

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

LENGUAJES DE PROGRAMACIÓN 1

Examen 2

(Primer Semestre 2018)

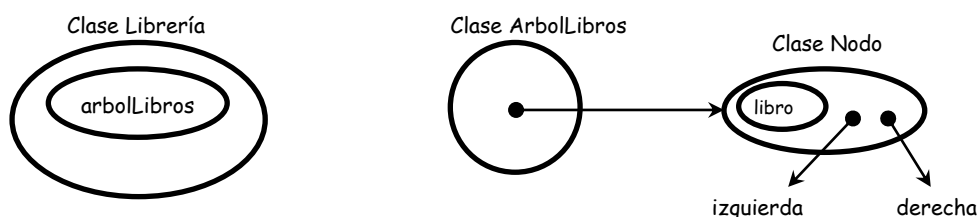
Indicaciones Generales:

- Duración: 3h.
- Se podrá usar como material de consulta solo sus apuntes de clase.
- No se pueden emplear variables globales, ni estructuras. **No se podrá emplear la clase string.** La biblioteca estándar de plantillas (STL) solo se podrá emplear en los puntos donde se indique explícitamente. No se pueden emplear las funciones malloc, realloc, strdup, strtok, tampoco las funciones incluidas en las bibliotecas cstdio, stdio.h o similares. No se podrán definir funciones (ni plantillas) independientes que no estén ligadas como métodos a alguna de las clases planteadas.
- La cláusula **friend** solo se podrá emplear para sobrecargar los operadores de inserción y extracción de flujo (<< y >>) y en el caso de clases auto referenciadas para ligar el nodo con la estructura **inmediata** a ella.
- Deberá respetar estrictamente el **encapsulamiento de datos** en las clases a todo nivel, no se considerará en la nota aquellas funciones que violen el encapsulamiento de los datos.
- Deberá modular correctamente el proyecto en archivos independientes. LAS SOLUCIONES DEBERÁN DESARROLLARSE BAJO UN ERICTO DISEÑO DESCENDENTE. Cada módulo **NO** debe sobrepasar las 20 líneas de código aproximadamente. El archivo **main.cpp** solo podrá contener la función **main** de cada proyecto. 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**.
- Se tomará en cuenta en la calificación el uso de comentarios relevantes.
- Los programas que presenten errores de sintaxis o de concepto se calificarán en base al 40% de puntaje de la pregunta. Los que no den resultados coherentes en base al 60%.
- La **presentación, la ortografía y la gramática de los trabajos influirá en la calificación**.

Puntaje total: 20 puntos

Cuestionario

Se desea elaborar una aplicación orientada a objetos que permita gestionar los libros de una librería mediante un árbol binario de búsqueda (ABB). Para manipular la información se desea cargar los datos de los archivos con los que se trabajará en objetos cuyas estructuras serán similares a la mostrada en la siguiente figura:



En donde se definen las siguientes clases:

"Libro", define el objeto "libro". Esta clase estará compuesta por los atributos: **codigo** (código del libro - char*), **titulo** (título del libro - char*), **autores** (lista de autores del libro - list <Autor>), **precio** (precio del libro - double), **nVend** (número de libros vendidos - int), **monto** (monto recaudado - double).

"Autor", contiene los datos de un autor, tendrá como atributos: **din** (int) y **nombre** (char*).

"ArbolLibros/Clase Nodo", permitirá definir un ABB.

"Librería", esta clase permitirá encapsular el árbol de libros. La clase tendrá un único atributo el cual será un objeto (no un puntero) de clase ArbolLibros.

Para el manejo de la información de la aplicación se cuentan con los siguientes archivos CSV:

Archivo 1
RM6523, Métodos numéricos, 25.30
PQ4355, Lenguaje de programación C++, 85.75
...

Archivo 2
10/05/2108,10:20:30,MG0507
10/05/2018,10:23:10,PQ355
...

Archivo 3
54612389, Disgram Paul, PQ4355, MG0547, ...
66337711, DiConstanzo Scheid, RM6523, ...
...

En el primer archivo se encuentra la lista de libros que vende la librería, aquí aparece el código, título y precio del libro. En el segundo archivo aparece las ventas que se han hecho de los libros en un periodo de tiempo, en cada línea aparece una fecha, la hora y código del libro que se vendió, los libros pueden aparecer varias veces en este archivo. En el tercer archivo se tiene un listado de todos los autores que han escrito algún libro de la librería, en cada línea aparece el DNI del autor, el nombre del autor y los códigos de los libros que han escrito. Los datos no se encuentran ordenados en ninguno de los archivos.

PREGUNTA 1 (1.5 puntos)

Elabore un proyecto denominado "**Pregunta01**", en él defina y pruebe una clase "Autor". La clase deberá contener los atributos definidos en la página anterior y todos los métodos que se requieran para satisfacer las necesidades de la aplicación que se va a desarrollar para la librería.

PREGUNTA 2 (4 puntos)

Elabore un proyecto denominado "**Pregunta02**", copie en él la clase desarrollada en la Pregunta 1 e incorpórela al proyecto, cualquier modificación que haga en este proyecto a la clase incorporada no modificará la calificación de la pregunta 1, NO SE HARÁN EXCEPCIONES.

