



UNIVERSIDADE DA CORUÑA

FACULTADE DE INFORMÁTICA

Grao en Intelixencia Artificial

Fundamentos de Programación II – Curso 2024/25

Práctica 3 (Listas posicionales ordenadas)

Un concesionario de automóviles desea mantener actualizada su lista de modelos disponibles y los componentes necesarios para ensamblarlos. Al vender un automóvil, se debe comprobar que se dispone de cada uno de los componentes en cantidad suficiente. Como consecuencia, el inventario de piezas se actualiza eliminando aquellas que se hayan agotado, y el catálogo de modelos se modifica eliminando aquellos para los cuales no hay suficientes componentes.

Realizar una implementación con orientación a objetos que use las estructuras de datos:

- Inventario de piezas: Se leen de un fichero y se gestiona con una **Lista Posicional Ordenada**, donde se almacenan las piezas disponibles (nombre y cantidad).
- Catálogo de modelos de autos: Se leen de un fichero y se gestiona con un **Diccionario**, donde cada modelo tiene su **Lista Posicional Ordenada** de piezas.
- Pedidos de clientes: Se almacenan en un fichero en el orden en que llegan.

El programa funcionará sin interactividad con el usuario y ejecutará las siguientes tareas:

1. Leer un fichero con las piezas disponibles y almacenarlas en una lista ordenada. Mostrar un mensaje del tipo:

```
-----STOCK-----  
<Pieza1>: <Cantidad1> | <Pieza2>: <Cantidad2> | ...
```

2. Leer un fichero con el catálogo de modelos de automóviles (y piezas necesarias) y almacenarlos en un diccionario. Se asume que este fichero es correcto y que todos los modelos pueden ensamblarse inicialmente.

3. Mostrar el catálogo (ordenado) de modelos en pantalla con el formato:

```
-----CATALOGO-----  
<Modelo1>  
  <Pieza1>: <Cantidad1> | <Pieza2>: <Cantidad2> | ...
```

4. Procesar un fichero con los pedidos de clientes, siguiendo estos pasos:

- a) Mostrar un mensaje del tipo:

```
Nuevo pedido: Modelo - Cliente
```

- b) Verificar si el modelo está en el catálogo.

- Si no existe, imprimir:

```
Pedido NO atendido. Modelo <Modelo> fuera de catálogo.
```

- Si existe, mostrar las piezas necesarias:

```
<Modelo>  
  <Pieza1> - <Cantidad1>  
  <Pieza2> - <Cantidad2>
```

- Si las piezas están disponibles, reducir las cantidades en el inventario y mostrar:

```
Pedido <Modelo> atendido.
```

- Si falta alguna pieza, eliminar el modelo del catálogo y mostrar:

```
Pedido <Modelo> NO atendido. Faltan:
  <Pieza1> - <Cantidad que falta>
  <Pieza2> - <Cantidad que falta>
Eliminado: <Modelo>
```

- Si una pieza se agota, eliminarla del inventario y también eliminar los modelos que la usan, mostrando:

```
Eliminada: Pieza <Pieza>
Eliminado: Modelo <Modelo1> dependiente
```

c) Mostrar el estado de piezas disponibles:

```
-----STOCK-----
<Pieza1>: <Cantidad1> | <Pieza2>: <Cantidad2> | ...
```

5. Mostrar el catálogo (ordenado) de modelos actualizado después del proceso.

```
-----CATALOGO-----
<Modelo1>
  <Pieza1>: <Cantidad1> | <Pieza2>: <Cantidad2> |
```

La implementación deberá satisfacer los siguientes requisitos:

1. Se utilizarán las clases que sean necesarias con las variables de instancia y los métodos necesarios, respetando los principios de **orientación a objetos**.
2. Se implementará usando el **TAD Lista Posicional Ordenada**, con compatibilidad para `array_ordered_positional_list` y `linked_ordered_positional_list`.
3. Realizar la gestión con el tipo `list` y copiarla sobre una lista posicional supone práctica no presentada.

Se entregará un archivo zip con todo el código fuente (comentado con `#` y clases/métodos con docstrings), y un **documento PDF** que incluya un breve manual de usuario y una breve pero exhaustiva descripción de las fases de desarrollo realizadas.

IMPORTANTE:

- En cada archivo y en la primera página del PDF se indicará el nombre y login de ambos miembros de la pareja de prácticas.
- Los nombres de carpetas y ficheros de código `.py` seguirán la convención de nombres *snake-case* (palabras unidas por barra baja, sin espacios y en minúsculas, ej. `mi_fichero.py`).

Fecha límite de entrega: domingo 27 de abril de 2025 a las 23:55.

Dónde se entrega: en el apartado **Práctica 3** del Campus Virtual.

Quién entrega: sólo uno de los miembros de la pareja deberá entregar la práctica.