

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ**  
**FACULTAD DE CIENCIAS E INGENIERÍA**

**LENGUAJES DE PROGRAMACIÓN 1**

**7ma práctica (tipo b)**  
**Segundo Semestre 2021**

**Indicaciones Generales:**

- Duración: 110 minutos.

Obligatoriamente los alumnos deberán mantener en todo momento el AUDIO Y VIDEO de sus computadores abierto de modo que puedan recibir los comunicados que se hagan durante el examen y la revisión de los trabajos que estén desarrollando. De tener algún problema deberán hacérselo saber de inmediato al profesor de su horario por correo. De no hacerlo, no se aceptarán reclamos alegando que no oyeron las indicaciones.

- No se pueden emplear estructuras ni variables globales Tampoco se podrán emplear las funciones malloc, realloc, strdup o strtok, ni se puede emplear cualquier función contenida en las bibliotecas stdio.h, cstdio o similares y que puedan estar también definidas en otras bibliotecas. **NO PUEDE EMPLEAR LA CLASE string.**
- 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 (una línea de código no es una línea de texto, es una instrucción que termina con un punto y coma). 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 cada archivo que implemente en los proyectos (.h y .cpp) deberá colocar un comentario en el que coloque claramente su nombre y código, de no hacerlo se le descontará 0.5 puntos por archivo.
- NO SE CALIFICARÁN aquellas funciones desarrolladas en el mismo archivo que la función main.
- El código comentado NO SE CALIFICARÁ **y esto incluye el comentar el llamado a la función.**
- Los proyectos deben obligatoriamente desarrollarse en NetBeans bajo el sistema operativo Windows. No se revisarán los proyectos desarrollados en otros sistemas operativos o IDEs.
- 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 DESCONTARÁN 2 PUNTOS EN LA NOTA FINAL DE LA EVALUACIÓN SI EN EL PROYECTO ENTREGADO SE ENCONTRASEN BIBLIOTECAS DE FUNCIONES QUE CONTENGAN FUNCIONES QUE NO TIENEN QUE VER CON EL PROBLEMA A RESOLVER**  
 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. ESTO Y EL HECHO DE ENCONTRAR CUALQUIER ARCHIVO YA SEA .cpp O .h CON FECHA U HORA DE CREACIÓN ANTERIOR A LA EVALUACIÓN SERÁ CONSIDERADO UNA FALTA DE PROBIDAD Y POR LO TANTO AMERITARÁ LA ANULACIÓN DE LA PRUEBA.

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

- **Puntaje total:** 20 puntos.

**Cuestionario:**

La finalidad principal de este laboratorio es la de reforzar los conceptos contenidos en los capítulos 6 y 7 del tema: "Programación orientada a objetos" y "Operadores Sobrecargados".

Cree una carpeta denominada **"Libros2021-2Lab07"**. DE NO COLOCAR ESTE REQUERIMIENTO SE LE DESCONTARÁ 2 PUNTOS DE LA NOTA FINAL.

**PARTE01 (12 puntos): CREACIÓN DE LAS CLASES**

Se solicita que desarrolle un proyecto **"LAB07\_PREG01\_CLASES"** dentro de la carpeta correspondiente, DE NO COLOCAR ESTE REQUERIMIENTO SE LE DESCONTARÁ 2 PUNTOS DE LA NOTA FINAL, en la cual se definan clases para el manejo de los libros y prestamos realizados a diferentes usuarios. Así como una serie de operadores sobrecargados que permitan manejar estas clases. A continuación, se definen las clases que serán necesarias:

➤ **Para manejar los usuarios:** La clase se denominará "**Usuario**" y deberá contener lo siguiente: 1) un atributo denominado **carne**, este almacenará el número de carné del usuario (**int**), 2) un atributo denominado **tipo** (**char**) guardará la letra correspondiente al tipo ("**E**", "**A**" o "**D**"), 3) un atributo denominado **nombre** definido por una cadena de caracteres, 4) un atributo denominado **libPrest**, definido por un arreglo de la clase **LibroPrestado**, considerar un máximo de 20 elementos, 5) un atributo denominado **numLibPrest** (**int**) que llevará la cuenta de los libros que tiene el usuario en calidad de préstamo, 6) un campo denominado **condicion** (**int**) que almacenará el estado del usuario: **1** si está **habilitado** o **0** si está **inhabilitado**.

