Lists

Any language needs a way to handle collections of objects and prolog is no exception. In Prolog, *a* list is an object that contains an arbitrary number of other objects within it. Lists correspond roughly to arrays in other languages, but, unlike an array, a list does not require you to declare how big it will be before you use it.

A list in PROLOG is a structure of the form [t1, t2, : : : , tn], The **brackets** are the beginning and the end of the list, and the **commas** separate the various elements.

A list that contains the numbers 1, 2, and 3 is written as [1, 2, 3]

The order of the elements in this list matters:

* Number "1" is the first element,
* "2" - the second,
* "3" - the third.

The list [ 1, 2, 3 ] is different from the list [ 1, 3, 2 ].

Here are some examples:

["dog", "cat", "canary"]

["valerie ann", "jennifer caitlin", "benjamin thomas"]

The same element can be present in the list several times, for example:

[1,2,1,3,1]

Declaring Lists

To declare the domain for a list of integers, you use the domains declaration, like this:

domains

name\_list = integer\*.

Heads and Tails

A list is really a recursive compound object. It consists of two parts: the head, of a list, which is the first element, and the tail, which is a list comprising all the subsequent elements.

**The tail of a list is always a list; the head of a list is an element.**

For example:

The head of [a, b, c] is a

The tail of [a, b, c] is [b, c]

What happens when you get down to a one-element list? The answer is that:

The head of [c] is c

The tail of [c] is []

If you take the first element from the tail of a list enough times, you will eventually get down to an **empty list ([ ]).**

The **empty list** cannot be broken into **head and tail**.

س1/ اكتب برنامج لإيجاد اكبر عنصر من قائمة :

Max([1,4,6,8],X).

X=8.

Max([X],X).

Max([H1,H2|T],X):- H1>=H2,Max([H1|T],X),!.

Max([H1,H2|T],X):- Max([H2|T],X),!.

س2/ أكتب برنامج لإدخال قائمة إذا كان العنصر اكبر من 5 يتم إضافة 2 وإذا اقل أو مساوي يتم إضافة 1

Test([1,5,3,8,2],L).

L=[2,6,4,10,3].

Test([],[]).

Test([H|T],[H1|T1]):- H > 5 , H1=H+2,Test(T,T1).

Test([H|T],[H1|T1]):- H < =5 , H1=H+1,Test(T,T1).

س 3/ برنامج يقوم بإضافة صفر بعد كل قيمة من قيم عناصر القائمة

T([1,2,5,7],L).

L=[1,0,2,0,5,0,7,0].

T([],[]).

T([H|T],[H,0|T1]):- T(T,T1).

س 4/ برنامج لحذف عنصر متكرر من قائمة

del(4,[1,4,2,5,7,4,9,4],L).

L= [1,2,5,7,9].

del(X,[],[]):-!.

del(X,[X|T],Y):- del(X,T,Y),!.

del(X,[H|T],[H|T1]):- del(X,T,T1).

س5/ برنامج يقوم بتصفير القيم الزوجية فقط

Test([5,7,4,3,2],L)

L=[5,7,0,3,0]

Test([],[]).

Test([H|T],[H1|T1]):- H mod 2 = 0, H1=0, Test(T,T1),!.

Test([H|T],[H|T1]):- Test(T,T1).

س6/ برنامج لإيجاد حاصل جمع الإعداد الزوجية فقط

Sum([1,4,9,3,2,10],M)

M=16

Sum([],0).

Sum([H|T],X):- H mod 2 =0, sum(T,X1), X=X1+H ,!.

Sum([H|T], X):- sum(T,X).

س7/ برنامج يقوم كما في المثال التالي

Test([2,1,3,4],L)

L=["even","odd","odd","even"].

Test([],[]).

Test([H|T],[X|T1]):- H mod 2 = 0 , X=even,Test(T,T1),!.

Test([H|T],[X|T1]):- X=odd, Test(T,T1).

س 8 / أكتب برنامج لصنع قائمة تحتوي على كل العناصر ضمن الفترة المعطاة

**Create a list containing all integers within a given range.**

Example: range(4,9,L). L = [4,5,6,7,8,9]

rang(Y,Y,[Y]).

range(X,Y,[X|T]):- X1=X+1,X1<=Y, range(X1,Y,T).

س 9/ أكتب برنامج لمضاعفة قيم عناصر قائمة

**Duplicate the elements of a list.**

Example: dupli([a,b,c,c,d],X). X = [a,a,b,b,c,c,c,c,d,d]

dupli([],[]).

dupli([H|T],[H,H|T1]):-dupli(T,T1).

س10/ برنامج لتقليل العناصر المتكررة بقائمة

**Eliminate consecutive duplicates of list elements.**

Example: compress([a,a,a,a,b,c,c,a,a,d,e,e,e,e],X). X = [a,b,c,d,e]

compress([],[]).

compress([H|T],[H|T1]):- del(H,T,L),compress(L,T1).

del(X,[],[]).

del(X,[H|T],[H|T1]):- X< > H, del(X,T,T1),!.

del(X,[H|T],L):- del(X,T,L).