



Evidencia 2. Proyecto integrador

Santiago Patricio Gómez Ochoa A01735171

Instituto Tecnológico y de Estudios Superiores de Monterrey

Programación Orientada a Objetos TC1030.303

Grupo 303

Periodo 3

14 de junio de 2024

ÍNDICE

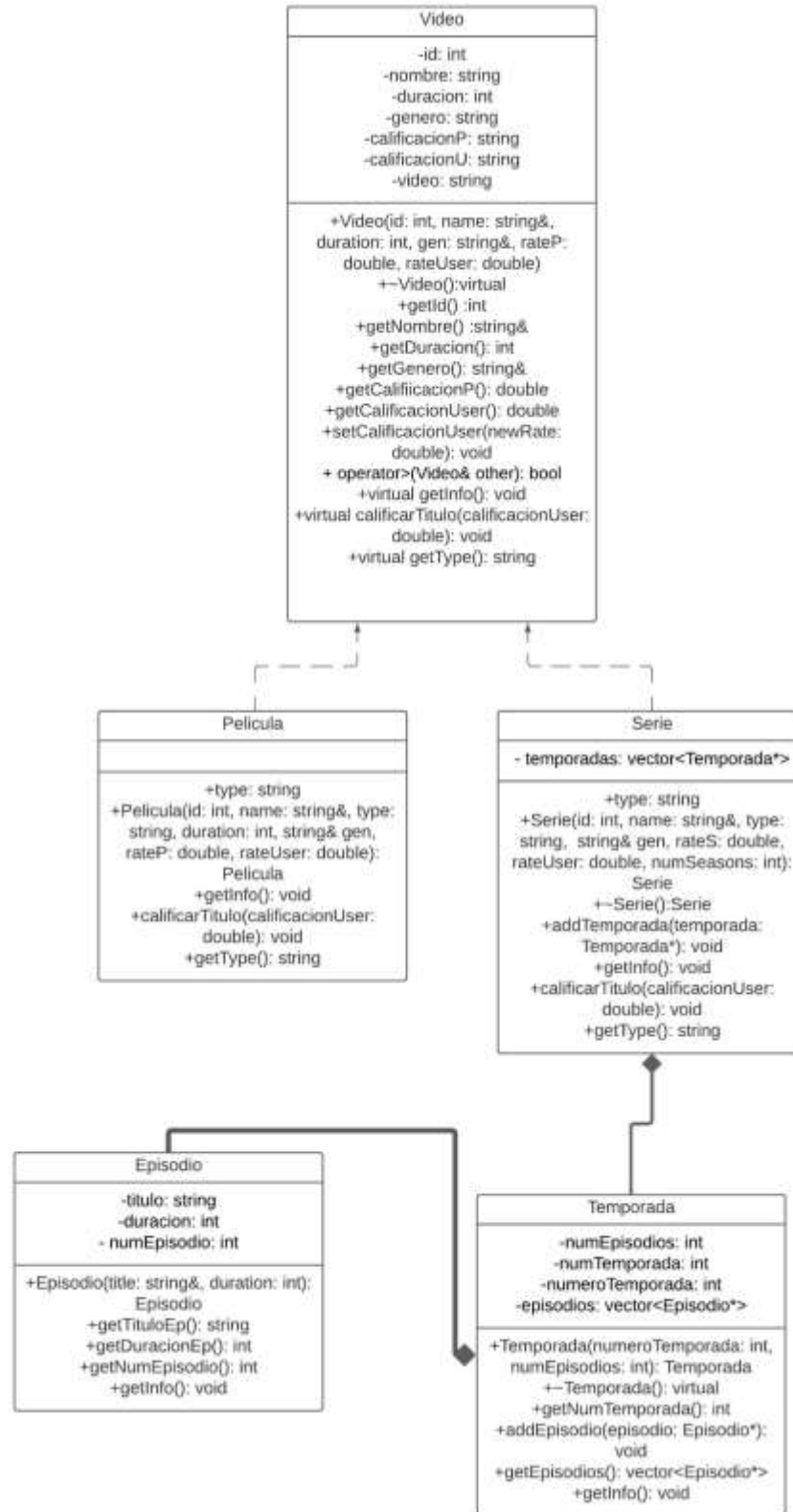
- **INTRODUCCIÓN** ----- Página 3
- **DIAGRAMA DE CLASES UML** ----- Página 4
- **EJEMPLO DE EJECUCIÓN** ----- Página 6
- **DISEÑO DEL PROGRAMA** ----- Página 16
- **LÓGICA PROPUESTA** ----- Página 17
- **CASOS DE ERROR** ----- Página 18
- **CONCLUSION PERSONAL** ----- Página 19
- **REFERENCIAS** ----- Página 20