➤ **Para manejar los libros prestados:** La clase se llamará "**LibroPrestado**" y deberá contener lo siguiente: 1) un atributo denominado **carne**, este almacenará el número de carné del usuario (**int**), 2) un atributo denominado **tipo** (**char**) guardará la letra correspondiente al tipo ("**E**", "**A**" o "**D**"), 3) un atributo denominado **codigoLib**, para almacenar el código del libro definido por una cadena de caracteres, 4) un atributo denominado **fechaDeRetiro** (**int**) que almacenará una fecha en que retiró el usuario el libro, en formato AAAAMMDD, 5) un atributo denominado **deuda** (**int**) que almacenará el valor de **1** si el libro superó el plazo de devolución y **0** si el usuario aún tiene plazo para seguir contando con el libro.

➤ **Para manejar los libros de la Biblioteca:** La clase se denominará "**Libro**" y deberá contener lo siguiente: 1) un atributo denominado **codigo** definido por una cadena de caracteres, 2) un atributo denominado **titulo** definido por una cadena de caracteres, 3) un atributo denominado **autor** definido por una cadena de caracteres, 4) un atributo denominado **stock** (**int**) con la cantidad de libros que posee la biblioteca, 5) un atributo denominado **precio** (**double**), con el valor del libro, 6) un atributo denominado **prestados** (**int**) con la cantidad de libros que se han prestado, 7) tres atributos denominados **cantEst**, **cantDoc** y **cantAdm** (**int**) que guardarán la cantidad de usuarios que prestaron el libro según el tipo.

### **"DEBE EMPLEAR OBLIGATORIAMENTE LOS NOMBRES DE LAS CLASES Y SUS ATRIBUTOS"**

Las operaciones que se 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 csv. La operación (**arch>>libro;**) involucrará un archivo y un objeto de la clase "**Libro**". Una línea de archivo tendrá la siguiente forma:

```
KC34197, Las_demoliciones_de_Paris, Zkenia/Chung/Zuniga, 2014, 15, 294.26
(codigo, titulo, autor, año, stock, precio)
```

- Sobrecargando el operador **>>** de modo que permita leer **un** usuario de un archivo csv. La operación (**arch>>usuario;**) involucrará un archivo y un objeto de la clase "**Usuario**". Una línea de archivo tendrá la siguiente forma:

```
17750908E, Hernandez_Barrios_Emilio
(carné:NumeroTipo, nombre)
```

- Sobrecargando el operador **>>** de modo que permita leer **un** libro prestado de un archivo csv. La operación (**arch>>libroprestado;**) involucrará un archivo y un objeto de la clase "**LibroPrestado**". Una línea de archivo tendrá la siguiente forma:

```
45834351A, LM05731, 22/09/2021
(carné:NumeroTipo, codigo de libro, fecha)
```

Estas sobrecargas pueden ser definidas como parte de la clase que manejan.

#### ➤ **Agregación:**

- Sobrecargando el operador **+=** de modo que permita agregar **un** préstamo a **un** usuario. La operación (**usuario += libroPrestado;**) colocará los datos de un préstamo (**LibroPrestado**) al final de su lista de préstamos de un usuario, modificando el número de libros prestados (**numLibPrest**). No debe modificar los atributos que no están involucrados en la operación.

- Sobrecargando el operador += de modo que permita actualizar la información de un libro. La operación (**libro += libroPrestado;**) modificara la cantidad de libros prestados y la cantidad de usuarios correspondiente. No debe modificar los atributos que no están involucrados en la operación.
- **Verificación de deudas en usuarios:** sobrecargando el operador \* de modo que permita verificar si el usuario debe uno o más libros. La operación (**usuario \* fecha**), donde fecha es un int, deberá marcar con **1** todos aquellos libros que el usuario tiene en su poder más días de los permitidos (ver función **"numeroDeDias"** descrita más adelante), con **0** aquellos que no, además colocar en el campo **condicion** el valor de **0** si se le inhabilitó en caso de deuda, si está habilitado debe tener el valor de **1**. Un docente puede tener en su poder un libro solo 15 días, un estudiante 10 días y un administrativo solo 7.

