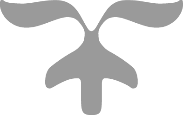


Instituto Tecnológico De Costa Rica



Curso: Lenguajes de programación

**Proyecto 1**

**Gestión de biblioteca**

Profesor:

Allan Rodríguez Dávila

Estudiantes:

Adjany Gard Alpizar Carné: 2021117164

Kevin Salazar Valles Carné: 2021138547

Segundo semestre, 2023

Fecha de entrega: 05 Septiembre

Tabla de contenidos

[**Pruebas de funcionalidad 2**](#_heading=h.gjdgxs)

[Requerimientos: 3](#_heading=h.1ivlfmf12ef4)

[**Manual de usuario 4**](#_heading=h.wm7kcypenal1)

[**Descripción del problema 9**](#_heading=h.md10yheicgxm)

[**Diseño del programa 10**](#_heading=h.1vpsvtgw3df6)

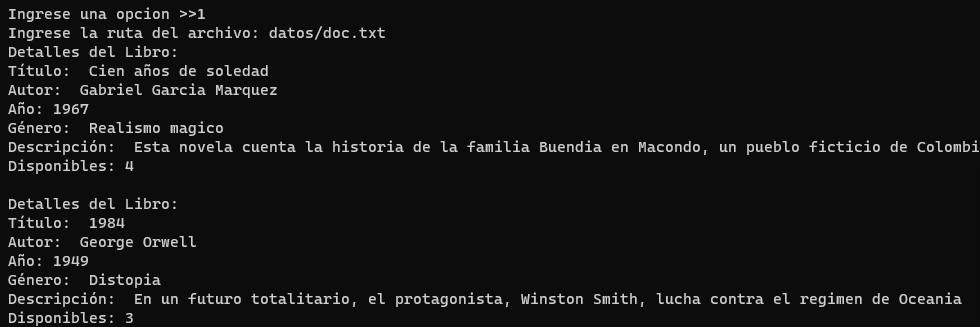
[**Librerías usadas 10**](#_heading=h.wcyg2rwhope7)

[**Análisis de resultados: 11**](#_heading=h.kufr6zg0jlvs)

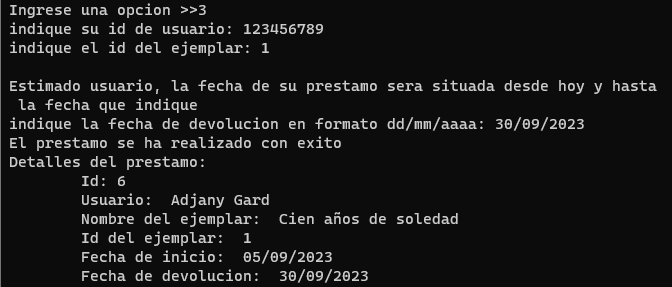
[**Bitácora. 12**](#_heading=h.d5ec2cu3zxcy)

