

Práctica General del Segundo Parcial

Mediante el enfoque OO realice el análisis y la programación de los siguientes ejercicios.

1. Cree la clase cronómetro que tenga miembros datos separados de tipo **int** para horas, minutos y segundos. Cree un constructor que inicialice sus atributos a cero, otro que lo inicialice en valores fijos y otro que copie el estado del objeto antes de cualquier ejecución. Entre sus funciones miembro se tiene una que visualiza el cronómetro en formato 11:59:59, otra que avance el cronómetro en un segundo, una que vuelva a colocar el cronometro a cero, una que lo ponga en marcha. Cuando cree su programa principal cree 3 objetos que activen los 3 constructores que crea. No olvide incluir un destructor en el mismo.
2. Realice una clase para generar las siguientes series para N términos:
 - a) 1, 6, 4, 9, 7, 12, 10, 15, 13, 18, 16, 21, 19, 24, 22, ...
 - b) 5, 4, 3, 2, 1, 2, 3, 4, 5, 4, 3, 2, 1, 2, 3, 4, 5, 4, 3, ...
 - c) 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, ...
 - d) 1, 1, 1, 1, 4, 7, 13, 25, 49, 94, 181, 349, 673, ...

Incluya al menos un constructor por defecto.

En el programa principal diseñe, el siguiente menú:

Manejo de Series

1. Leer el tamaño de la serie
2. Generar serie doble
3. Generar serie descende asciende
4. Generar serie de primos
5. Generar serie de cuádruple fibonacci
0. Salida

Digite la opción Elegida → _

3. Cargar elementos en un vector P y determine la ubicación del mayor, del menor elemento, el promedio, la cantidad de elementos pares e impares contenidos en el vector. También implemente el mecanismo necesario para efectuar la ordenación ascendente y la descendente del vector. Cree funciones miembros que permitan realizar estas tareas de manera independiente.
4. Cree la clase pila, para manipular pilas, cree los atributos y funciones miembro que considere necesarias y las funciones básicas además emplee una función amiga que me permita comparar si dos pilas tienen en la cima dos elementos pares, si es así que la

función amiga vacíe los elementos de la primera pila en la otra, si no, que intercambie los elementos que tiene la pila en la cima.

5. Implementar una clase que realice las operaciones principales (Creación, Adición, Eliminar un elemento, Extraer un elemento, Mostrar los elementos) de una cola simple.
6. Implementar una clase que realice las operaciones principales de una cola circular.
7. ¿Cual es el constructor que se llama en las siguientes declaraciones?

```
class prueba_total{  
private:  
    int num;  
public:  
    prueba_total( ) { num = 0 ; }  
    prueba_total ( int n = 2 ) {num = n ; }  
    ~prueba_total();  
};  
  
prueba_total obj1;
```

Practica Grupal**Fecha de Entrega:** 1 de Octubre.

8. ¿Qué es una sobrecarga de funciones?
9. ¿Puedes sobrecargar el constructor? Ejemplificalo.
10. ¿Qué es una sobrecarga de operadores? ¿Cuántos tipos de sobrecarga de operadores existen?
11. Diseña un programa completo que realice la sobrecarga del operador `=`, el cual compare si dos objetos de clase número son iguales.
12. Diseña un programa para administrar una clase matriz, efectúa la sobrecarga de funciones, y del operador `-`, y de un operador binario. Crea un programa principal en el cual muestres como funcionan tus sobrecargas de funciones y operadores.