INTRODUCCIÓN

La situación problema plantea el desafío de diseñar un sistema de streaming de video bajo demanda que pueda gestionar tanto películas como series, brindando a los usuarios la capacidad de explorar y seleccionar contenido de acuerdo con sus preferencias. Mediante el diseño propuesto de las clases involucradas, se busca diseñar un menú con distintas opciones, en el que se pueda observar un catálogo de series y películas. El usuario podrá acceder a su información utilizando el número de identificación relacionado a cada uno de los videos, y poder ver su duración, calificación, y en el caso e las series, sus episodios por temporada al igual que sus títulos y sus respectivas duraciones. Incluso tendrá la opción de asignarle una calificación propia a cada video. Este escenario implica el modelado de clases que representen adecuadamente la estructura de los videos, la gestión de calificaciones y la navegación dentro del catálogo de contenido. La implementación eficiente de estos requisitos mediante conceptos de programación orientada a objetos garantizará un sistema flexible y escalable, capaz de adaptarse a futuras ampliaciones y mejoras.

DIAGRAMA DE CLASES UML

Para facilitar el diseño del programa, se diseñó primero un diagrama de clases UML. Esto es para tener una idea más clara de cómo empezar a diseñar las clases padres y las clases derivadas, al igual que comenzar con el diseño y la estructura del programa, al visualizar los atributos, métodos y constructores de cada clase. Este es el diagrama que se realizó:



Este diagrama indicaría que Video es la clase base y Pelicula y Serie heredan de ella. Esto significa que Pelicula y Serie tienen todas las características de Video, incluyendo id, nombre, duracion, genero, calificacionP, calificacionU, y video. Además, añaden sus propias características y métodos. Serie se compone de Temporadas. Una Serie no puede existir sin sus Temporadas. Temporada se compone de Episodios, indicando que una Temporada no existe sin Episodios, y Serie no puede existir sin Temporada. El diagrama también muestra las relaciones entre las diferentes clases y sus métodos. Por ejemplo, la clase Video tiene un método llamado getInfo que devuelve información sobre el video. La clase Pelicula tiene un método llamado calificarTitulo que permite al usuario calificar el título de la película.

EJEMPLO DE EJECUCIÓN

```

----- Bienvenido -----
Selecciona una de las opciones del menú||

----- MENÚ -----
1. Mostrar todos los videos
2. Mostrar Películas
3. Mostrar Series
0. Salir

Seleccione una opción: 1

Pulsa la tecla <D> para ver el catálogo en general
Pulsa la tecla <C> para ver el catálogo ordenado por calificación de IMDB
Pulsa la tecla <G> para ver el catálogo ordenado por género
Pulsa la tecla <Q> para regresar al menú principal
d

----- CATALOGO -----

----- Películas -----
1. El Caballero de la Noche
2. Batman v Superman: El origen de la Justicia
3. Rebel Moon - Parte 1: La niña del fuego
4. Seven
5. La chica del tren
6. El muñeco de nieve
7. La Lista de Schindler
8. Climax
9. El incidente
10. Bastardos sin Gloria
11. The Mean One

```

```

----- Series -----
12. Arcane
13. Invencible
14. La Maldición de Hill House
15. The I-Land
16. Breaking Bad
17. La Niebla

Ingresa el ID de algún video para ver su información o 0 para regresar: 4

ID: 4
Título: Seven
Duración: 130 min
Género: Misterio
Calificación IMDB: 4.5
No has calificado este título

¿Desea calificar este título?
Pulse < 1 > para confirmar
Pulse < 0 > para regresar
1

Ingresa tu calificación (de 0 a 5): 4.5
Calificación guardada correctamente.