# Pruebas de funcionalidad

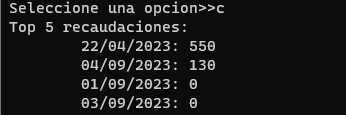
Agregado de libros desde txt



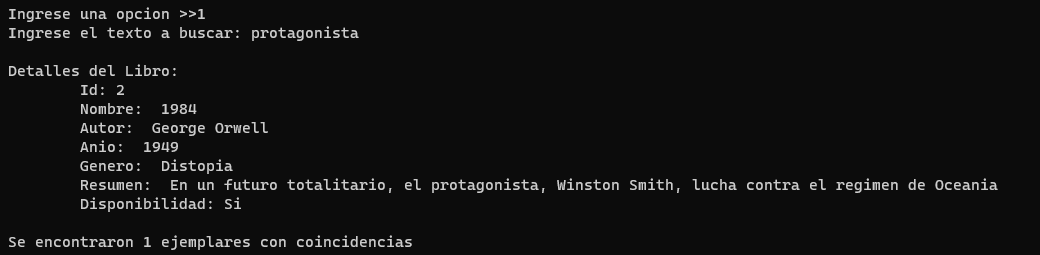
Prestar un libro

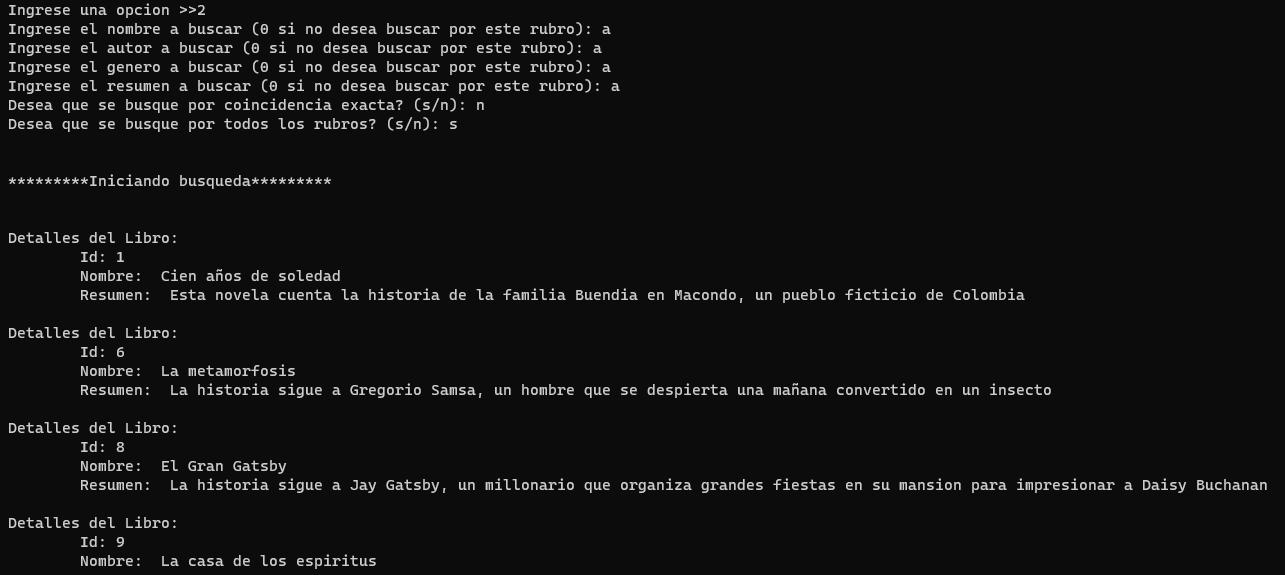


Muestras de estadísticas



Búsquedas





## Requerimientos:

Para utilizar este programa es necesario un entorno Linux. Ya que si se utiliza otro entorno se generarán errores porque la librería para el manejo de JSON solo sirve en este entorno. Es necesario un compilador adecuado para el programa, si se está utilizando el entorno Linux no es necesario utilizar compiladores externos. Para utilizar el programa de forma correcta es necesario tener un archivo txt que contenga los nombres de los ejemplares disponibles además de los archivos JSON necesarios para la persistencia de la información. Además es necesario los siguientes archivos encabezados: structs.h, opcionesOperativas.h, opcionesAdministrativas.h, auxiliares.h y manejoJson.h

# 

# Manual de usuario

Para ejecutar el programa se hará uso de un ambiente operativo basado en Unix (cualquier versión de Linux).

