

Apellidos:

Nombre:

---

**Ejercicio 1** [2 puntos] Una lista está acotada si todos sus elementos son menores que su longitud. Definir la relación `lista_acotada(+L)` que se verifique si todos los elementos de la lista de números `L` son menores que la longitud de `L`. Por ejemplo,

```
?- lista_acotada([1,0,2]).
Yes
?- lista_acotada([1,3,2]).
No
```

.....  
**Solución:** La definición de `lista_acotada` es

```
lista_acotada(L) :-
    length(L,N),
    lista_acotada_aux(L,N).
```

donde `lista_acotada_aux(+L,+N)` se verifica si todos los elementos de la lista de números `L` son menores que `N`. Por ejemplo,

```
?- lista_acotada_aux([1,5,3],7).
Yes
?- lista_acotada_aux([1,5,3],5).
No
```

y está definida por

```
lista_acotada_aux([],_).
lista_acotada_aux([X|L],N) :-
    X < N,
    lista_acotada_aux(L,N).
```

**Ejercicio 2** [2 puntos] Un número es libre de cuadrados si no es divisible por el cuadrado de ningún número mayor que 1. Definir la relación `libre_de_cuadrados(+N)` que se verifique si el número `N` es libre de cuadrados. Por ejemplo,

```
?- libre_de_cuadrados(30).
Yes
?- libre_de_cuadrados(12).
No
```

.....  
**Solución:** La definición de `libre_de_cuadrados` es

```
libre_de_cuadrados(N) :-
    M is floor(sqrt(N)),
    not((between(2,M,X), N mod (X*X) == 0)).
```

---

**Ejercicio 3** [2 puntos] Definir la relación `suma_libres_de_cuadrados(+L,-S)` que se verifique si S es la suma de los números libres de cuadrados la lista numérica L. Por ejemplo,

```
?- suma_libres_de_cuadrados([6,12,18,30],S).
S = 36
```

Nota: Dar dos definiciones, una con negación y otra con corte.

.....

**Solución:** La definición de `suma_libres_de_cuadrados` usando la negación es

```
suma_libres_de_cuadrados([],0).
suma_libres_de_cuadrados([X|L],S) :-
    libre_de_cuadrados(X),
    suma_libres_de_cuadrados(L,S1),
    S is X+S1.
suma_libres_de_cuadrados([X|L],S) :-
    not(libre_de_cuadrados(X)),
    suma_libres_de_cuadrados(L,S).
```

y la definición usando corte es

```
suma_libres_de_cuadrados([],0).
suma_libres_de_cuadrados([X|L],S) :-
    libre_de_cuadrados(X), !,
    suma_libres_de_cuadrados(L,S1),
    S is X+S1.
suma_libres_de_cuadrados([_X|L],S) :-
    % not(libre_de_cuadrados(_X)),
    suma_libres_de_cuadrados(L,S).
```

---

**Ejercicio 4** [2 puntos] Definir la relación `todos_iguales(+L)` que se verifique si todos los elementos de la lista L son iguales entre sí. Por ejemplo,

```
?- todos_iguales([a,a,a]).
Yes
?- todos_iguales([a,b,a]).
No
?- todos_iguales([]).
Yes
```

.....

**Solución:** La definición de `todos_iguales` es

```
todos_iguales([]).
todos_iguales([_]).
todos_iguales([X,X|L]) :-
    todos_iguales([X|L]).
```

---

**Ejercicio 5** [2 puntos] Un polígono se representa por su nombre y las longitudes de sus lados. Definir la relación `es_equilátero(+P)` que se verifica si el polígono P es equilátero (es decir, que todos sus lados son iguales). Por ejemplo,

```
?- es_equilátero(triángulo(4,4,4)).  
Yes  
?- es_equilátero(cuadrilátero(3,4,5,3)).  
No
```

.....  
**Solución:** La definición de `es_equilátero` es

```
es_equilátero(P) :-  
    P =.. [_|L],  
    todos_iguales(L).
```

---