```

```

ID: 4
Título: Seven
Duración: 130 min
Género: Misterio
Calificación IMDB: 4.5
Tu calificación: 4.5

Pulsa la tecla <D> para ver el catálogo en general
Pulsa la tecla <C> para ver el catálogo ordenado por calificación de IMDB
Pulsa la tecla <G> para ver el catálogo ordenado por género
Pulsa la tecla <Q> para regresar al menú principal
c

----- Peliculas -----

Calificación 5:
7. La Lista de Schindler - Calificación: 5
10. Bastardos sin Gloria - Calificación: 5

Calificación 4:
1. El Caballero de la Noche - Calificación: 4.6
4. Seven - Calificación: 4.5

```

Calificacion 3:

- 2. Batman v Superman: El origen de la Justicia - Calificacion: 3.7
- 8. Climax - Calificacion: 3.7
- 5. La chica del tren - Calificacion: 3.5
- 3. Rebel Moon - Parte 1: La niña del fuego - Calificacion: 3.3

Calificacion 2:

- 9. El incidente - Calificacion: 2.3
- 6. El muñeco de nieve - Calificacion: 2

Calificacion 1:

- 11. The Mean One - Calificacion: 1

----- Series -----

Calificacion 4:

- 12. Arcane - Calificacion: 4.5
- 16. Breaking Bad - Calificacion: 4.5
- 13. Invencible - Calificacion: 4.4
- 14. La Maldición de Hill House - Calificacion: 4.3

Calificacion 2:

- 15. The I-Land - Calificacion: 2.2

Calificacion 1:

- 17. La Niebla - Calificacion: 1

Ingresar el ID de algún video para ver su información o 0 para regresar: 16

ID: 16

Título: Breaking Bad

Género: Drama

Calificación IMDB: 4.5

No has calificado este título

Número de temporadas: 2

Temporada 1:

1. Principio del fin -- 58 min
2. El gato en la bolsa -- 48 min
3. Y la bolsa en el río -- 48 min
4. Cíncer -- 48 min
5. Materia Gris -- 48 min
6. Un loco pulido de nada -- 48 min
7. Acuerdo no violento -- 47 min

Temporada 2:

1. Siete con treinta y siete -- 47 min
2. A las brasas -- 48 min
3. Picadura de una abeja muerta -- 47 min
4. Abajo -- 47 min
5. Rotura -- 47 min
6. Cucull -- 47 min
7. Negro y azul -- 47 min
8. ¡Llama a Saul! -- 47 min
9. Cuatro días afuera -- 47 min
10. Terminado -- 47 min

¿Desea calificar este título?

Pulse < 1 > para confirmar

Pulse < 0 > para regresar

1

Ingresa tu calificación (de 0 a 5): 5

Calificación guardada correctamente.

ID: 16

Título: Breaking Bad

Género: Drama

Calificación IMDB: 4.5

Tu calificación: 5

Número de temporadas: 2

Temporada 1:

1. Principio del fin -- 58 min
2. El gato en la bolsa -- 48 min
3. Y la bolsa en el ríjio -- 48 min
4. Cíñcer -- 48 min
5. Materia Gris -- 48 min
6. Un loco puñado de nada -- 48 min
7. Acuerdo no violento -- 47 min

Temporada 2:

1. Siete con treinta y siete -- 47 min
2. A las brasas -- 48 min
3. Picadura de una abeja muerta -- 47 min
4. Abajo -- 47 min
5. Rotura -- 47 min
6. Cucí -- 47 min
7. Negro y azul -- 47 min
8. Tíllama a Saul! -- 47 min
9. Cuatro dñjas afuera -- 47 min
10. Terminado -- 47 min

Pulsa la tecla <D> para ver el catílogo en general

Pulsa la tecla <C> para ver el catílogo ordenado por calificaciín de IMDB

Pulsa la tecla <G> para ver el catílogo ordenado por gñero

Pulsa la tecla <Q> para regresar al mení principal

g

----- CATALOGO -----

----- Peliculas -----

Gñero Acciín:

1. El Caballero de la Noche
2. Batman v Superman: El origen de la Justicia
3. Rebel Moon - Parte 1: La niña del fuego

Género Misterio:

4. Seven
5. La chica del tren
6. El muñeco de nieve

Género Drama:

7. La Lista de Schindler
8. Climax
9. El incidente
10. Bastardos sin Gloria

Género Terror:

11. The Mean One

----- Series -----

Género Acción:

