

Reto # 5

[Página Principal](#) / [Mis cursos](#) / [Reto#5](#) / [RETO # 5](#) / [Reto 5: Grupo P55](#)[Descripción](#)[Entrega](#)[Editar](#)[Ver entrega](#)

Reto 5: Grupo P55

Número máximo de ficheros: 1**Tipo de trabajo:** Individual**Coleccionista de vajillas**

Juan es coleccionista de vajillas. Cada pieza de la vajilla tiene un código. El colecciona vajillas de cerámica, porcelana, loza y cristal. Cada pieza de la vajilla tiene un código único. A veces se quiebran algunas de las piezas de una vajilla y necesita reemplazarlas para completar la vajilla. En otras ocasiones sus amigos le regalan vajillas que ya tenía, por lo cual tiene algunas vajillas repetidas.

Juan se unió a un grupo de coleccionistas de vajillas, por lo cual creó dos listas que le permitirán realizar intercambios: una lista con los códigos de las piezas que le faltan y otra lista con los códigos de las piezas que tiene repetidas para cambiar. Estos coleccionista solo cambian solo cambian piezas del mismo material.

Juan decidió crear una librería en Python que ayude a realizar dichos cambios. Esta librería se llama "vajilla" y consta como mínimo de las siguientes funciones (vale 1 punto):

1. (Vale 1) La función "material" que dada una lista de los tipos de todas las materiales (en la posición i-ésima de la lista está el material de dicha pieza), genera una lista con los materiales de las piezas sin repetición.

Por ejemplo si el inventario total tiene diez piezas y cada una es de los siguientes materiales:

```
['porcelana','cristal','ceramica','ceramica','ceramica','porcelana','cristal','ceramica','ceramica','ceramica']
```

la función debe retornar la lista

```
['porcelana','cristal','ceramica']
```

Nótese que los materiales aparecen solo una vez.

2. (Vale 1) La función "mefaltandelmateriale" que dada una lista con los códigos de las piezas que le faltan, la lista de los materiales de cada pieza y un material (en ese orden), les retorne una lista con los códigos de las piezas del material indicado que le faltan.

Por ejemplo si se ejecuta la función:

```
mefaltandelmateriale([1,3,6,8],  
['porcelana','cristal','ceramica','ceramica','ceramica','porcelana','cristal','ceramica','ceramica','ceramica'], 'ceramica')
```

debe retornar la lista

```
[3,8]
```

Nótese que los códigos de piezas empiezan en cero (0) y van hasta el nueve (9) en este ejemplo (siempre empiezan en cero).

Por ejemplo si ejecutan la

función:

```
mefaltandelmateriale([1,3,6,8],  
['porcelana','cristal','ceramica','ceramica','ceramica','porcelana','cristal','ceramica','ceramica','ceramica'], 'cristal')
```

Se debe retornar la lista:

```
[1,6]
```

3. (Vale 1) La función **"notengo"** que dada una lista con los códigos de las piezas que tiene otra persona y una lista con los códigos de las piezas que tengo retorna la lista con los códigos que le interesan a Juan de la otra persona.

Por ejemplo si se llama:

```
notengo([3,5,7,10,15,16],[4,10,5,8])
```

Se debe retornar la lista:

```
[3,7,15,16]
```

4. (Vale 1) Para simplificar la primera versión de la librería decidieron pensar que todas las piezas son del mismo material y que la lista que maneja cada persona indica los códigos de las piezas que tienen para cambiar y aquellos códigos que no están en dicha lista son los que necesitan. De esta manera deben crear una función **"puedocambiar"** que reciba los códigos de las piezas que tiene otra persona y la lista de códigos que Juan y que retorne los códigos de las piezas que puede cambiar.

Por ejemplo, llamar la función:

```
puedocambiar([3,5,7,10,15,16],[4,10,5,8])
```

Debe retornar

```
2
```

Porque a la otra persona solo le interesan dos códigos que tiene Juan: [4,8], mientras que a Juan le interesan cuatro piezas que la otra persona tiene: [3,7,15,16]

Entrada:

Este programa no requiere entrada. Ni generará salida. Se requiere que el estudiante genere un archivo con el nombre **vajilla.py** y que se respeten los nombres de las funciones dadas y sus parámetros.

[VPL](#)

Desarrollamos actividades de calidad en cursos, diplomados y eventos, relacionadas con las diferentes áreas de ingeniería, que contribuyen al desarrollo integral de las personas y las organizaciones, con el fin de crear nuevas oportunidades en el ámbito social y laboral.

[Leer más »](#)

Info

Unidad de educación continua

Cursos a la medida

Portal Universidad Nacional de Colombia

Formalización

Política de privacidad

Contacto

Calle 44 # 45-67 Bloque B5 piso 1 Unidad Camilo Torres Bogotá, Colombia

☎ Teléfono : (57) + 316 5000

✉ Correo electrónico : uec_fibog@unal.edu.co

Redes sociales



Derechos Reservados © 2021 - Universidad Nacional de Colombia

[Resumen de retención de datos](#)