Antes de ejecutar el programa, se debe tomar en cuenta que se debe instalar una librería especializada para ejecutar correctamente el programa, esto para el manejo de archivos Json, esto con el siguiente comando

>> sudo apt intstall libjson-c-dev

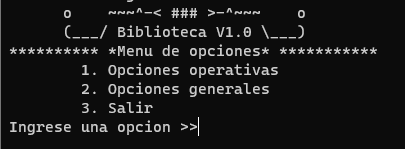
De forma que para agregarlo al archivo en el cual se necesita solo sería importar de esta forma:

#include <json-c/json.h>

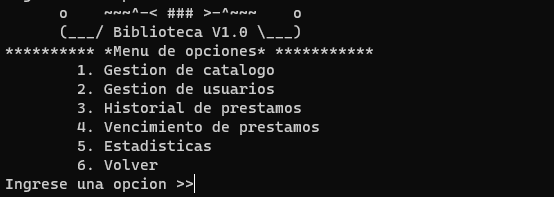
Ahora se procede con la ejecución del programa, para lo cual se usa gcc, usando el siguiente comando se puede compilar y ejecutar el archivo que posee el main:

>> gcc programa.c -o programa -ljson-c && ./programa

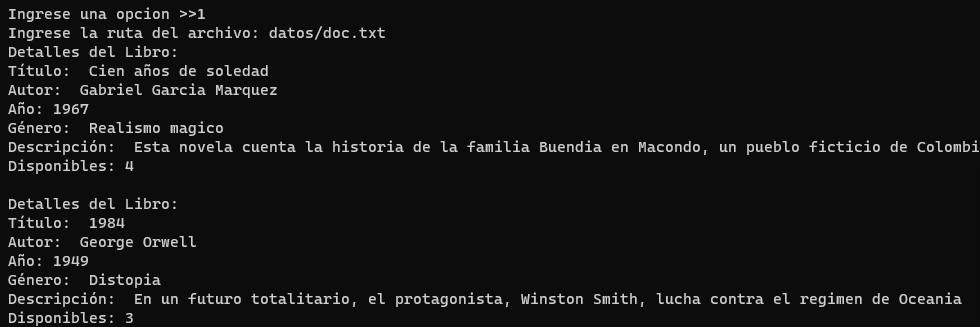
Esto nos llevará directo al menú:



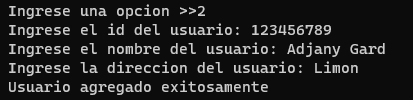
A continuación se presentan 3 funciones, de las cuales las primeras dos abren paso a submenús de opciones específicas, seguidamente veremos cada una de las opciones, empezando por las opciones operativas:



La primera opción permite abrir un menú con 6 opciones, cada una con una función diferente. La primera opción nos permite cargar libros desde un archivo .txt al programa, archivo el cual se puede especificar su ruta:

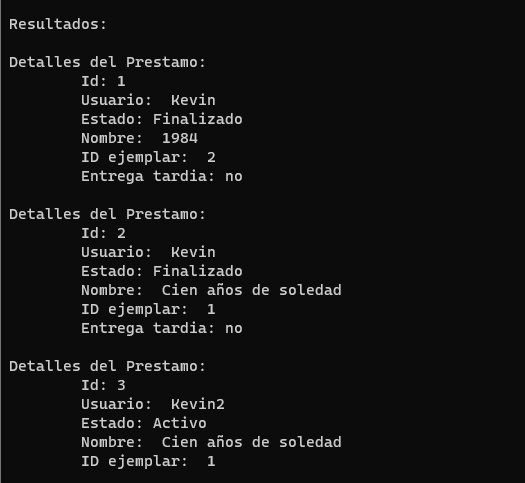


La opción 2 permite agregar un usuario al programa, pidiendo sus respectivos datos:

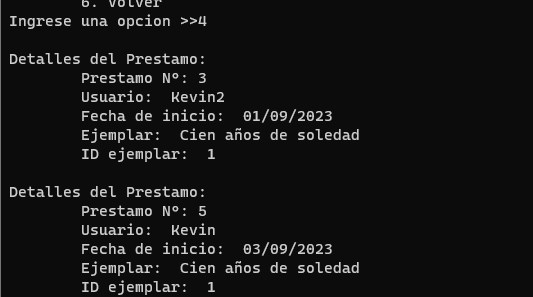


La opción 3 muestra el historial de los préstamos en un rango de fechas:

