

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

PROGRAMACIÓN 2
1ra práctica (tipo b)
Primer Semestre 2024

Indicaciones Generales:

- Duración: 110 minutos.

NO SE PERMITE EL USO DE APUNTES DE CLASE, FOTOCOPIAS NI MATERIAL IMPRESO

- No se pueden emplear **variables globales, NI OBJETOS** (con excepción de los elementos de `iostream`, `omanip` y `fstream`). **NO PUEDE UTILIZAR LA CLASE `string`**. Tampoco se podrán emplear las funciones `malloc`, `realloc`, `memset`, `strtok` o `strdup`, igualmente no se puede emplear cualquier función contenida en las bibliotecas `stdio.h`, `cstdio` o similares y que puedan estar también definidas en otras bibliotecas. **NO PODRÁ EMPLEAR PLANTILLAS EN ESTE LABORATORIO**
- Deberá modular correctamente el proyecto en archivos independientes. LAS SOLUCIONES DEBERÁN DESARROLLARSE BAJO UN ESTRICTO DISEÑO DESCENDENTE. **Cada función NO debe sobrepasar las 20 líneas de código aproximadamente**. El archivo `main.cpp` solo podrá contener la función `main` de cada proyecto y el código contenido en él solo podrá estar conformado por tareas implementadas como funciones. En el archivo `main.cpp` deberá colocar un comentario en el que coloque claramente su nombre y código, **de no hacerlo se le descontará 0.5 puntos en la nota final**.
- El código comentado NO SE CALIFICARÁ. De igual manera NO SE CALIFICARÁ el código de una función si esta función no es llamada en ninguna parte del proyecto o su llamado está comentado.
- Los programas que presenten errores de sintaxis o de concepto se calificarán en base al 40% de puntaje de la pregunta. Los que no muestren resultados o que estos no sean coherentes en base al 60%.
- Se tomará en cuenta en la calificación el uso de comentarios relevantes.

SE LES RECUERDA QUE, DE ACUERDO AL REGLAMENTO DISCIPLINARIO DE NUESTRA INSTITUCIÓN, CONSTITUYE UNA FALTA GRAVE COPIAR DEL TRABAJO REALIZADO POR OTRA PERSONA O COMETER PLAGIO.

**NO SE HARÁN EXCEPCIONES ANTE CUALQUIER TRASGRESIÓN DE LAS
INDICACIONES DADAS EN LA PRUEBA**

- **Puntaje total:** 20 puntos.

INDICACIONES INICIALES

Cree un proyecto de C++ en NetBeans siguiendo estrictamente las indicaciones que a continuación se detallan:

- La unidad de trabajo será **t:** (Si lo coloca en otra unidad, no se calificará su laboratorio y se le asignará como nota cero)
- Cree allí una carpeta con el nombre **"Lab01_2024_1_CO_PA_PN"** donde **CO** indica: Código del alumno, **PA** indica: Primer Apellido del alumno y **PN** primer nombre (de no colocar este requerimiento se le descontará 3 puntos de la nota final). **Allí colocará los proyectos solicitados en la prueba.**

Cuestionario:

La finalidad principal de este laboratorio es la de reforzar los conceptos contenidos en el capítulo 1 del curso: "Funciones y alcance de variables". En este laboratorio se desarrollará **una biblioteca estática de funciones** en la que se implementen sobrecargas de operadores y funciones que le permitan solucionar el problema planteado. Además, se le solicitará enlazar una biblioteca estática, proporcionada por los profesores del curso para que sea utilizada obligatoriamente en los proyectos solicitados.

En la carpeta solicitada anteriormente, cree **dos carpetas** denominadas **"Parte1_CrearBiblioteca"** y **"Parte2_UsarBiblioteca"**, allí colocará los proyectos solicitados en las partes 1 y 2 respectivamente de este laboratorio. **DE NO COLOCAR ALGUNO DE ESTOS REQUERIMIENTO SE LE DESCONTARÁ 3 PUNTOS DE LA NOTA FINAL. NO SE HARÁN EXCEPCIONES**

PARA LA APERTURA DE LOS ARCHIVOS DE TEXTOS SE LE ESTÁ PROPORCIONANDO UNA BIBLIOTECA ESTÁTICA, OBLIGATORIAMENTE DEBERÁ ENLAZAR Y UTILIZAR ESTA

BIBLIOTECA EN LOS PROYECTOS QUE ASÍ LO REQUIERAN. DE UTILIZAR OTRAS FUNCIONES O INSTRUCCIONES PARA ESTE FIN SE LE DESCONTARÁ 2 PUNTOS EN SU NOTA FINAL.

