

C++:

Un Local de despacho de hamburguesas, desea realizar un sistema de gestión para el reparto de mercadería a sus clientes por “delivery”. Los productos que elaboran son: Hamburguesa doble carne (**dc**), simple carne (**sc**) y pollo (**p**). El sistema utiliza las siguientes clases *inconclusas*:

```
class Cliente
{
    char nombre[20];
    char direccion [20];

public:
    Cliente();
    Cliente(char *);
        //nombre
    Cliente(char*,char *);
        //nombre,direccion
    char* getCliente();
    char* getDireccion();
    void set_direccion(char*);
};
```

```
class Costo
{
    xxx float dc; //inicialmente valor 100
    xxx float sc; //inicialmente valor 100
    xxx float p;  //inicialmente valor 100

public:
    Costo();

    void set_costo(float , float , float );
        // dc , sc , p
    float get_dc(void);
    float get_sc(void);
    float get_p(void);
};
```

```
class Envio
{
    Cliente c;
    bool estado;           //inicialmente en false = No entregado
    int cant_dc;
    int cant_sc;
    int cant_p;

    float Total;

    //Agregar estas variables "Recaudacion" y "cant_envio" con sus tipos y
    //modificadores que correspondan

public:
    Envio();
    Envio(Cliente);
    Envio(Cliente,int,int,int); // los 3 enteros del argumento son las
        //cantidades de cada producto

    void set_estado(bool);
    void get_cuenta();         //obtiene la cuenta del pedido y almacena el
        //Total

    //implementar una función que permita imprimir los datos del envío a
    //saber: N° del pedido, nombre del Cliente y dirección, TOTAL y el estado
    //Entregado o a Entregar

    float get_Recaudacion();
    void Entregado();          //cambia el estado a true= Entregado
    //Implementar una función "Recaudado()" de forma que al llamarla sin la
    //necesidad de acceder a ella a través de un objeto de la clase permita
    // saber la recaudación del día
};
```

La Clase cliente cuenta con los datos nombre y dirección para el envío. La clase Costo contiene los precios de cada uno de los productos, y puede ser actualizada en cualquier momento.

La clase Envío deberá heredar la clase Costo (ver punto 2) para elaborar los pedidos. Los miembros de la misma son: Cliente, la cantidad para cada producto pedido, una variable estado (para determinar si fue entregado o a Entregar) y el saldo a abonar (Total). Además deberá implementar variables que permitan obtener la cantidad total de envíos realizada (**cant_envios**) y la recaudación total de los envíos entregados (**recaudacion**)

Se pide:

1) Desarrolle los constructores de la clase Cliente y el constructor parametrizado de la clase Costo. Para esta última, genere la sobrecarga del operador << que permita en cualquier momento imprimir por pantalla el costo unitario de los diferentes productos, mediante la instrucción **cout << c**, en donde **c** es un objeto de la clase **Costo**. Genere un main() en donde se instancie un objeto de tipo Cliente utilizando el constructor parametrizado, y un objeto de la clase Costo con su constructor por defecto, para luego imprimir por pantalla los valores de los costos mediante el objeto **cout**.

2) Modifique la declaración de la clase **Envio**, para que herede de **Costo**, y desarrolle todos los métodos de la misma, de manera que se pueda ejecutar el siguiente código:

```
Cliente A("Pepe", "J.B. Justo 3213");
Cliente B("Cacho", "Yapeyu 197");
Cliente C("Jose", "Medrano 75");

Envio E(A, 2, 1, 1);
cout<<E;

E.set_costo(200, 200, 200);

Envio E1(B, 2, 1, 1);
cout<<E1;

E1.Entregado();
cout<<E1;

//Imprimir recaudación total de todos los envíos
```

3) Agregue a la clase **Envio** las variables **recaudación** y **cant_envios_activos**, con los tipos y modificadores que considere adecuados, de manera que lleven la cuenta de la cantidad de dinero total recaudado y la cantidad de envíos que se encuentren pendientes de entregar.

- a. Modifique los métodos constructores de la clase **Envio** y cualquier otro que considere necesario, de manera que el valor de la variable refleje en todo momento la cantidad de dinero recaudado y la cantidad de pedidos que todavía no se hayan entregado.
- b. Genere una línea en el main que imprima por pantalla el valor de dichas variable.