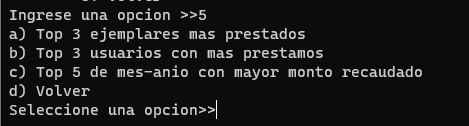




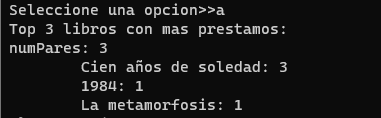
La función 4 muestra los préstamos que se encuentran próximos a vencer (en los próximos 3 días) o aquellos que se encuentran vencidos:



La opción 5 abre paso a otro submenú donde se muestran 3 opciones propias a las estadísticas de los datos del programa. Las mismas serán mostradas a continuación:



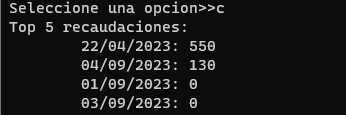
La primera opción muestra los 3 ejemplares que más se han prestado



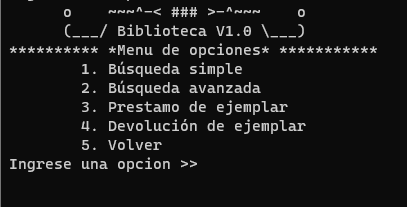
La segunda opción muestra los 3 usuarios que más préstamos han hecho:

# 

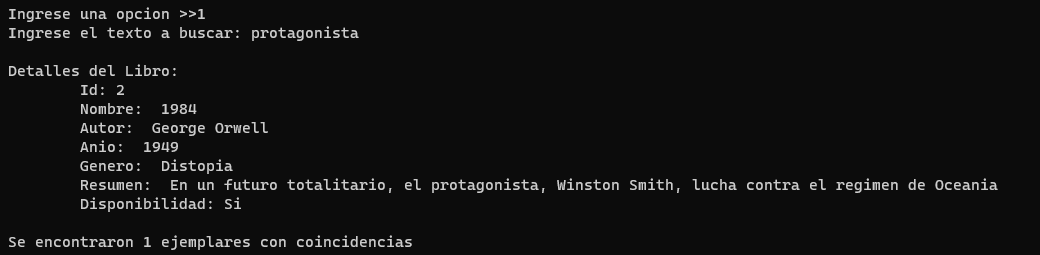
La última opción muestra un top 5 de los meses en los cuales más recaudaciones ha habido:



