

```

"""Se requiere un sistema para llevar control de libros de una biblioteca personal.
Para ello se registran los siguientes datos:
Nombre del libro
Genero ("Accion", "SCY-FY", "Novela")
Cantidad de paginas
La carga finaliza cuando en nombre del libro se ingresa "FIN" o se hayan cargado 50 libros.
Al finalizar la carga se debe mostrar:
A) Promedio de páginas del género Acción.
B) Porcentaje de libros del género Novela que tienen entre 100 y 500 páginas.
C) Nombre del libro con menos paginas.
D) Validar los datos con criterio logico."""
# Sistema de control de libros para biblioteca personal. Inicialización de variables
acumuladoras y contadores
suma_paginas_accion = 0 # Acumula el total de páginas de libros de Acción
libros_accion = 0 # Cuenta cuántos libros de Acción se registraron
libros_novela = 0 # Cuenta cuántos libros de Novela se registraron
libros_SCYFY = 0 # Cuenta cuántos libros de SCY-FY se registraron
cont_libros = 0 # Contador general de libros registrados
cont_novela_rango = 0 # Cuenta libros de Novela con 100-500 páginas

print("\n-----Sistema de Registro de Libros-----") # Mensajes informativos al
usuario
print("\nLa carga finaliza al registrar 50 libros o al escribir FIN en el nombre del
libro.")
print("Géneros: ACCION, SCY-FY, NOVELA.")

nombre_libro = input("\nIngrese el nombre del libro: \t").upper() # Solicitar el nombre del
primer libro antes de entrar al bucle

while nombre_libro == "": # Validar que el nombre del primer libro no esté vacío
    nombre_libro = input("\nError: Ingrese el nombre del libro: \t").upper()

while cont_libros < 50 and nombre_libro != 'FIN': # Bucle principal: mientras no se
alcancen 50 libros Y no se escriba "FIN"

    genero_libro = input("Ingrese el género (ACCION, SCY-FY, NOVELA): \t").upper() #
Solicitar el género del libro
    while genero_libro != "ACCION" and genero_libro != "SCY-FY" and genero_libro != "NOVELA": # Validar el género
        genero_libro = input("Error: Ingrese el género (ACCION, SCY-FY, NOVELA):
\t").upper()

    paginas = int(input("Ingrese la cantidad de páginas: \t")) # Solicitar la cantidad de
páginas del libro
    while paginas <= 0: # Validar que la cantidad de páginas sea mayor que 0
        paginas = int(input("Error: Ingrese la cantidad de páginas: \t"))

    if genero_libro == "ACCION": # Procesar libros de género ACCION
        suma_paginas_accion += paginas # Acumular páginas para calcular promedio después
        libros_accion += 1 # Incrementar contador de libros de Acción

```

```

if genero_libro == "NOVELA":
    libros_novela += 1 # Contar todas las novelas registradas.
    if (paginas >= 100 and paginas <= 500): # Procesar el género NOVELA de rango de
100-500 páginas
        cont_novela_rango += 1 # Contar solo las novelas dentro del rango especificado

if cont_libros == 0: # Si es el primer libro
    min_paginas = paginas # Inicializar el mínimo con las páginas del primer libro
    nombre_min = nombre_libro # Guardar el nombre del primer libro
else: # Para los libros siguientes
    if paginas < min_paginas: # Si las páginas actuales son menores que el mínimo
        min_paginas = paginas # Actualizar el mínimo
        nombre_min = nombre_libro # Actualizar el nombre del libro con menos páginas

if genero_libro == "SCY-FY": # Procesar el género SCY-FY y encontrar el que tiene más
páginas
    if libros_SCYFY == 0: # Si es el primer libro de SCY-FY
        max_paginas = paginas # Inicializar el máximo con este libro
        nombre_max = nombre_libro # Guardar su nombre
    else: # Para los siguientes libros de SCY-FY
        if paginas > max_paginas: # Si tiene más páginas que el máximo actual
            max_paginas = paginas # Actualizar el máximo
            nombre_max = nombre_libro # Actualizar el nombre
    libros_SCYFY += 1 # Incrementar contador de libros SCY-FY

cont_libros += 1 # Incrementar el contador general de libros procesados

nombre_libro = input("\nIngrese el nombre del libro: \t").upper() # Solicitar el nombre
del siguiente libro
while nombre_libro == "": # Validar que el nuevo nombre no esté vacío
    nombre_libro = input("\nError: Ingrese el nombre del libro: \t").upper()

print(f"\nCarga finalizada. Se registraron {cont_libros} libros.") # Mensaje indicando que
la carga finalizó

print("\n-----Resultados-----") # Título de la sección de resultados

if cont_libros > 0: # Mostrar resultados solo si se registró al menos un libro
    # 1. Calcular y mostrar el promedio de páginas del género Acción
    if libros_accion > 0: # Solo si hay libros de Acción
        promedio_accion = suma_paginas_accion / libros_accion # Calcular promedio
        print(f"\nPromedio de páginas del género Acción: \t{promedio_accion:.2f}")
    else: # Si no hay libros de Acción
        print("\nPromedio de páginas del género Acción: \t(No hay libros de Acción)")
    # 2. Calcular y mostrar el porcentaje de libros de Novela con 100-500 páginas
    if libros_novela > 0: # Esta condición siempre será False
        porcentaje_novela = (cont_novela_rango / libros_novela) * 100 # Calcular
porcentaje
        print(f"\nPorcentaje de libros del género Novela con entre 100 y 500 páginas:
\t{porcentaje_novela:.2f}%")
    else:
        print("\nPorcentaje de libros del género Novela con entre 100 y 500 páginas: \t(No
hay libros de Novela)")
    # 3. Mostrar el libro con menos páginas

```

```
if min_pginas: # Si existe un valor para min_pginas (siempre será True si
cont_libros > 0)
    print(f"\nNombre del libro con menos páginas: \t'{nombre_min}' ({min_pginas} páginas)")
else:
    # Este mensaje es incorrecto, debería referirse al libro con menos páginas, no a
SCY-FY
    print("\nNo se registraron libros del género SCI-FY.")
# Mostrar el libro de SCY-FY con más páginas
if libros_SCYFY > 0: # Si se registraron libros de SCY-FY
    print(f"\nEl máximo de SCI-FY es: \t'{nombre_max}' ({max_pginas} páginas)")
else: # Si no hay libros de SCY-FY
    print("\nNo se registraron libros del género SCI-FY.")
else: # Si no se registró ningún libro
    print("No se registraron libros. Fin del programa.")
```