

PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ
FACULTAD DE CIENCIAS E INGENIERÍA

LENGUAJES DE PROGRAMACIÓN 1

8va práctica (tipo b)
Segundo 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**. **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**
- **EL PROYECTO DEBERÁ SER DESARROLLADO BAJO EL PARADIGMA DE PROGRAMACIÓN ORIENTADO A OBJETOS POR LO QUE 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.**
- Deberá mantener en todo momento el encapsulamiento de todos los atributos de las clases, esto implica también que un método NO puede devolver un puntero a algún dato del objeto. También debe guardar los estándares en la definición y uso de todas las clases desarrolladas. No deberá repetir código en los diferentes métodos. No se considerará en la nota las clases que violen esto.
- Cada método NO debe sobrepasar las 20 líneas de código aproximadamente.
- 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 un método si 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.
- **TAMPOCO SE PODRÁ EMPLEAR LA CLÁUSULA **protected** NI LA CLÁUSULA **friend**, SALVO EN LAS CLASES AUTOREFERENCIADAS PARA DECLARAR AMIGA A LA CLASE QUE CONTENGA LA COLA, SE PROHÍBE QUE LOS NODOS DECLAREN **friend** A OTRAS CLASES QUE NO TENGAN QUE VER CON LA COLA COMO, POR EJEMPLO, LOS ATRIBUTOS DEL NODO, DE HACERLO SE NO SE LE CALIFICARÁN LA PREGUNTA.**

SE LES RECUERDA QUE, DE ACUERDO CON EL 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_2"** 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 8 y 9: "Polimorfismo, Punteros a objetos y Clases autoreferenciadas".

PARTE 01 (16 puntos): CREACIÓN DE LAS CLASES

Se solicita que desarrolle un proyecto "PREG01_LAB08" 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 polimorfismo:

- ❖ Para manejar los productos: La clase se denominará "Producto" y deberá contener lo siguiente: 1) un atributo denominado **nombre** (**char***) y 2) un atributo denominado **precio** (**double**).
- ❖ Para manejar a los productos bebibles: La clase se denominará "Bebida" y deberá contener lo siguiente: 1) un atributo denominado **tamano** (**char***), este campo almacena el tamaño de bebida que se puede solicitar (ej., "grande", "mediano"). Además, esta clase posee datos heredados de la clase **Producto**.
- ❖ Para manejar a los productos de entrada: La clase se denominará "Entrada" y deberá contener lo siguiente: 1) un atributo denominado **picante** (**bool**), donde se almacena si la entrada es picante o no. Además, esta clase posee datos heredados de la clase **Producto**.
- ❖ Para manejar a los productos de platos de fondo: La clase se denominará "PlatoFondo" y deberá contener lo siguiente: 1) un atributo denominado **proteina** (**int**), este atributo almacena un número que identifica la proteína empleada (1: Pollo, 2: Carne, 3: Pescado, 4: Lácteos) en el plato. Además, esta clase posee datos heredados de la clase **Producto**.
- ❖ Para manejar un Detalle de la Comanda: la clase se denominará "DetalleComanda" y deberá contener lo siguiente: 1) un atributo **id** (**int**), 2) un atributo denominado **pedido** este es un puntero de la clase **Producto**.
- ❖ Para manejar los nodos de la lista: La clase se denominará "Nodo" y deberá contener lo siguiente: 1) un atributo denominado **dato**, de la clase **DetalleComanda**, 2) un atributo denominado **sig**, este atributo es un puntero a la clase **Nodo** (autoreferenciado). Solo puede usar **friend** para la gestión de los nodos con la cola.
- ❖ Para manejar la Cola de detalle de comandas: La clase se denominará "Cola" y deberá contener lo siguiente: 1) un atributo denominado **longitud** (**int**), este atributo representa la cantidad de nodos que tiene la cola, 2) un atributo denominado **cabeza**, este atributo es un puntero de clase **Nodo** y 3) un atributo denominado **cola**, este atributo es un puntero de clase **Nodo**. La cola debe ser implementada utilizando una lista simplemente enlazada. **Recuerde que al ser una cola no es posible recorrerla**, por tal motivo para su manejo debe implementar los métodos **encola** y **desencola** de forma obligatoria, estos dos métodos recibirán y devolverán un objeto de la clase **DetalleComanda**.
- ❖ Para manejar todos los detalles de las comandas: La clase se denominará "Restaurante" y deberá contener lo siguiente: 1) un atributo denominado **Ccomanda**, este atributo es de la clase **Cola**, aquí se guardarán todos los detalles de las comandas

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

Con las clases indicas debe realizar las siguientes operaciones:

- En la clase **Restaurante** debe implementar el método **carga**, que se encarga de la lectura del archivo "comandas2.csv", con la finalidad de cargar en la cola de **detalle de comandas** la información brindada.

Comandas2.csv						
142	B	Cafe	2	grande		
142	B	Agua Mineral	2.5	pequeno		
142	E	Quesadillas	5.5	sin picante		
142	E	Bruschetta	4	picante		
142	P	Pasta Alfredo	10	4		
142	P	Pollo a la Brasa	15	1		
...						
ID Comanda, tipo de producto, nombre del producto, precio del producto, tamaño/picante/proteína.						

- En la clase **Restaurante** implementar el método **atiende**, que se encargue obtener el **detalle de la comanda** y realizar la impresión de esta en un archivo de prueba debidamente tabulado (**sin usar el carácter '\t'**).

Consideraciones:

Para el desarrollo de ambas preguntas debe considerar el siguiente código, con excepción del método **elimina** que solo debe estar en la pregunta 2:

```
#include "Restaurante.h"

using namespace std;

int main(int argc, char** argv) {
    Restaurante res;

    res.carga();
    res.elimina(3);
    res.atiende();

    return 0;
}
```

**NO PUEDE
CAMBIAR
ESTE CÓDIGO**

Recuerde que si no usa polimorfismo la respuesta no será válida.

PARTE 2 (4 puntos): Proceso final.

Emplee el mismo proyecto de la pregunta anterior. El proyecto ejecutará las tareas descritas a continuación:

- **Cargar** la cola de **detalle de comandas** de acuerdo con lo indicado en la pregunta anterior.
- En la clase **Restaurante** debe implementar el método **elimina**, que se encargará de eliminar de la cola la proteína que recibe como parámetro. Recuerde que debe liberar la memoria comprometida.
- Finalmente **atienda** cada detalle de comandas mostrando un reporte similar al siguiente:

```
Detalle de Comandas:
=====
Codigo de comanda: 142
Nombre: Quesadillas
Precio: 5.50
Sin Picante

Codigo de comanda: 142
Nombre: Bruschetta
Precio: 4.00
Picante

Codigo de comanda: 142
Nombre: Pasta Alfredo
Precio: 10.00
Proteina: Lacteos

Codigo de comanda: 142
Nombre: Pollo a la Brasa
Precio: 15.00
Proteina: Pollo

Codigo de comanda: 161
Nombre: Agua Mineral
Precio: 2.50
Tamano: mediano
...
```

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.**

Profesores del curso: Rony Cueva
Andrés Melgar
Miguel Guanira

Erasmus Gómez
Erick Huiza

San Miguel, 22 de noviembre del 2024.