Seguidamente quedaría por visualizar las opciones generales

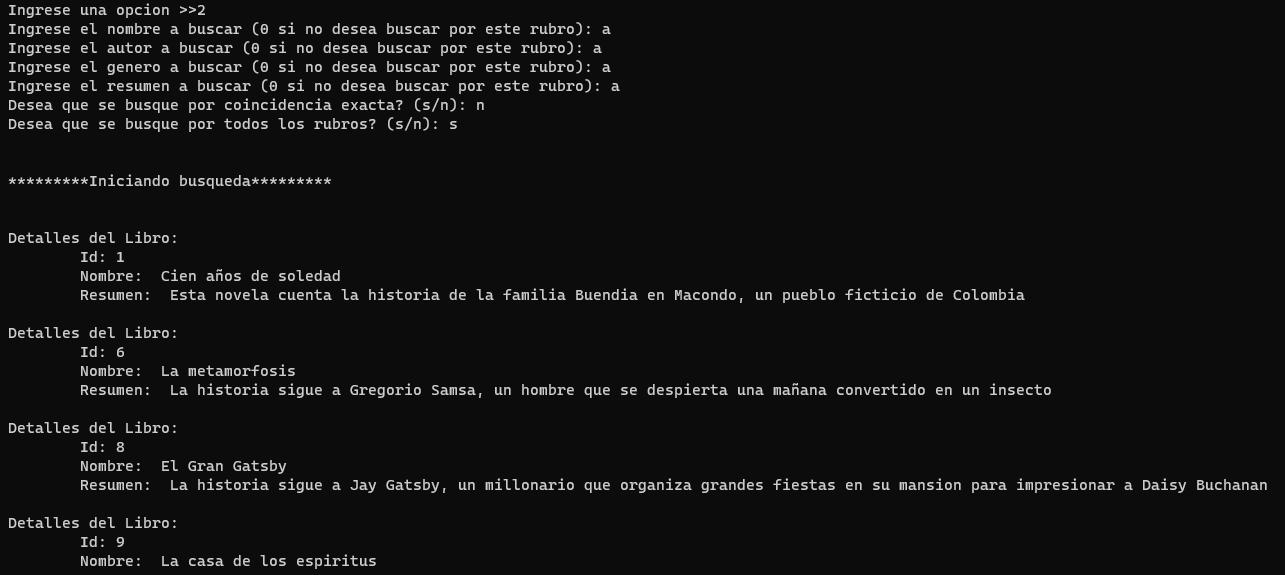


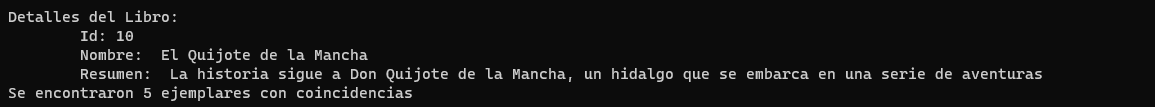
La opción 1 hace una búsqueda simple entre los ejemplares, de forma que se le pasa una palabra y busca aquellos ejemplares que coinciden en algún rubro con esa palabra:



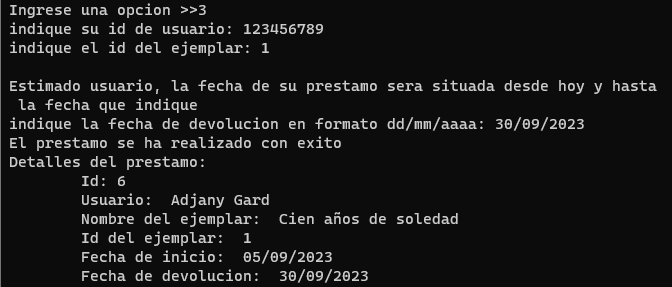
Nota: si bien la función cumple con su cometido, su búsqueda no es 100% precisa por lo que puede presentar fallas…

La segunda función también es una búsqueda pero de forma avanzada, eso quiere decir que permite buscar por coincidencia exacta o simple, por uno o todos los rubros indicados:

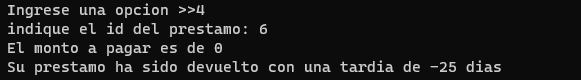




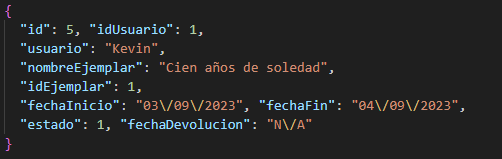
La tercera función permite al usuario hacer el préstamo de un determinado ejemplar:

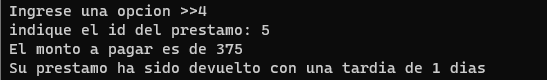


La cuarta opción permite devolver un préstamo hecho previamente:



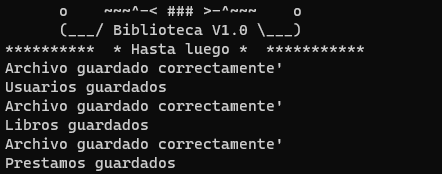
Como en este ejemplo se ha devuelto el libro el mismo día del préstamo, intentemos con otro (agregado manualmente en los archivos para fines de uso):





El cálculo de pago se hace automáticamente según las reglas dadas para el sistema.

