

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

LENGUAJES DE PROGRAMACIÓN 1

8va 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 de C que gestionen memoria como `malloc`, `realloc`, `memset`, `strdup`, `strtok` o similares, 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.
- Deberá mantener en todo momento el encapsulamiento de todos los atributos de las clases, así como guardar los estándares en la definición y uso de todas las clases desarrolladas. No se considerará en la nota las clases que violen esto.
- Salvo en la sobrecarga de los operadores `>>` y `<<`, no se podrán definir funciones (ni plantillas de funciones) independientes que no estén ligadas como métodos a alguna de las clases planteadas. Tampoco se podrá emplear la cláusula `protected` ni la cláusula `friend`, de hacerlo se no se le calificarán las clases involucradas.

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 **"CO_PA_PN_Lab08_2024_1"** donde **CO** indica: Código del alumno, **PA** indica: Primer Apellido del alumno y **PN** primer nombre (de no colocar este requerimiento se le descontarán 3 puntos de la nota final). **Allí colocará los proyectos solicitados en la prueba.**

Cuestionario:

Se solicita que desarrolle un proyecto **"LAB08_POLITAD"** dentro de la carpeta correspondiente, **DE NO COLOCAR ESTE REQUERIMIENTO SE LE DESCONTARÁ 2 PUNTOS DE LA NOTA FINAL**, en la cual se declaren las clases descritas con las relaciones necesarias, que permitan manipularlas empleando herencia:

- **Para manejar los libros:** La clase se denominará **"Libro"** y deberá contener lo siguiente: 1) un atributo denominado **nombre** (**char***) definido por una cadena dinámica de caracteres que representa el título del

libro, 2) un atributo denominado **paginas** (**int**) que representa la cantidad de páginas que tiene el libro, 3) un atributo denominado **peso** (**double**) que representa el peso del libro.

➤ **Para manejar los libros tipo enciclopedia:** La clase se denominará "**Enciclopedia**" y deberá contener lo siguiente: 1) un atributo denominado **sku** (**int**) que representa el código de la enciclopedia, 2) un atributo denominado **anho** (**int**) que representa el año que fue publicada la enciclopedia, 3) un atributo denominado **vigencia** (**int**) que representa si el libro aún tiene vigencia, inicialmente en 1. Esta clase posee datos heredados de la clase **Libro**.

➤ **Para manejar los libros tipo novela:** La clase se denominará "**Novela**" y deberá contener lo siguiente: 1) un atributo denominado **autor** (**char***) definido por una cadena dinámica de caracteres que representa el nombre del autor del libro. Esta clase posee datos heredados de la clase **Libro**.

➤ **Para manejar los libros tipo revista:** La clase se denominará "**Revista**" y deberá contener lo siguiente: 1) un atributo denominado **ISSN** (**int**) que representa el código ISSN de la publicación, 2) un atributo denominado **anho** (**int**) que representa el año que fue publicada la revista, 3) un atributo denominado **numero** (**int**) que representa de la revista, 4) un atributo denominado **vigencia** (**int**) que representa si la publicación aún tiene vigencia, inicialmente en 1. Esta clase posee datos heredados de la clase **Libro**.

➤ **Para manejar los nodos de la lista:** La clase se denominará "**Nodo**" y deberá contener lo siguiente: 1) un atributo denominado **plibro**, este es un puntero de la clase **Libro**, 2) un atributo denominado **sig**, este atributo es un puntero a la clase **Nodo** (autoreferenciado).

➤ **Para manejar la lista de libros:** La clase se denominará "**Lista**" y deberá contener lo siguiente: 1) un atributo denominado **peso** (**double**), este atributo representa la suma de los pesos de los libros asignados al estante y almacenados en la lista, este atributo inicialmente está en 0, 2) un atributo denominado **ini**, este atributo es un puntero de clase **Nodo**. Esta lista no tiene un orden determinado. Esta estructura es una lista simplemente enlazada.

➤ **Para manejar los estantes:** La clase se denominará "**Estante**" y deberá contener lo siguiente: 1) un atributo denominado **clase** (**char**), este atributo guardará el tipo de estante que puede ser horizontal y vertical, 2) un atributo denominado **id** (**int**), este atributo guardará el identificador del estante, 3) un atributo denominado **capacidad** (**double**), este atributo almacena el peso máximo que puede soportar el estante, 4) un atributo denominado **Llibros**, este atributo de la clase **Lista**, sirve para guardar todos los libros que se asignan al estante, recuerde que no puede colocar más peso que la **capacidad** del estante en ningún momento.

