

## **Proyecto Programado #1**

### ***Gestión de biblioteca***

### **Introducción**

Una biblioteca es el lugar donde se guardan libros, sin embargo, en la actualidad esta concepción se ha visto superada para pasar a referirse tanto a las colecciones bibliográficas y audiovisuales, como a las instituciones que las crean y las ponen en servicio para satisfacer las necesidades de los usuarios.

Las bibliotecas pretenden responder a la amplia gama de necesidades que pueden demandar sus usuarios. Además de obras literarias clásicas, sus fondos pueden estar integrados por textos que proporcionan información sobre servicios sociales, obras de referencia, discos, películas, libros recreativos, entre otros.

El sistema para gestionar una biblioteca puede incluir funciones operativas para la administración del catálogo, así como, funciones para los usuarios como por ejemplo búsqueda y préstamo de libros.

### **Proyecto a desarrollar**

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, no se implementará interfaz gráfica. La información se almacenará y consultará desde archivos json (json-c).

El programa deberá desplegar un menú al usuario con dos submenús o grupo de funciones (el menú debe mostrarse recurrentemente después de cada opción realizada -a excepción del salir- y tener la opción de subir al menú principal desde los menús secundarios) y salir.

### **Opciones Operativas**

La solución debe disponer de una sección de opciones operativas en donde se accederá a todas estas funcionalidades.

Se habilitará un menú donde se colocarán las siguientes funcionalidades:

- Gestión de catálogo
- Gestión de usuarios
- Historial de préstamos
- Vencimiento de préstamos
- Estadísticas
- Volver

### **Gestión de catálogo**

El sistema debe permitir dar mantenimiento a la información del catálogo, se debe permitir incluir en lote (incluye con información correcta y únicamente nuevos); la información será el nombre, autor, año publicación, género, resumen y cantidad. La unicidad la dará el nombre. Se debe indicar una ruta de archivo que contendrá los registros por incluir, con la siguiente forma:

Cien años de soledad|Gabriel García Márquez|1967|Realismo mágico|Esta novela cuenta la historia|4  
1984|George Orwell|1949|Distopía|En un futuro totalitario, el protagonista lucha contra el régimen|3  
Orgullo y prejuicio|Jane Austen|1813|Novela romántica|La historia sigue Elizabeth Bennet mientras|2

Según la cantidad indicada se genera un identificador para cada ejemplar (se almacenan ejemplares de producciones bibliográficas). Los datos se almacenan en otro archivo tipo json.

### **Gestión de usuarios**

El sistema debe permitir crear usuarios, donde se indicará el número de identificación, el nombre y la dirección. La unicidad la dará el número de identificación.

### **Historial de préstamos**

El sistema deberá mostrar todos los préstamos generados en el sistema en un rango de fechas indicado por el usuario (inicio-fin) -considerando la fecha de entrega-, la información a mostrar será: identificador de préstamo, usuario, estado (activo o finalizado), nombre, identificador de ejemplar y si presentó entrega tardía (si o no) -aquellos con estado finalizado-.

### **Vencimiento de préstamos**

El sistema deberá mostrar los préstamos vencidos y próximos a vencer (préstamos que vencen de 0 a 3 días), la información a mostrar por cada préstamo será: identificador de préstamo, usuario, fecha de entrega, nombre e identificador de ejemplar. Considera fecha de sistema.

## **Estadísticas**

Se deberán mostrar las siguientes estadísticas (indican el código o letra de esta):

- A. Top 3 de producciones (nombre) más prestadas.
- B. Top 3 de usuarios con más préstamos.
- C. Top 5 de mes-año con mayor monto recaudado (según fecha de inicio).

## **Opciones Generales**

Para acceder a estas funcionalidades el usuario deberá ingresar por medio del Menú Principal y se deben habilitar las siguientes funcionalidades (Menú General):

- Búsqueda simple
- Búsqueda avanzada
- Préstamo de ejemplar
- Devolución de ejemplar
- Volver

### **Búsqueda simple**

El sistema deberá solicitar al usuario un texto y con lo escrito por el usuario buscará aplicando la técnica “contiene” y “o”, es decir, buscará si el texto indicado está contenido en el nombre, autor o resumen. Por cada ejemplar que haga “match” se mostrará identificador, nombre, resumen y estado (disponible o no).

### **Búsqueda avanzada**

El sistema le permitirá al usuario buscar por: nombre, autor, género y/o resumen, con la técnica “contiene” o “exacta”, Se le preguntará al usuario por cada uno de los campos si desea buscar un texto (vacío indica no) y además si desea aplicar la técnica “contiene” o “exacta”, y si se aplicará el operador “o” o “y” entre los campos. Por cada ejemplar que haga “match” se mostrará identificador, nombre, resumen.

### **Préstamo de ejemplar**

El sistema debe permitir incluir un préstamo de ejemplar, el usuario deberá indicar usuario, el identificador de ejemplar, una fecha de inicio y final. El sistema verificará que para las fechas indicadas el ejemplar esté disponible, si existe y está disponible, se le generará un comprobante indicando: identificador de préstamo, usuario, fecha inicio, fecha de entrega, nombre e identificador de ejemplar.