12. Arcane
13. Invencible

Género Misterio:

17. La Niebla

Género Drama:

15. The I-Land
16. Breaking Bad

Género Terror:

14. La Maldición de Hill House

Ingresar el ID de algún video para ver su información o 0 para regresar: 0

Pulsar la tecla <D> para ver el catálogo en general

Pulsar la tecla <C> para ver el catálogo ordenado por calificación de IMDB

Pulsar la tecla <G> para ver el catálogo ordenado por género

Pulsar la tecla <Q> para regresar al menú principal

q

```

----- MENÚ -----
1. Mostrar todos los videos
2. Mostrar Películas
3. Mostrar Series
0. Salir

Seleccione una opción: 2

Pulse la tecla <D> para ver todo el catálogo de películas
Pulse la tecla <C> para ver el catálogo de películas ordenado por calificación de IMDb
Pulse la tecla <G> para ver el catálogo de películas ordenado por género
Pulse la tecla <Q> para regresar al menú principal
g

Géneros:
Acción:
1. El Caballero de la Noche
2. Batman v Superman: El origen de la Justicia
3. Rebel Moon - Parte 1: La niña del fuego

Misterio:
4. Seven
5. La chica del tren
6. El muñeco de nieve

Drama:
7. La lista de Schindler
8. Climax
9. El incidente
10. Bastardos sin Gloria

Terror:
11. The Mean One

Ingresa el ID de alguna película para ver su información o 0 para regresar: 4

```

```

ID: 4
Título: Seven
Duración: 130 min
Género: Misterio
Calificación IMDB: 4.5
Tu calificación: 4.5

¿Desea calificar este título?
Pulse < 1 > para confirmar
Pulse < 0 > para regresar
0

Pulse la tecla <D> para ver todo el catálogo de películas
Pulse la tecla <C> para ver el catálogo de películas ordenado por calificación de IMDB
Pulse la tecla <G> para ver el catálogo de películas ordenado por género
Pulse la tecla <Q> para regresar al menú principal
q
----- MENÚ -----
1. Mostrar todos los videos
2. Mostrar Películas
3. Mostrar Series
0. Salir

Seleccione una opción: 3

Pulse la tecla <D> para ver todo el catálogo de series
Pulse la tecla <C> para ver el catálogo de series ordenado por calificación de IMDB
Pulse la tecla <G> para ver el catálogo de series ordenado por género
Pulse la tecla <Q> para regresar al menú principal
c

```

```

----- Serie -----

Calificacion 4:
12. Arcane - Calificacion: 4.5
16. Breaking Bad - Calificacion: 4.5
13. Invencible - Calificacion: 4.4
14. La Maldición de Hill House - Calificacion: 4.3

Calificacion 2:
15. The I-Land - Calificacion: 2.2

Calificacion 1:
17. La Niebla - Calificacion: 1

Ingresa el ID de alguna serie para ver su información o 0 para regresar: 15

ID: 15
Título: The I-Land
Género: Drama
Calificación IMDB: 2.2
No has calificado este título
Número de temporadas: 1

Temporada 1:
1. Magnífico nuevo mundo -- 42 min
2. Espléndidos palacios -- 43 min
3. Tramoya Instrumental -- 39 min
4. Cuántas criaturashermosas -- 41 min
5. Torres orladas de nubes -- 41 min
6. El mismísimo globo -- 37 min
7. El abismo oscuro del pasado -- 38 min

¿Desea calificar este título?
Pulse < 1 > para confirmar
Pulse < 0 > para regresar
1

```

Ingresa tu calificaci|n (de 0 a 5): 0.1
 Calificaci|n guardada correctamente.

ID: 15
 T|tulo: The I-Land
 G|nero: Drama
 Calificaci|n IMDB: 2.2
 Tu calificaci|n: 0.1
 N|mero de temporadas: 1

Temporada 1:

1. Magn|fico nuevo mundo -- 42 min
2. Espl|ndidos palacios -- 43 min
3. Tramoya Instrumental -- 39 min
4. Cu|ntas criaturashermosas -- 41 min
5. Torres orladas de nubes -- 41 min
6. El mism|simo globo -- 37 min
7. El abismo oscuro del pasado -- 38 min

Pulsa la tecla <D> para ver todo el cat|logo de series

Pulsa la tecla <C> para ver el cat|logo de series ordenado por calificaci|n de IMDB

Pulsa la tecla <G> para ver el cat|logo de series ordenado por g|nero

Pulsa la tecla <Q> para regresar al men|| principal

q

----- MEN|Ü -----

1. Mostrar todos los videos
2. Mostrar Pel|culas
3. Mostrar Series
0. Salir

Seleccione una opci|n: 0

Cerrando sesi|n...

