

"""El cine Atlas de Flores decide digitalizar el control de ventas de entradas para las películas infantiles que estarán en cartelera durante las vacaciones de invierno. Para la venta de las entradas se pide al usuario que ingrese la siguiente información:

Cantidad de entradas. (valor entero)

Nombre de película. (cadena de caracteres)

Se pide:

a) Generar un arreglo con las películas guardando la cantidad de entradas vendidas en un arreglo paralelo.

b) La carga finaliza cuando en cantidad se ingresa un 0.

c) En una función calcular y retornar la película que más vendió.

d) PLUS: buscar si la película ingresada ya existe en el arreglo, y si es así, acumular la cantidad de entradas para esa película."""

```
def cargar_datos(arre_pel, arre_cant):
```

```
    # Corrección sugerida: Inicializar la primera cantidad sin usar 0 o -1. Se toma directamente del teclado.
```

```
    cantidad = int(input("Ingrese la cantidad de entradas (0 para finalizar): "))
```

```
    # Corrección sugerida: Se elimina el comentario sobre el primer while incorrecto.
```

```
    while cantidad != 0: # Bucle principal para la carga (finaliza con 0)
```

```
        while cantidad < 0: # Validación de cantidad no negativa
```

```
            # Corrección sugerida: Si la cantidad es negativa, se pide de nuevo SIN pedir película
```

```
            print("Error - La cantidad no puede ser negativa.")
```

```
            cantidad = int(input("Error - Ingrese la cantidad de entradas (0 para finalizar): "))
```

```
            # Corrección sugerida: Se eliminan las siguientes líneas que cargaban datos inválidos dentro del bucle de error:
```

```
            # pelicula = input("Ingrese el nombre de la película : ").upper()
```

```
            # while pelicula == "":
```

```
            #     pelicula = input("Error - Ingrese el nombre de la película : ").upper()
```

```
            # arre_pel.append(pelicula)
```

```
            # arre_cant.append(cantidad)
```

```
    # Corrección sugerida: Solo si la cantidad es válida (>=0) y != 0, se pide la película y se procesa.
```

```
    if cantidad > 0:
```

```
        pelicula = input("Ingrese el nombre de la película : ").upper()
```

```
        while pelicula == "": # Validación de película no vacía
```

```
            pelicula = input("Error - Ingrese el nombre de la película : ").upper()
```

```
        # Corrección sugerida: Implementación del PLUS (punto d: buscar y acumular)
```

```
        # 1. Buscar si la película existe. Usaremos una variable 'indice' para almacenar la posición.
```

```
        indice = 0
```

```
        posicion_encontrada = -1
```

```
        while indice < len(arre_pel):
```

```
            if arre_pel[indice] == pelicula:
```

```
                posicion_encontrada = indice # Guarda la posición
```

```
                # Corrección sugerida: No se usa break, por lo que el bucle seguirá hasta el final, pero la posición_encontrada mantendrá el primer índice.
```

```
                indice = indice + 1
```

```
        if posicion_encontrada >= 0:
```

```
            # 2. Si existe (posición >= 0), acumular
```

```

        arre_cant[posicion_encontrada] = arre_cant[posicion_encontrada] +
cantidad
        print(f"Entradas acumuladas para: {pelicula}")
    else:
        # 3. Si no existe (posición == -1), agregar nuevos datos (punto a)
        arre_pel.append(pelicula)
        arre_cant.append(cantidad)
        print(f"Nueva película registrada: {pelicula}")

    # pedir el siguiente dato para la próxima iteración del while (punto b)
    cantidad = int(input("Ingrese la cantidad de entradas (0 para finalizar): "))

def calcular_max(arre_cant):
    # Corrección sugerida: Se agrega una validación para evitar el error IndexError si
    # la lista está vacía.
    if len(arre_cant) == 0:
        return -1 # Se usa un valor fuera de índice para indicar lista vacía

    max_ventas = arre_cant[0]
    pos_max = 0

    for i in range(len(arre_cant)):
        if arre_cant[i] > max_ventas:
            max_ventas = arre_cant[i]
            pos_max = i
    return pos_max # Retorna la posición de la película más vendida (punto c)

def mostrar_datos(arre_pel, arre_cant): # Corrección sugerida: Eliminar el parámetro
'arre_pre' o 'arre_pre' ya que no existe
    titulo1 = "Película"
    # Corrección sugerida: Se elimina el comentario sobre titulo2 (Precio) ya que no se
    # usa en la salida
    titulo2 = "Cantidad"
    print(f"{titulo1:<15}{titulo2:>15}") # Corrección sugerida: Ajustar el formato para
2 columnas
    for i in range(len(arre_pel)):
        # Corrección sugerida: Ajustar el formato para 2 columnas
        print(f"{arre_pel[i]:<15}{arre_cant[i]:>15}")

peliculas = []
cantidades = []
print("\n-----\n")
cargar_datos(peliculas, cantidades)
print("\n-----\n")
if len(peliculas) > 0:

    # Calcular el máximo una sola vez

    # Corrección sugerida: Llamada correcta a calcular_max para obtener el índice del
    máximo.
    indice_max = calcular_max(cantidades)

    # Corrección sugerida: Mostrar los datos antes de imprimir el máximo
    mostrar_datos(peliculas, cantidades)

    print("\n-----\n")
    # opcion 1, busca solo el primer maximo
    print("Resultado de buscar la primer pelicula mas taquillera:\n")
    # Corrección sugerida: Usar 'indice_max' que contiene la posición
    print(f"La pelicula mas taquillera fue {peliculas[indice_max]} con
{cantidades[indice_max]} entradas vendidas.")

```

```
else:
    print("No se registraron entradas vendidas.")
print("\n-----\n")
print("Fin del programa...")
print("\n-----\n")
```