### **Devolución de ejemplar**

El sistema debe permitir devolver un ejemplar, para esto solicitará el identificador de préstamo y la fecha de devolución. El sistema calculará el monto asociado al préstamo de la siguiente manera: Préstamo de 1 a 7 días:

| Duración préstamo | Tarifa día préstamo | Tarifa día tardía |
|-------------------|---------------------|-------------------|
| 1 a 7 días        | 150                 | 75                |
| 8 a 15 días       | 125                 | 50                |
| 16 días o más     | 100                 | 25                |

Se le muestra al usuario un comprobante indicando el monto a cancelar.

### Salir

Los datos del sistema son persistentes, toda la información indicada anteriormente debe ser almacenada en archivos tipo json.

### Otros aspectos

- a. Se deberá respetar los principios de unicidad, seguridad e integridad.
- b. El programa deberá hacer uso de struct.

### Puntos Extra

Se darán **2.5 puntos adicionales** al entregar a más tardar el miércoles 23 de agosto a las 11:55:55 PM el Documento de Requerimientos, ver plantilla suministrada en el Tec Digital. Debe subirse en la documentación llamada “Proyecto Programado I (archivos adicionales)” debajo de la carpeta de “Proyectos”.

Se darán **5 puntos adicionales** si el comprobante de devolución se genera en pdf.

### Aspectos técnicos

El proyecto deberá estar escrito en el lenguaje de programación C (no C++), y deberá de funcionar en el sistema operativo elegido por el estudiante, se recomienda Linux. En caso de requerir librerías adicionales para compilar y ejecutar el programa, deberán especificarlo en la documentación, ya que de lo contrario se descontarán puntos en la evaluación.

Deberán utilizar el sistema de control de versiones GitHub, el repositorio deberá ser público o incluir al profesor en el control de acceso de este.

Se valorará el aporte generado por cada estudiante, considerando entre otras cosas, los commits generados por cada uno. Por lo que el puntaje obtenido por cada uno de los estudiantes puede ser diferenciado.

## Documentación

La documentación es un aspecto de gran importancia en el desarrollo de programas, especialmente en tareas relacionadas con el mantenimiento de estos.

Para la documentación interna, deberán incluir comentarios descriptivos para cada función, con sus entradas, salidas, restricciones y **objetivo**.

La documentación externa deberá incluir:

1. Portada.
2. Manual de usuario: **instrucciones de compilación, ejecución y uso**.
3. Pruebas de funcionalidad: incluir *screenshots*.
4. Descripción del problema.
5. Diseño del programa: decisiones de diseño, algoritmos usados, diagrama de archivos.
6. Librerías usadas: manejo de archivos json-c, etc.
7. Análisis de resultados (lista de cotejo): objetivos alcanzados, objetivos no alcanzados, y razones por las cuales no se alcanzaron los objetivos (en caso de haberlos).
8. Bitácora (autogenerada en git, commit por usuario incluyendo comentario).

## Forma de trabajo

El trabajo se debe realizar en parejas.

## Evaluación

La evaluación se va a centrar en dos elementos: programación y documentación.

El proyecto programado tiene un valor de **10%** de la nota final, en el rubro de Proyectos.

Desglose de la evaluación del proyecto programado:

1. Documentación interna 2 pts.
2. Documentación externa 8 pts.
3. Funcionalidad 80 pts (ver detalle en Proyecto a Desarrollar)
4. Revisión del proyecto (según completitud del proyecto y gestión del tiempo) 5 pts.
5. Hora de Entrega 5 pts.

## Aspectos administrativos

Debe crear un archivo **.zip** ("PP1\_Est1\_Est2.zip") que contenga únicamente un archivo **info.txt** y 2 carpetas llamadas **documentacion** y **programa**, en la primera deberá incluir el documento de *word* o pdf solicitado y en la segunda los archivos y carpetas necesarias para la implementación de este proyecto programado, y/o link en git del repositorio. El archivo **info.txt** debe contener la siguiente información (cualidades):

- a. Nombre del curso
- b. Número de semestre y año lectivo
- c. Nombre de los Estudiantes
- d. Número de carnet de los estudiantes
- e. Número de proyecto programado
- f. Fecha de entrega
- g. Estatus de la entrega (debe ser **CONGRUENTE** con la solución entregada):  
[Deplorable | Regular | Buena | MuyBuena | Excelente | Superior]

## Entrega

Deberá subir el archivo antes mencionado al TEC Digital en el curso de LENGUAJES DE PROGRAMACIÓN GR 60, en la asignación llamada "P1" debajo del rubro de "Proyectos". En la evaluación del Proyecto el rubro de "Hora de Entrega" valdrá por 5 puntos de la nota total del proyecto, según la siguiente escala:

- a. Si se entrega antes de las 11:55:55 **PM** del lunes 04 de septiembre de 2023, 5 puntos.
- b. Si se entrega antes de las 11:55:55 **AM** del martes 05 de septiembre de 2023, 2.5 puntos.
- c. Si se entrega antes de las 11:55:55 **PM** del martes 05 de septiembre de 2023, 0 puntos.

**NO SE ACEPTARÁN** trabajos que contengan "commits" posterior a esta fecha.

Todos los miembros del grupo deberán participar de la revisión, ya que de lo contrario no se les asignará el puntaje correspondiente.