DISEÑO DEL PROGRAMA

En sí en este programa se utilizaron 4 conceptos importantes: herencia, polimorfismo, composición y sobrecarga de operadores. Se agregó herencia en la clase Video como clase base, y se crearon las clases Pelicula y Serie que heredan de ella. Esto se hizo para que Pelicula y Serie compartan las características y métodos comunes de Video, como id, nombre, duracion, genero, calificacionP, calificacionU, y video. Se utilizó la palabra clave : para indicar la herencia, por ejemplo: `class Pelicula : public Video`. De esta manera, Pelicula hereda todos los miembros de Video y puede agregar sus propias características y métodos.

Se agregó polimorfismo en los métodos `getInfo()` y `calificarTitulo()` de las clases Video, Pelicula y Serie. Esto se hizo para que cada clase pueda tener su propia implementación de estos métodos, pero con la misma firma. Por ejemplo, en la clase Video, el método `getInfo()` devuelve información general sobre el video, mientras que en la clase Pelicula, el método `getInfo()` devuelve información específica sobre la película, como el título y la calificación. Se utilizó la palabra clave `virtual` para indicar que los métodos `getInfo()` y `calificarTitulo()` son virtuales y pueden ser sobrecargados en las clases derivadas. Se agregó composición en la clase Serie con la clase Temporada, y en la clase Temporada con la clase Episodio. Esto se hizo para que una Serie pueda tener varias Temporadas, y cada Temporada pueda tener varios Episodios.

Se utilizó un vector de punteros a Temporada en la clase Serie para almacenar las temporadas, y un vector de punteros a Episodio en la clase Temporada para almacenar los episodios. Se agregaron métodos para agregar y eliminar temporadas y episodios, como `addTemporada()` y `addEpisodio()`, para que se puedan manejar las relaciones entre las clases. Se agregó la sobrecarga de operadores para permitir la comparación de objetos Video por su calificación.

Se agregó la sobrecarga del operador `>` en la clase Video para permitir la comparación de dos objetos Video por su calificación. Para agregar la sobrecarga del operador `>`, se creó un método en la clase Video con la firma `bool operator>(const Video& other) const;`. Este método toma otro objeto Video como parámetro y devuelve un booleano que indica si el objeto actual tiene una calificación mayor que el objeto pasado como parámetro. Luego, se

implementó el método para que compare las calificaciones de los dos objetos Video y devuelva true si la calificación del objeto actual es mayor que la del objeto pasado como parámetro, y false en caso contrario. En resumen, esas son las partes en las que se utilizaron herencia, polimorfismo, composición y sobrecarga de operadores. El resto del código consiste en la creación de los objetos, funciones necesarias para el funcionamiento del código, y en darle un formato más estético al menú.

LÓGICA PROPUESTA

La lógica del programa en general es un menú principal en el que hayan 4 opciones:

- Mostrar todos los videos
- Mostrar Películas
- Mostrar Series
- Salir

En la primera opción, se mostraría el catálogo completo con todas las películas y todas las series disponibles, mostrando sus ID y sus títulos. El usuario podría ingresar en la terminal cualquiera de los ID para visualizar la información de cada video. En el caso de las películas, se muestra su ID, su título, duración en minutos, género, la calificación que le dan los medios, y la calificación del usuario. En caso de que el usuario no la haya calificado, el programa lo indica. Por otro lado, para las series aparece su ID, título, género, calificación de los medios, calificación del usuario (igual indica si no se ha calificado) y cantidad de temporadas, seguido de sus episodios por temporada, en el que vienen título y duración de cada uno. De igual manera, se puede elegir entre ver todo el catálogo en general, ordenarlo por calificación, u ordenarlo por género.