Defina y pruebe en el proyecto la clase "Libro". La clase definirá los constructores que requiera, su destructor, métodos selectores que se requieran, debe incluir además en entre los selectores uno que permita colocar un autor en su lista de autores de manera ordenada por el nombre del autor, y otro que devuelva los datos del autor dada la ubicación del autor en la lista. También deberá implementar una sobrecarga del operador ++ que permita incrementar en uno el número de libros vendidos y acumular el monto recaudado. Ningún método de esta clase podrá tener como argumento o devolver un resultado que sea de la clase Autor (tampoco puntero a esa clase). Además deberá definir todos los métodos y sobrecargas que se requieran para satisfacer las necesidades de la aplicación que se va a desarrollar para la librería.

PREGUNTA 3 (6 puntos)

Elabore un proyecto denominado "**Pregunta03**", copie en él las clases desarrolladas en las Pregunta 2 e incorpórelas al proyecto, cualquier modificación que haga en este proyecto a las clases incorporadas no modificará la calificación de las preguntas anteriores, NO SE HARÁN EXCEPCIONES.

Defina y pruebe en el proyecto las clases "ArbolLibros" y "Nodo" que permita definir un ABB. Los métodos del árbol permitirán crear el árbol, insertar un libro en el árbol de modo que quede ordenado por el código, también permitirá buscar y modificar los datos de un libro para incrementar los números y montos del libro o para agregar un autor a uno de los libros. Deberá definir también un método que permita eliminar completamente el ABB y otro que permita elaborar un reporte en un archivo de textos de acuerdo al siguiente ejemplo:

RELACIÓN DE LIBROS DE LA LIBRERÍA							
No.	CODIGO	Título	Autores		Precio	Vendidos	Monto
1)	PQ4355	Lenguaje de programación C++	64882343	Castro David	85.75	5	428.75
			54612389	Disgram Paul			
			23661199	Márquez Ana			
			...				
2)	AR7723	Gramática y Ortografía	77345665	Guzmán Paula	120.30	9	1082.70
			10032006	Rojas Naomi			
			...				
			...				

Los datos deben aparecer correctamente tabulados sin el uso del carácter '\t'.

La prueba de las clases deberá, de manera sencilla, mostrar un reporte luego de leer tres libros, agregar dos o tres autores a los libros y registrar 4 o 5 ventas, finalmente eliminar el ABB.

Ningún método de esta clase podrá tener como argumento o devolver un resultado que sea de la clase Autor (tampoco puntero a esa clase).

PREGUNTA 4 (3.5 puntos)

Elabore un proyecto denominado "**Pregunta04**", copie en él las clases desarrolladas en las Preguntas 3 e incorpórelas al proyecto, cualquier modificación que haga en este proyecto a las clases incorporadas no modificará la calificación de las preguntas anteriores, NO SE HARÁN EXCEPCIONES.

Defina y pruebe en el proyecto la clase "Libreria". Esta clase permitirá crear el ABB, leyendo los datos desde el archivo que contiene los datos de los libros, completar los datos de los libros leyendo los autores desde el archivo correspondiente, determinar el número de libros vendidos y el monto recaudado, leyendo la información del otro archivo, y finalmente mostrar todos los datos de los libros en un archivo en el reporte indicado. La clase tendrá un único atributo que será un objeto (no un puntero) de clase ArbolLibros y los métodos convenientes.

La función "main" de este proyecto debe recibir obligatoriamente los nombres de los archivos por sus argumentos (línea de comandos).

PREGUNTA 5 (5 puntos)

Elabore un proyecto denominado "**Pregunta05**", copie en él solo la clase desarrollada en la Pregunta 1 e incorpórela al proyecto, cualquier modificación que haga en este proyecto a la clase incorporada no modificará la calificación de la pregunta 1, NO SE HARÁN EXCEPCIONES.

Elabora una plantilla de clases que permita manejar un ABB de manera genérica. La plantilla deberá poder crear el árbol e imprimir los datos ordenados, correctamente tabulados, etiquetados y con un título que explique la naturaleza de los datos (no podrá emplear el carácter '\t').

Para probar la plantilla deberá crear un árbol con valores de punto flotante (double) y luego otro con autores empleando la clase "Autor" y con los datos tomados del archivo de autores.

CONSIDERACIONES FINALES:

- Cree en el computador una carpeta de trabajo con la siguiente ruta: c:\temp\Examen2. En ella colocará los proyectos que den solución al problema planteado.
- De no respetarse el nombre de los proyectos se descontará 1 punto por cada trasgresión.
- La calificación se otorgará por proyecto desarrollado. Por ninguna razón se asignará puntaje a dos o más preguntas por el mismo proyecto.

NO SE HARÁN EXCEPCIONES

Al finalizar el examen, comprima¹ la carpeta **Examen2** en un archivo con nombre <código del alumno con 8 dígitos>.zip y súbalo a la intranet del curso, en el enlace Documentos, en la carpeta \Examen2\<código del horario>\<aula>. El acceso a la Intranet quedará cerrado automáticamente a las 11:05 am. por lo que el alumno que no suba alguno de los proyectos a la Intranet recibirá como nota CERO en esa pregunta. **NO SE HARÁN EXCEPCIONES.**

Profesores del curso: Miguel Guanira

San Miguel, 5 de julio del 2018.

¹ Para evitar problemas en la corrección de la prueba, utilice el programa de compresión que viene por defecto en el Windows (Zip) **no use 7z.**