La función que queda por ver es salir, la cual permite cerrar el programa. Una vez que se ejecuta esa función se produce la persistencia hacia los archivos .json



# Descripción del problema

Se deberá crear un Sistema de gestión de biblioteca en C que permita gestionar las principales funciones operativas y de usuario. El programa se utilizará a través de la consola.

La información se almacenará y consultará desde archivos json. El programa deberá desplegar un menú al usuario con dos submenús o grupo de funciones. El menú debe ser 100% navegable por lo que el usuario siempre debe de tener la opción de volver.

A nivel de opciones operativas se deben manejar las siguientes acciones:

Gestión de catálogos: Esta funcionalidad carga el catálogo de ejemplares disponibles al programa.

Gestión de usuarios: El sistema debe permitir la creación de nuevos usuarios.

Historial de préstamos: El sistema deberá mostrar todos los préstamos generados en el sistema en un rango de fechas indicado por el usuario.

Vencimiento de préstamos: El sistema deberá mostrar los préstamos vencidos y próximos a vencer.

Estadísticas: Se deberán mostrar estadísticas como el top 3 de ejemplares más prestados, usuarios con más préstamos y los meses-año con mayor recaudación.

Pare el apartado de las opciones administrativas son necesarias las siguientes funcionalidades:

Búsqueda simple: El sistema deberá solicitar al usuario un texto y con lo escrito por el usuario buscará aplicando la técnica “contiene” y “o”.

Búsqueda avanzada: El sistema le permitirá al usuario buscar por: nombre, autor, género y/o resumen, con la técnica “contiene” o “exacta”,

Préstamo de ejemplar: El sistema debe implementar las funciones necesarias para que un usuario pueda pedir un préstamo.

Devolución de ejemplar: El sistema debe implementar las funciones necesarias para que se pueda hacer la devolución de un ejemplar prestado.

Además, es necesario que la información del programa sea persistente mediante el uso de la librería json-c.

# 

# Diseño del programa

# Librerías usadas

json-c/json.h: Librería utilizada para el manejo de archivos JSON.

stdio.h: Libreria utilizada para el ingreso y el despliegue de la información desde el teclado.

stdlib.h: La librería estándar de c.

string.h: Librería utilizada para el manejo de cadenas de caracteres.

time.h: Librería para el manejo del tiempo

stdbool.h: Librería utilizada para implementar booleanos.

# 

# 

# 

# 

# 

# 

# 

# Análisis de resultados:

|  |  |  |  |
| --- | --- | --- | --- |
| Funcionalidad | Requerimiento | Logrado | No logrado |
| Opciones operativas | El sistema debe permitir dar mantenimiento a la información del catálogo, se debe permitir incluir en lote (incluye con información correcta y únicamente nuevos); | X |  |
| Almacenar los datos en un archivo JSON | X |  |
| Implementación del menú necesario para la navegabilidad entre opciones | X |  |
| El sistema debe permitir crear usuarios | x |  |
| Mostrar los préstamos en un rango de fecha dado por el usuario | x |  |
| Mostrar los préstamos que están próximos a vencer | x |  |
| Cargar todos los ejemplares de un archivo txt | x |  |
| Mostrar el top 3 de producciones más prestadas. | x |  |
| Mostrar top 3 usuarios con más préstamos | x |  |
| Mostrar Top 5 de mes-año con mayor monto recaudado. | x |  |
| Utilizar el manejo de fechas | x |  |
| Leer datos de un archivo json | x |  |
| Opciones generales | Implementación de la búsqueda simple. | x |  |
|  | Implementación de la búsqueda avanzada | x |  |
|  | El sistema permite el préstamo de un ejemplar | x |  |
|  | El sistema permite la devolución de un ejemplar | x |  |
|  | Se le muestra el usuario los montos a cancelar | x |  |

# Bitácora.

Link: <https://github.com/Kevin-Salazar-itcr/Proyecto01-Lenguajes>