TAMBIÉN SE LE PROPORCIONARÁ UN ARCHIVO DE CABECERA (.h) CON LAS ESTRUCTURAS DE DATOS QUE UTILIZARÁ PARA DESARROLLAR ESTE LABORATORIO, NO PUEDE MODIFICAR ESTE ARCHIVO, SI MODIFICA DE ALGUNA FORMA ESTE ARCHIVO SE LE DESCONTARÁ 5 PUNTOS EN SU NOTA FINAL.

PARTE 01 (12 puntos): CREACIÓN DE LA BIBLIOTECA ESTÁTICA

Se solicita que desarrolle una biblioteca estática en la cual se defina una serie de operadores sobrecargados que permitirá manejar las estructuras de datos que se le está proporcionando.

Las operaciones que la biblioteca estática permitirá realizar a través de sobrecargas de operadores se definen a continuación:

➤ **Lectura:**

- Sobrecargando el operador >> de modo que permita leer un libro de un archivo de tipo CSV. La operación (**arch >> libro;**) involucrará un archivo separado por comas y una variable de tipo **"struct Libro"**. La sobrecarga deberá devolver un valor de tipo **bool**, **true** si se pudo leer la información y **false** si se llegó al final de archivo. Una línea de archivo tendrá la siguiente forma:

```
QRH6838, Las cronicas de Narnia, C. S. Lewis, 43, 162.37
(código, título, autor, stock y precio)
```

- Sobrecargando el operador >> de modo que permita leer un cliente de un archivo de tipo CSV. La operación (**arch >> cliente;**) involucrará un archivo separado por comas y una variable de tipo **"struct Cliente"**. La sobrecarga deberá devolver un valor de tipo **bool**, **true** si se pudo leer la información y **false** si se llegó al final de archivo. Una línea de archivo tendrá la siguiente forma:

```
45834351, Santisteban Meza Rosa
(DNI y nombre)
El resto de campos deben inicializarse adecuadamente
```

➤ **Operaciones:**

- Sobrecargando el operador >> de modo que permita determinar si se puede o no atender un pedido. La operación **libroPedido >> arregloLibros;** tomará el código del libro pedido de la variable **"libroPedido"** (**struct LibroSolicitado**), buscará el libro en el arreglo **"arregloLibros"** (**struct Libro**) y verificará si se puede atender el pedido. Si puede atenderlo, debe descontar el stock del libro y colocará el precio del libro y el valor **true** en los campos correspondientes de la variable **"libroPedido"**, si no puede atenderlo solo colocará el valor **false** en el campo correspondiente de la variable **"libroPedido"**. La sobrecarga devolverá un valor **true** si se pudo atender el pedido y **false** de lo contrario.

Como no se puede llevar la cuenta de datos en el arreglo, el arreglo deberá tener como último dato un registro con el código con valor "FIN".

El stock NO puede ser negativo.

- Sobrecargando el operador << de modo que permita agregar un pedido a un cliente. La operación **cliente << pedido;** agregará el pedido (**struct LibroSolicitado**) al cliente (**struct Cliente**). La sobrecarga devolverá un valor **true** si se pudo colocar el pedido y **false** si no se pudo porque el arreglo está lleno.
- Sobrecargando el operador ++ de modo se permita calcular el monto que debe pagar un cliente por los libros que se le entregarán. La operación **++cliente** deberá poder calcular y asignar el pago total por los libros que se le entregarán al campo correspondiente del **"cliente"** (**struct Cliente**).

➤ **Impresión:**

- Sobrecargando el operador << de modo que permita imprimir la información de un libro. La operación (**arch << libro;**) permitirá imprimir en un archivo de textos los datos contenidos en una variable de tipo **"struct Libro"**. El formato será el siguiente:

SFZ1256	La Iliada	Homero	34	154.35
---------	-----------	--------	----	--------

Código, título, autor, stock y precio

Los datos deben aparecer con un formato adecuado y alineados correctamente, tanto para los enteros (hacia la derecha), valores de punto flotante (a la derecha y al punto decimal) y cadenas de caracteres (a la izquierda). **No se dará puntaje a esta parte si no respeta esta condición. NO SE HARÁN EXCEPCIONES.**

- Sobrecargando el operador << de modo que permita imprimir la información de un cliente. La operación (`arch << cliente;`) permitirá imprimir en un archivo de textos los datos contenidos en una variable de tipo `"struct Cliente"`. El formato será el siguiente:

```
45834351  Santisteban Meza Rosa
Libros entregados:
  Pedido No.  Codigo  Precio
    00025    SFZ1256  154.35
    ...      ...      ...
Total a pagar: ...
Libros no entregados por falta de stock:
  Pedido No.  Código
    00102    RII8346
    ...      ...
```

