



PRACTICA 1

Manual Técnico

Richard Alexandro Marroquin

202102894

Universidad de San Carlos de
Guatemala

Proyecto 1

Requisitos del sistema

Para desarrollar la aplicación, se necesitará lo siguiente:

Python 3.x instalado en el sistema.

Graphviz instalado en el sistema (opcional).

Funcionalidades del programa

La aplicación tendrá las siguientes funcionalidades:

Mostrar la lista de películas: La aplicación mostrará la lista completa de películas ordenadas alfabéticamente por el nombre de la película.

Buscar películas por género: La aplicación permitirá buscar películas por género. El usuario ingresará el género y la aplicación mostrará la lista de películas que pertenecen a ese género.

Buscar películas por año de estreno: La aplicación permitirá buscar películas por año de estreno. El usuario ingresará el año y la aplicación mostrará la lista de películas que se estrenaron en ese año.

Mostrar detalles de una película: La aplicación permitirá mostrar los detalles de una película. El usuario ingresará el nombre de la película y la aplicación mostrará el nombre de la película, los actores que participan en ella, el año de estreno y el género.

Generar un diagrama con las relaciones entre las películas: La aplicación generará un diagrama utilizando Graphviz que mostrará las relaciones existentes entre las películas. Por ejemplo, si dos películas tienen un actor en común, habrá una línea que las conecte.

Implementación del programa

Para implementar el programa, se pueden seguir los siguientes pasos:

Leer el archivo .lfp y guardar la información de las películas en una estructura de datos adecuada. Se puede utilizar una lista de diccionarios, donde cada diccionario representa una película con sus diferentes campos (nombre, actores, año de estreno, género).

Implementar la funcionalidad de mostrar la lista de películas ordenadas alfabéticamente. Para esto, se puede utilizar la función `sorted()` de Python para ordenar la lista de películas por el nombre de la película.

Implementar la funcionalidad de buscar películas por género. Para esto, se puede recorrer la lista de películas y buscar las películas que pertenecen al género ingresado por el usuario.

Implementar la funcionalidad de buscar películas por año de estreno. Para esto, se puede recorrer la lista de películas y buscar las películas que se estrenaron en el año ingresado por el usuario.

Implementar la funcionalidad de mostrar los detalles de una película. Para esto, se puede recorrer la lista de películas y buscar la película que tiene el nombre ingresado por el usuario.

Implementar la generación del diagrama utilizando.

El programa que he propuesto tiene como objetivo procesar la información contenida en un archivo de texto plano que contiene listas de valores separados por punto y coma que representan información sobre películas. La lógica del programa se divide en varias funciones que realizan diferentes tareas:

La función `leer_archivo` se encarga de leer el archivo y convertir cada línea en un diccionario que representa una película. Para ello, utiliza la biblioteca estándar de Python `csv` para leer el archivo y separar los valores por punto y coma. La información de cada película se guarda en un diccionario con claves como "nombre", "actores", "año" y "generos".

La función `mostrar_lista_peliculas` se encarga de mostrar la lista de películas ordenadas alfabéticamente. Para ello, utiliza la función `sorted` de Python para ordenar la lista de películas por nombre, y luego muestra el nombre y el año de estreno de cada película.

La función `buscar_por_genero` se encarga de buscar películas por género. Recibe como parámetro la lista de películas y un género, y devuelve una lista de películas que pertenecen al género buscado. Para ello, itera sobre la lista de películas y verifica si el género buscado está en la lista de géneros de cada película.

La función `buscar_por_año` se encarga de buscar películas por año de estreno. Recibe como parámetro la lista de películas y un año, y devuelve una lista de películas que fueron estrenadas en el año buscado. Para ello, itera sobre la lista de películas y verifica si el año de estreno de cada película es igual al año buscado.

La función `mostrar_detalle_pelicula` se encarga de mostrar los detalles de una película específica. Recibe como parámetro la lista de películas y el nombre de la película buscada, y muestra el nombre, los actores, el año de estreno y los géneros de la película correspondiente. Para ello, itera sobre la lista de películas y busca la película con el nombre buscado.

La función `generar_diagrama` se encarga de generar un diagrama utilizando la biblioteca `Graphviz` de Python que muestra las relaciones entre las películas. Cada película es representada por un nodo, y las relaciones entre las películas se representan mediante aristas que conectan los nodos correspondientes. En particular, cada actor que participa en una película se conecta al nodo de la película, y cada género al que pertenece una película se conecta al nodo del género.

En cuanto a los paradigmas utilizados en este programa, se puede decir que se utilizó principalmente el paradigma de programación funcional. En particular, las funciones que implementan la lógica del programa se diseñaron de forma modular y con un enfoque en la inmutabilidad de los datos, lo que significa que se evitó modificar los datos originales y se prefirió devolver nuevos datos en su lugar. También se utilizó el paradigma de programación estructurada, ya que el programa se organizó en funciones que realizan tareas específicas y se evitaron los saltos incondicionales y los bucles infinitos. Además, se utilizaron algunas características del paradigma de programación orientada a objetos, como el uso de diccionarios para representar las películas y el uso de métodos de objetos para acc

Menú del programa

```
def Mostrar_peliculas():
    print("*****Películas*****")
    for pelicula in lst_peliculas:
        print("Nombre: ", pelicula.nombre)
        print("Año: ", pelicula.year)
        print("Genero: ", pelicula.genero)
        print("*****")
    input("Enter para continuar")
```

Mostrar Películas

```
def Mostrar_actores():
    seleccion = None
    print("*****Selección de película*****")
    for pelicula in lst_peliculas:
        print(str(lst_peliculas.index(pelicula)+1)+".- "+pelicula.nombre)
    print("*****")
    while True:
        try:
            seleccion = int(input("Selecciona una película: ")) - 1
            break
        except:
            print("Selección no válida")

    if abs(seleccion) > len(lst_peliculas):
        print("Selección no válida")
        return

    pelicula_seleccionada = lst_peliculas[seleccion]
    print("Actores: ")
    for actor in pelicula_seleccionada.actores:
        print(" "+str(pelicula_seleccionada.actores.index(actor)+1)+".- "+actor)
    input("Enter para continuar")
```

Filtros

```
def Filtrado_por_actor():
    try:
        actor_seleccionado = input("Escribe el nombre de un actor para desplegar
        existe = lst_actores.index(actor_seleccionado.strip())
    except:
        print("No se ha encontrado un actor llamado así, revisa mayúsculas y min
        return

    print("Películas en las que participa "+actor_seleccionado+": ")

    for pelicula in lst_peliculas:
        if actor_seleccionado in pelicula.actores:
            print(" Nombre: ",pelicula.nombre)
            print(" Año: ", pelicula.year)
            print(" Genero: ", pelicula.genero)
            print("*****")
    input("Enter para continuar")
```