➤ **Para manejar la biblioteca:** La clase se denominará "**Biblioteca**" y deberá contener lo siguiente: 1) un atributo denominado **AEstantes** dado por un arreglo estático de la clase **Estante**, donde se guardarán los libros. El arreglo puede tener una dimensión de 10 posiciones.

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

Consideraciones:

Para el desarrollo de la pregunta debe considerar el siguiente código:

```
#include "Biblioteca.h"

int main(int argc, char** argv) {
    Biblioteca bli;

    bli.carga();
    bli.llena();
    bli.baja();
    bli.muestra();

    return 0;
}
```

**NO PUEDE
CAMBIAR
ESTE CÓDIGO**

Con las clases indicadas debe realizar las siguientes operaciones:

- (2 puntos) En la clase **Biblioteca** implementar el método **carga**, que se encarga de la lectura del archivo "Estantes2.csv" y cargar la información en el arreglo **AEstantes**.
- (10 puntos) En la clase **Biblioteca** implementar el método **llena**, que se encarga de la lectura del archivo "Libros3.csv" y colocará el libro leído en el estante que aún tiene capacidad para soportarlo. Desde luego el libro leído se asigna en la lista **Llibros** que posee cada estante, actualizando su atributo **peso** con cada libro asignado. Para la lectura de los datos correspondiente de cada libro debe utilizar el método polimórfico **lee**, ya que los mismos pueden ser novelas, enciclopedias o revistas. Para diferenciar cada tipo de libro en el archivo correspondiente las novelas se representan con la letra "N", las enciclopedias con la "E" y las revistas con la "R". **Para esta pregunta puede emplear un puntero a la clase polimórfica como parámetro de los métodos relacionados a su almacenamiento. Si en caso hay libros que no pueden colocarse en los estantes no los considere, desde luego debe liberar la memoria si en caso ya la ha reservado.**
- (4 puntos) En la clase **Biblioteca** implementar el método **baja**, que se encarga de actualizar la vigencia de los libros que tienen como año 2020 o menos, la actualización consiste en cambiar el atributo **vigencia** inicialmente en 1 y colocarlo en 0. Para este paso debe utilizar un método polimórfico **actualiza**. **Este método debe ser puro.**
- (4 puntos) En la clase **Biblioteca** implementar el método **muestra**, que se encarga de realizar la impresión de un archivo debidamente tabulado (**sin usar el carácter '\t'**), que muestre los datos de cada estante y los libros asignados de acuerdo con el tipo su tipo correspondiente. Para este paso debe utilizar el método polimórfico **imprime**. El reporte debe tener el siguiente formato:

```

Estantes:
=====
Id: 1
Tipo: Horizontal
Capacidad: 20
Libros Contenidos:
=====
Titulo:Diamantes y pedernales
Peso: 0.4
Autor:Jose Maria Arguedas

Titulo:ENCICLOPEDIA SALVAT
Peso: 12
SKU:78323      Año: 2015      Vigencia: 0

Titulo:El otonio del patriarca
Peso: 0.5
Autor:Gabriel Garcia Marquez

Titulo:El viejo y el mar
Peso: 0.4
Autor:Hernest Hemingway
...

```

Se recomienda revisar los archivos que servirán para la lectura de datos, los cuales se describen a continuación:

Estantes2.csv
H,1,20
V,2,10
H,3,15
...

Clase, Identificador, Peso o Capacidad que soporta

Libros3.csv
N,Diamantes y pedernales,120,0.4,Jose Maria Arguedas
E,ENCICLOPEDIA SALVAT,2000,12,78323,2015
R,Boletin de Arqueologia PUCP,40,0.25,10292004,2023,3
R,Boletin de Arqueologia PUCP,45,0.25,10292004,2023,2
...

Tipo, Título, Páginas, Peso, Autor/ISSN/SKU, Año, Número .

Recuerde que si no usa polimorfismo la respuesta no será válida, así mismo siempre debe mantener el encapsulamiento debe realizar cada operación donde le corresponde, por ejemplo, si va a leer o imprimir un libro esta operación la debe realizar en la clase que le corresponde. La cláusula *friend* solo se puede emplear para acceder desde la TAD al nodo, en este caso desde la lista a los nodos de la lista.

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
Erick Huiza
Miguel Guanira

Erasmus Gómez
Heider Sánchez

San Miguel, 13 de junio del 2024.