➤ **Impresión:**

- Sobrecargando el operador << de modo que permita imprimir la información de un libro prestado. La operación (**arch<<libroPrestado;**) permitirá imprimir en un archivo de textos los datos contenidos en un objeto de la clase **"LibroPrestado"**. El formato será el siguiente:

AB12345	03/09/2021
---------	------------

El reporte debe estar perfectamente tabulado (**sin usar el carácter '\t'**).

- Sobrecargando el operador << de modo que permita imprimir la información de un usuario. La operación (**arch<< usuario;**) permitirá imprimir en un archivo de textos los datos contenidos en un objeto de la clase **"Usuario"**. **Para el imprimir el detalle de libros prestados deberá utilizar el operador sobrecargado anterior.** El formato será el siguiente:

Carne: 30018447		
Arca/Amezquita/Edric-Ronald		
Estudiante		
INHABILITADO		
AB12345	03/09/2021	
VG765430	28/08/2021	Deuda
...	...	...

El reporte debe estar perfectamente tabulado (**sin usar el carácter '\t'**).

- 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 un objeto de la clase **"Libro"**. El formato será el siguiente:

Codigo	: KC34197
Titulo	: Las_demoliciones_de_Paris
Autor	: Zkenia/Chung/Zuniga
Stock	: 15
=====	
Prestamo a docentes:	2
Prestamo a alumnos:	5
Prestamo a administrativos:	0
-----	
Total de Prestamos:	7
=====	

El reporte debe estar perfectamente tabulado (**sin usar el carácter '\t'**).

**Consideraciones:**

La solución debe contemplar la elaboración del proyecto de implementación, y la prueba de las sobrecargas en el main, no hay problema si para esta labor se excede en el número de líneas en el main. La prueba de las sobrecargas debe ser hecha lo más simple posible pero que se muestre claramente que son correctas.

**PARTE 2 (8 puntos): Prueba final.**

Desarrolle un proyecto denominado **"LAB07\_PREG02\_CARGA"** en el cual se utilizará obligatoriamente las clases desarrolladas en la pregunta anterior. El proyecto ejecutará las tareas descritas a continuación utilizando las sobrecargas definidas anteriormente:

- Leer los datos de los usuarios contenidos en un archivo csv como se muestra a continuación y los coloque en un arreglo fijo de objetos de la clase **Usuario**. Se deben utilizar las sobrecargas desarrolladas en la primera pregunta.

- b) Leer los datos de los libros de la Biblioteca contenidos en un archivo csv como se muestra a continuación y los coloque en un arreglo fijo de la clase **Libro**. Además, se sabe que para que un libro sea válido debe tener un stock mayor de 0. Se deben utilizar las sobrecargas desarrolladas en la primera pregunta.
- c) Leer uno a uno los préstamos que se han realizado y agréguelos al arreglo **libPrest** de la clase **Usuario** y además actualice las cantidades prestadas a cada usuario de la clase **Libro**. Se deben utilizar las sobrecargas desarrolladas en la primera pregunta.
- d) Leer una fecha en formato aaaammdd y determine las deudas de los usuarios contenidos en el arreglo de la clase **Usuario**, utilizando la sobrecarga del primer proyecto. Se deben utilizar las sobrecargas desarrolladas en la primera pregunta.
- e) Emitir un reporte en el que se muestren los datos de todos los usuarios. Se debe utilizar las sobrecargas de impresión de la primera pregunta.
- f) Emitir un reporte en el que se muestren los datos de todos los libros. Se debe utilizar las sobrecargas de impresión de la primera pregunta.

Para realizar las operaciones solicitadas debe desarrollar las funciones necesarias, en una biblioteca exclusiva para estas acciones. **No puede manejar variables enteras en el main para el control del número de libros o usuarios.** Finalmente, no puede agregar más atributos de los indicados en el enunciado, pero si los métodos que sean necesarios.

Al finalizar la práctica, comprima la carpeta de su proyecto 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.** Luego súbalo a la tarea programa en Paideia para este laboratorio.

Profesores del curso: Miguel Guanira  
Rony Cueva

San Miguel, 12 de noviembre del 2021.