

Completar y/o realizar TODO AQUELLO QUE ESTÁ ESCRITO EN COLOR ROJO. (flechas señalan lo principal)

RECUPERATORIO INFORMATICA II C++	T o t a l H o j a s	Duración 8:30 a 10:00	29 de Noviembre de 2016
-------------------------------------	-----------------------	---------------------------------	------------------------------------

Nombre y Apellido	N° Legajo	Calificación	Docente Evaluador	
			Nombre	Firma

Una pinturería industrial prepara sus pedidos a través de un sistema informatizado (SI) que está basado en funciones escritas en C++. El SI utilizara dos clases a saber: clase **Color** y clase **Preparacion**.

Combinando los tres Colores primarios se obtienen los demás

- **Clase Color:** Incluye las variables miembro correspondiente a los colores primarios **R, G, B** (rojo, verde, azul), cuyos valores relativos (en términos porcentuales) definen los diferentes colores posibles. Además posee otras variables miembro en donde se almacenará el **costo por litro** de dicha combinación y otra llamada **name** que permitirá conocer el nombre de la mezcla asignado (ej: Turquesa, Violeta, Rojo, etc.)

1) Complete todos los métodos de la clase Color, tanto aquellos que están prototipados como aquellos que se encuentran descriptos por un comentario color rojo en los archivos correspondientes ".h" y ".cpp"

#define red 0 #define green 1 #define blue 2

class Color
{

protected:

int R,G,B; // porcentuales por litro de Preparacion

private:

char* name; // permite acceder al nombre del Color

float costo; // costo por litro de la combinación

public:

*/*Función miembro inline que al ejecutarse, instanciando un objeto de la clase, inicialice a valores por defecto sus variables miembro (R,G,B todas en 33, name en "Blanco" y costo en 1000,00) o través de los argumentos que reciba como parámetros*/*

// Función miembro inline que al ejecutarse, al ser "abandonado" un objeto de la clase, libere recursos del sistema.

void set_name (tipo ..?); // Permite asociar un nombre a una combinación

tipo ..? get_name (void); // Permite acceder al nombre de una combinación

void set_costo (float); // Setea costo por litro de la combinación

float get_costo (void); // Retorna el costo

int get_Color (int); // Retorna el contenido de R, G o B – utilice macros como argumento

Color& operator = (Color&); // Sobrecarga de operador =

*/*sobrecarga del operador de extracción de flujo (>>) que se usa con el objeto cin para ingresar por consola todos los miembros de la clase Color*/*

/ Función void crear_color(...) tal que permita ejecutarse desde el programa principal (main) así : C.crear_color (A, B); ... siendo A , B , C objetos de la clase, donde las variables miembro (R, G, B) de C deberán ser el promedio de aquellas recibidas como argumento, y el nombre name, la concatenación de los nombres */*

};

2) Desarrolle una clase llamada Preparacion (derivada de la clase Color) y sus funciones asociadas en los archivos correspondientes ".h" y ".cpp", cuyos miembros son:

Debe definirlo

float costo; // costo total de la preparación.

int litros; // cantidad de litros a preparar

xxxx cant; // cantidad de objetos tipo "preparación" creados.

// constructor parametrizado con argumentos Color& e int // Color a preparar y litros. DEBE inicializar la CB

// constructor parametrizado con argumentos int, int, int, char*, float, int // DEBE inicializar la CB

// Destructor

void set_costo (float); // Setea el costo por litro, y actualiza el de preparación

float get_litros (); // Retorna los litros de la preparación

Preparación& operator= (Preparación&) // Sobrecarga de operador =

// Realizar la sobrecarga de los operadores "++" y "+" de forma que permita incrementar los litros así:

// ++A; y A + 3 ; //Siendo A un objeto de la clase Preparacion.

*/*sobrecarga del operador de extracción de flujo (>>) que se usa con el objeto cin para ingresar por consola todos los miembros de la clase Color y los litros de la Preparación. El código debe estar optimizado*/*

3) Realice un programa main() haga uso de las clases realizadas.