1), age (P2, A2), A1 > A2.

c-

- 1- age (aly, X).
- 2- male (X).
- 3- age (X, Y), Y < 50.
- 4- age (X, Y), Y < 40 , female (X).
- 5- older (X, hany).
- 6- older (aly, X).

Write a prolog program that represent the following database about the reigns **حكم فترة**

a- **Facts**: using predicate reigns to represent the following table.

Prince name	Year from	To
Aly	844	878
Samy	878	916
Hany	916	950

b- **Rules** : define the predicate prince(X,Y), which is true if the prince named X was on the throne **العرش** during year Y

c-

- 1) Who is on the throne at year 916?
- 2) Was samy on the throne at year 1979?

a- reigns (aly, 844, 878).

reigns (samy, 878, 916).

reigns (hany, 916, 950).

b- prince(X, Y) :- reigns(X, A, B),

Y >= A,

Y <= B.

c- prince (samy, 916).

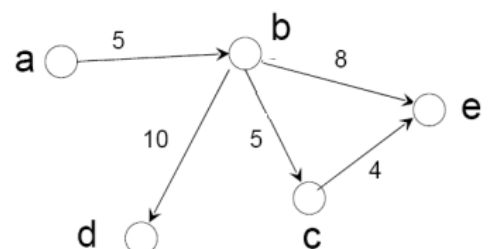
prince (aly, 1979).

Write a Prolog program that represents the following directed graph on five nodes:

a) Express the above graph in Prolog facts using the predicate link: to describe which nodes are connected and distance between them. (Nodes are connected one-way direction).

b) Write a recursive predicate route(X,Y,D) which is true if we can go from node X to node Y by following any number of links, and D represent corresponding distance.

c) 1) What nodes are direct connected to point b and their distances?



2) Which nodes are terminated at node c and their distances?

Express the following information about flights as a collection of Prolog facts:

Flight Number	From	To	Duration of Flight (Hours)
BA134	Larnaca	Rome	3.5
AL222	Rome	Paris	2.5
BA657	Larnaca	London	6.2
AF101	Paris	Brussels	0.6
AL711	Rome	Madrid	2.2

(b) Write Prolog clauses to define a predicate $\text{travel}(X, Y)$ which is true when city Y can be reached by some sequence of (one or more) flights from city X .

(c) Use Prolog queries to answer the following questions:

- (i) Which cities can be reached by a single flight from Larnaca?
- (ii) How long does flight AL711 take?
- (iii) From which cities can you fly into London (by a single flight)?
- (iv) Which cities can be reached by some sequence of flights from Rome?
- (v) Can one travel from Larnaca to Brussels?
- (vi) From which cities can travel to Paris by some sequence of flights?

Sol

a) facts:

$\text{flight}(\text{ba134}, \text{larnaca}, \text{rome}, 3.5).$

$\text{flight}(\text{al222}, \text{rome}, \text{paris}, 2.5).$

$\text{flight}(\text{ba657}, \text{larnaca}, \text{london}, 6.2).$

$\text{flight}(\text{af101}, \text{paris}, \text{brussels}, 0.6).$

$\text{flight}(\text{al711}, \text{rome}, \text{madrid}, 2.2).$

b) rules:

$\text{travel}(X, Y):- \text{flight}(_, X, Y, _).$

$\text{travel}(X, Y):- \text{flight}(_, X, Z, _), \text{travel}(Z, Y).$

c) queries:

1) $\text{flight}(_, \text{larnaca}, X, _).$

2) $\text{flight}(\text{al711}, _, _, X).$

3) $\text{flight}(_, X, \text{london}, _).$

4) $\text{travel}(\text{roma}, X).$

5) $\text{flight}(_, \text{larnaca}, \text{brussels}, _).$

6) $\text{travel}(X, \text{roma}).$

a- Write a prolog program to find the output of the following Boolean function:

$F = A (B' + C)$, and use it to find F if $A = 0$, $b=1$, $c=0$.

Sol:

a-

$$F = AB' + AC$$

not(0,1).

not(1,0).

and(0,0,0).

and(0,1,0).

and(1,0,0).

and(1,1,1).

or(0,0,0).

or(0,1,1).

or(1,0,1).

or(1,1,1).

fun(A,B,C,F):-not (B, X1), and (A,X1,X2), and(A,C,X3),or(X2,X3,F).

b- fun(0,1,0,F).

write a prolog program to calculate the summation of the integers from 1 to any given number. and then use it to find the summation from 1 to 10.

Sol:

a-

sum(1,1).

sum(N,S):- N>1,

N1 is N-1,

sum(N1, S1),

S is N + S1.

b-sum(10,S).

write a prolog program to calculate the summation of the odd numbers from 1 to any given odd number N. and then use it to find the summation from 1 to 5.

Sol:

a-

sum(1,1).

sum(N,S):- N>1,

N1 is N-2,

sum(N1, S1),

S is N + S1.

b-sum(5,S).

write a prolog program to calculate the factorial of any integer numbers. and then use it to find the factorial of 4.

Sol:

a-

fac(0,1).

fac(N,F) :- N > 0,

N1 is N - 1,

fac(N1, F1),

F is N * F1.

b-fac(4,F).

Write a program that read student total marks and print succeed if his total marks ≥ 50 , else write fail. (Define this procedure in a more efficient way using cuts), and test it.

```
test:-
write('enter score :'), read(X),process(X).
process(X):-X>=50,!, write('succeed').
process(X):- write('fail').
```

The relation between X and Y can be specified by two rules:

Rule 1: if $X \leq 3$. then $Y = 2X$

Rule 2: if $X > 3$. Then $Y = 4X+5$

Write a Prolog program that reads from the user the value of X and then output the value of Y, and test it.

```
test:-
read(X), process(X).
process(X):-
X=<3,!, Y is 2*X, write(Y).
process(X):-
Y is 4*X+5, write(Y).
```

Write a Prolog program that represents the following database about the age and sex of students:

- Facts; use the predicate student to represent the relationship between a student name, sex, and age.
- Rules; using the predicate older: Person 1 is older than Person 2 if age of Person1 is greater than age of Person2.
- Write the following questions using Prolog syntax:
 - What is Hany's age?
 - Who is male?
 - List names with age < 30 , and they are female
 - Find students older than Mona
 - Find students with ages less than Ahmed.
 - Find students younger than Ahmed and older than Mona.

Name	Sex	Age
Ahmed	male	30
Hany	male	25
Rwan	female	20
Mona	female	18

- student (ahmed, male, 30).
student (hany, male, 25).
student (rwan, female, 20).
student (mona, female,18).
- older(P1,P2):-student (P1, _, A1), student (P2, _, A2), A1>A2.
- 1- student (hany, _,A).
- 2- student (X, male, _).
- 3- student(X, female,A), A<30.
- 4- older (X, mona).
- 5- older (ahmed,Y).
- 6- older (ahmed,Y), older (Y, mona).