Los datos deben aparecer con un formato adecuado y alineados correctamente, tanto para los enteros (hacia la derecha), valores de punto flotante (a la derecha y al punto decimal) y cadenas de caracteres (a la izquierda). **No se dará puntaje a esta parte si no respeta esta condición. NO SE HARÁN EXCEPCIONES.**

Consideraciones:

La solución debe contemplar la elaboración de: 1) un proyecto de implementación y prueba de las sobrecargas, denominado `"Fuentes_Biblioteca_2024_1"`, 2) un proyecto que genere la biblioteca estática (.a) denominado `"Biblioteca_Compilada_2024_1"` y 3) un proyecto donde se pruebe la biblioteca ya compilada `"Prueba_Biblioteca_Compilada_2024_1"`. La prueba de las sobrecargas para el primer y tercer proyecto deben ser hecha lo más simple posible pero que muestre claramente que son correctas. Los tres proyectos deberán colocarse en la carpeta `"Parte1_CrearBiblioteca"`.

NO SE DEBE SOLUCIONAR AQUÍ EL PROBLEMA DE LA PARTE 2, DE HACERLO NO SE LE CALIFICARÁ LA SEGUNDA PARTE DE ESTE LABORATORIO.

EN LAS PRUEBAS PARA LOS PROYECTOS 1 Y 3 SE PUEDE HACER DIRECTAMENTE EN LA FUNCIÓN MAIN, SE PUEDE HACER USO DE "HARD CODE" Y SE PUEDE SOBREPASAR LA CANTIDAD EL NUMERO DE LÍNEAS.

EN LA BIBLIOTECA ESTÁTICA, NO PUEDE DESARROLLAR OTRAS FUNCIONES O SOBRECARGAS ADICIONALES A MENOS QUE ESTÉN DIRECTAMENTE RELACIONADAS A LAS SOBRECARGAS SOLICITADAS.

PARTE 2 (8 puntos): REUTILIZACIÓN DE LA BIBLIOTECA ESTÁTICA.

Desarrolle un proyecto denominado `"UnaAplicacionConBibEstatica"` (dentro de la carpeta `"Parte2_UsarBiblioteca"`) en el cual se utilizarán obligatoriamente las sobrecargas de la biblioteca estática (compilada) `"libBiblioteca_Compilada_2024_1.a"`.

LA PREGUNTA NO SE EVALUARÁ SI EN ESTE PROYECTO NO SE USAN (O NO SE VE QUE USAN) LAS SOBRECARGAS IMPLEMENTADAS EN LA BIBLIOTECA COMPILADA (.A) Y LA BIBLIOTECA Y ESTRUCTURAS PROPORCIONADAS.

El proyecto ejecutará las tareas descritas a continuación utilizando las sobrecargas definidas en la biblioteca:

- a) Leer los datos de los libros contenidos en un archivo CSV como se muestra a continuación y los coloque en un arreglo de estructuras del tipo `"struct Libro"`:

```
ETZ8565,Confesiones de una mascara,Yukio Mishima,27,175.68
LQL0880,Contra el fascismo,Umberto Eco,34,71.63
...
```

Código del libro, nombre, autor, stock, precio

- b) Leer los datos de los clientes contenidos en un archivo CSV como se muestra a continuación y los coloque en un arreglo de estructuras del tipo “**struct Cliente**”:

49516360,Mendoza Egusquiza Rosario
34460612,Morales Valverde Ines Martha
...

DNI del cliente, Nombre del Cliente

- c) Leer los datos de cada pedido contenidos en un archivo de CSV como se muestra a continuación:

000097,42302422, TAB4639 RWW7975 ZOY6470 KTA8754
000098,45181396, KLO8691 UCJ9882
...

Número del pedido, DNI del cliente, lista de libros solicitados en el pedido.

Un pedido puede estar compuesto por varios libros

Con la información de cada línea deberá modificar el stock de libros en el arreglo, así como también registrar los libros pedidos de los clientes. **Cada libro pedido debe registrarse independientemente. DEBE EMPLEAR OBLIGATORIAMENTE LOS OPERADORES SOBRECARGADOS.**

- d) Emitir un reporte con la información completa de los libros y los clientes. El reporte debe tener un título y subtítulos adecuados.

Al finalizar la práctica, comprima la carpeta dada en las indicaciones iniciales empleando el programa Zip que viene por defecto en el Windows, **no se aceptarán los trabajos compactados con otros programas como RAR, WinRAR, 7zip o similares.**

Profesores del curso: Rony Cueva
Erasmus Gómez
Erick Huiza
Heider Sánchez
Miguel Guanira

San Miguel, 5 de abril del 2024.