

**Parte 1. (plazo: jueves 7 de septiembre)** Un conjunto puede representarse por una lista. Por ejemplo, el conjunto{3,1,2} puede representarse por la lista(3,lista(1,lista(2,None))). Escribir el módulo Conjunto con las siguientes funciones (siguiendo estrictamente la receta de diseño) que permiten realizar operaciones entre los conjuntos x e y:

| operación   | Encabezamiento función            |
|---|-----------------------------------|
| $x \cup y$  | def union(x,y):                   |
| $x \cap y$  | def inter(x,y):                   |
| cardinal de x   | cardinal(x):                      |
| ¿ a pertenece a x?  | def pertenece(a,x): #True o False |
| ¿x es conjunto?   | def esConjunto(x):                |
| Recuerde que los conjuntos no aceptan elementos repetidos |                                   |

**Parte 2. (plazo: martes 12 de septiembre)** Agregar al módulo Conjunto las siguientes funciones (siguiendo estrictamente la receta de diseño):

| operación                   | Encabezamiento función        |
|-----------------------------|-------------------------------|
| $x - y$                     | def resta(x,y):               |
| $x == y$                    | def igual(x,y): #True o False |
| $x \subset y$               | def sub(x,y): #True o False   |
| $x \supset y$               | def super(x,y): #True o False |
| string {..., ..., ..., ...} | def str(x):                   |

**Parte 3 (plazo: viernes 15 de septiembre)** Escribir la función recursiva Prueba que use el módulo Conjunto para establecer el diálogo indicado en el siguiente ejemplo:

```
ingrese elementos Conjunto A (o . para terminar)
elemento?a
elemento?b
elemento?c
elemento?d
elemento?e
elemento?.
A={a,b,c,d,e}
Cardinal=5
```

```
ingrese elementos Conjunto B (o . para terminar)
elemento?a
elemento?e
elemento?i
elemento?o
elemento?u
elemento?.
B={a,e,i,o,u}
Cardinal=5
```

```
Operación(+*-=<>.)?+
A unión B={a,b,c,d,e,i,o,u}
Cardinal=8
```

```
Operación(+*-=<>.)?*
A intersección B={a,e}
Cardinal=2
```

```
Operación(+*-=<>.)?-
A - B={b,c,d}
Cardinal=3
```

```
Operación(+*-=<>.)?=
A=B es No
```

```
Operación(+*-=<>.)?<
A<B es No
```

```
Operación(+*-=<>.)?>
A>B es No
```

```
Operación(+*-=<>.)?.
```

Otros conjuntos (si/no) ?no (en caso de responder 'si' debe repetirse todo el diálogo)

### Notas.

- Para leer un string utilice la función `raw_input(...)`. Por ejemplo: `p=raw_input('ingrese una palabra:')`
- Escriba y use las siguientes funciones recursivas (en el mismo módulo que la función Prueba)

-para leer elementos de un conjunto (hasta ingresar elemento con valor fin)

```
def leer(L=None, fin='fin') :
    #L: lista para acumular los elementos del conjunto
    #fin: str para indicar fin de los elementos
```

-para leer y realizar operaciones entre dos conjuntos A y B (hasta ingresar operación fin)

```
def operaciones(A,B, fin='.') :
```

### Importante.

- La tarea **puede** ser realizada en grupos de **dos** personas
- El plazo para inscribir el grupo vence impostergablemente el viernes 1 de septiembre y **cada integrante debe indicar el nombre de su compañero(a)** inscribiéndolo en la sección tareas de U-cursos.
- En caso de trabajar en grupo:
  - un integrante debe entregar la parte 1
  - el otro compañero(a) debe entregar la parte 2
  - y alguno de los dos integrantes debe entregar la parte 3.
  - Los dos compañeros(as) tendrán la misma nota por el trabajo completo.
- Si no se recibe inscripción de compañero(a) de grupo en el plazo indicado se entenderá que la tarea se realizará individualmente.