

PARCIAL SOBRE POO EN C++ - valor: 15% de la nota final de la asignatura.

Nombres completos del estudiante: _____

Instrucciones:
Todas las respuestas deben ser consignadas en este documento.
Puede utilizar **replit** o la **IDE** que el docente le autorice en caso de traer su propio PC.

Información importante: no puede acceder a ningún otro sitio web, o buscar soluciones en Google. No debe utilizar ninguna ayuda escrita ni usar copilot. Lo único que deberá utilizar para dar solución al examen será la IDE autorizada por el docente durante la presentación del parcial y este documento de Word para consignar sus repuestas.
Favor colocar el móvil en modo avión y abstenerse de usarlo mientras presenta este examen.

1. Dominio conceptual (valor 5 % de este examen)

Ejercicio de apareamiento. Arrastré las puntas de las flechas (con clic sostenido) de los recuadros de la izquierda a la definición que corresponde a cada concepto en los recuadros de la derecha (*no se guíe de los colores*).

struct

Clase

Puntero

Destructor

Lista enlazada

Método especial invocado mediante el operador "delete" que permite liberar la memoria ocupada por un objeto

Es una variable que guarda una dirección de memoria

Estructura de datos que permite manipular direcciones de memoria a través de punteros tipo nodo

Es una plantilla en la que se definen los atributos y métodos que tendrán los objetos.

Se conocen como registros. Permiten almacenar la información de los objetos



2. Herencia Múltiple y Archivos Cabecera(.h)
(valor 45 % de este examen)

El ornitorrinco es un animal muy especial que ha tenido pensando a los biólogos, pues implementa las capacidades de otros animales, lo que le permite ser mamífero, colocar huevos, respirar bajo el agua y hasta envenenar.
Lo anterior lo convierte en un candidato ideal para que apliquemos el tema de la herencia múltiple en C++.

En el siguiente código que te estoy facilitando, puedes observar que está todo junto y tu misión será separar las clases generando los respectivos `‘.h’` y `‘.cpp’` para cada clase. Finalmente deberás crear un archivo llamado **main.cpp** en el que se creen ***punteros** de la clase **Ornitorrinco**, que hagan uso de todos los métodos sobrescritos de las clases padres, de modo que la ejecución del programa permita salidas similares a esta:

```
C:\Users\Asus\CLionProjects\orni\cmake-build-debug\orni.exe
Respirando por branquias ✓
He puesto: 2 huevitos ✓
He puesto: 3 huevitos ✓

Process finished with exit code 0
```

***Favor pegar debajo de cada una de las siguientes etiquetas el código correspondiente a cada clase por separado:

Mamifero.h
Mamifero.cpp

Oviparo.h
Oviparo.cpp

Anfibio.h
Anfibio.cpp

Venoso.h
Venoso.cpp

main.cpp

**Screen Shot del programa en ejecución:

3. Polimorfismo y Abstracción (valor 50 % de este examen)

Cree una clase virtual llamada **PiezaDeAjedrez** que contenga una función anónima pura de nombre **movimiento()**; la cual deberá ser sobre escrita por las siguientes clases:

Peón, Torre, Caballo, Alfil, Dama y Rey.

Finalmente, cree un archivo **main.cpp** donde declare un único puntero que se instancie de forma iterativa en cada una de las clases de modo que se muestre un mensaje descriptivo de cada pieza de ajedrez, por ejemplo, si se instanció como caballo que muestre un mensaje como: **“soy el caballo y me muevo en L 3 casillas a la vez, 2 al lado y una arriba o abajo, o viceversa”**. Si está instanciado como alfil **“Soy el alfil y me muevo en diagonal siempre por el mismo color”** y así con cada una de las 6 **“clases”** de piezas que existen en el juego del Ajedrez.

***Favor pegar debajo de cada una de las siguientes etiquetas el código correspondiente a cada clase por separado:

Torre.h
Torre.cpp

Caballo.h
Caballo.cpp

Afil.h

Alfil.cpp

Reina.h

Reina.cpp

Rey.h

Rey.cpp

main.cpp

**Screen Shot del programa en ejecución:

Nota: una vez finalice su examen. Adjunte este archivo de ser posible por la plataforma Interactiva, en caso contrario envíelo a mi correo institucional: anarvae1@eafit.edu.co
Con el asunto: “**LENG_PROG_SOLUCIÓN_PARCIAL_2**”.
Antes de adjuntar el archivo guárdelo con su nombre, ejemplo: [Parcial_2_Miguel_Zapata.docx](#)

¡Ánimo que hemos estudiado y repasado mucho!

"Any fool can write code that a computer understands. Good programmers write code that humans can understand." Martin Fowler