En cuanto a la opción 2 y 3, ocurre lo mismo, sin embargo, en la opción 2 sólo se visualizan las películas y en la 3 sólo las series. Cabe recalcar que para las series, el ID se queda igual que en el catálogo completo, en lugar de comenzar por el número 1 otra vez. Esto se realiza por simplicidad a la hora de elegir un ID, ya que se consideró que era mejor que todos los videos tuvieran un ID fijo. De igual manera, se consideró como una mejor opción tener 2 calificaciones, y que la calificación de los medios fuera fija desde un inicio (y no el

promedio de la calificación de los episodios), ya que se pensó que así se igualaría más a un formato de una plataforma de streaming real, en donde rara vez se califican individualmente a los episodios, en lugar de a la serie en general. Además, se tendrían que calificar todos los episodios para que la serie tuviera una calificación, o tomar el promedio de unos pocos que sí estén calificados. Sin embargo, no sería una calificación muy precisa.

CASOS DE ERROR

Los siguientes casos se identificaron como las situaciones en las que el programa no funcionaría:

- **Entrada de datos inválida:** Si el usuario proporciona datos inválidos al ingresar la calificación de un video, como un valor fuera del rango permitido (0 a 5), el programa puede comportarse de manera inesperada o mostrar mensajes de error inadecuados. Sin embargo, más que el programa deje de funcionar, solamente regresa a preguntar si se desea ver el catálogo completo, por calificación o por género.
- **Memoria insuficiente:** Si el sistema se queda sin memoria mientras está en ejecución debido a una cantidad excesiva de datos o a una mala gestión de la memoria dinámica, el programa podría fallar o volverse extremadamente lento.
- **Errores de lógica:** Si hay errores de lógica en el código, como condiciones incorrectas en las estructuras de control o cálculos incorrectos en las funciones, el programa puede producir resultados incorrectos o comportarse de manera impredecible. Este sería más un error a la hora de programarlo que un error durante la ejecución.
- **Dependencias externas:** Ya que lo que sería programa *main* depende de un archivo independiente, y uno de ellos también depende de otro archivo, se necesita que el usuario tenga los otros dos archivos en la misma carpeta donde tenga el *main*. De lo contrario, no funcionará el programa.

CONCLUSIÓN PERSONAL

Considero que este fue un reto bastante interesante y entretenido para poner en práctica los nuevos conceptos aprendidos. Si bien es cierto que tuve que investigar mucho acerca de las sintaxis, los propósitos de los punteros y sobre todo de la sobrecarga de operadores, considero que logré aplicarlos de una forma muy interesante a mi código. También considero que es posible que mi forma de abordar el problema, por ejemplo, para la creación de los videos y la definición de las funciones en el *main*, tal vez haya hecho que tuviera que realizar cosas más complicadas y por eso mi *main* es tan extenso. Sin embargo, aun así considero que pude comprender bien el objetivo y los temas requeridos para cumplirlo.

REFERENCIAS

C++ Pointers. (2024). *Geeks for Geeks*. Recuperado el 5 de junio de 2024, de

<https://www.geeksforgeeks.org/cpp-pointers/>

C++ Polymorphism. (2023). *Geeks for Geeks*. Recuperado el 5 de junio de 2024, de

<https://www.geeksforgeeks.org/cpp-polymorphism/>

Clases abstractas (C++). (2023). *Microsoft*. Recuperado el 7 de junio de 2024, de

<https://learn.microsoft.com/es-es/cpp/cpp/abstract-classes-cpp?view=msvc-170>

Inheritance in C++. (2024). *Geeks for Geeks*. Recuperado el 5 de junio de 2024, de

<https://www.geeksforgeeks.org/inheritance-in-c/>

Operators Overloading in C++. (2023). *Geeks for Geeks*. Recuperado el 5 de junio de 2024,

de <https://www.geeksforgeeks.org/operator-overloading-cpp/>

Pointers vs References in C++. (2023). *Geeks for Geeks*. Recuperado el 5 de junio de 2024,

de <https://www.geeksforgeeks.org/pointers-vs-references-cpp/>

References in C++. (2024). *Geeks for Geeks*. Recuperado el 5 de junio de 2024, de

<https://www.geeksforgeeks.org/references-in-cpp/>

Sobrecarga de operadores (solo C++). (2024). *IBM*. Recuperado el 6 de junio de 2024, de

<https://www.ibm.com/docs/es/i/7.5?topic=only-overloading-operators-c>