

Asignación de objetos en C++

Formas de asignación (1)

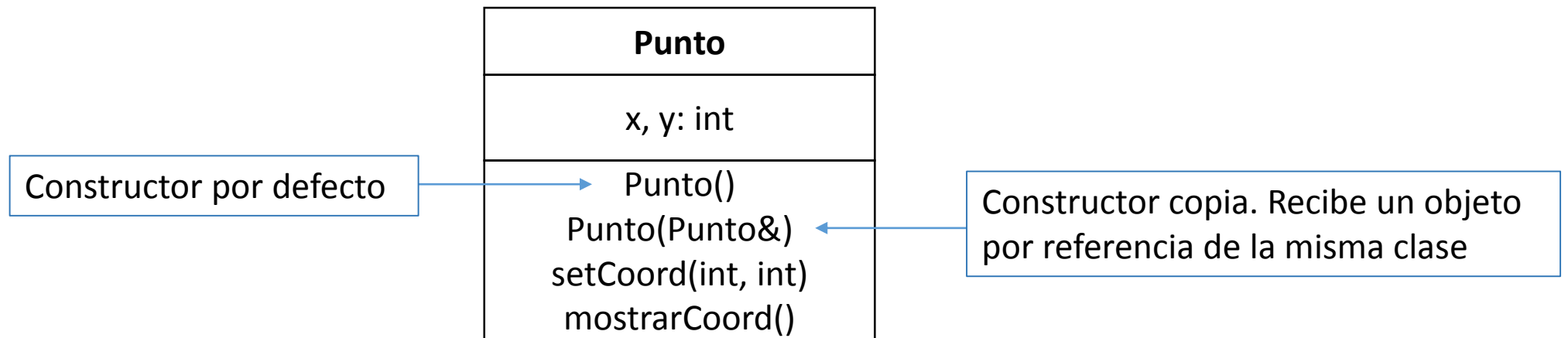
Para asignar un objeto en C++ se utilizan básicamente 4 formas:

1. Declarando un objeto y haciendo uso de comportamiento (métodos *set*) [forma tradicional]
2. Declarar un objeto B como referencia de un objeto A previamente creado
3. Declarar un objeto B como apuntador de un objeto A previamente creado
4. Creando un constructor copia, y enviando un objeto B, para que adquiriera los valores de un objeto A previamente creado

Ejemplo de asignación (1)

Vamos a implementar una clase ***Punto*** que declare dos atributos de tipo int: ***x*** e ***y***, y dos constructores: uno por defecto que asigna cero a ***x*** e ***y***, y otro (Copia) que recibirá como argumento otro objeto de la clase ***Punto***.

Diagrama de Clases



Ejemplo de asignación (2)

Declaración de la Clase (.h)

```
class Punto
{
public:
    int x, y;
    Punto();
    Punto(Punto&);
    virtual ~Punto();

    void setCoord(int, int);
    void mostrarCoord();

protected:

private:
};
```

Constructor
copia

Ejemplo completo comprimido
Asignacion_objetos.zip

Implementación de la interfaz (.cpp)

```
Punto::Punto() {
    x = y = 0;
}

Punto::Punto(Punto& obj) {
    x = obj.x;
    y = obj.y;
}

Punto::~~Punto() {
    //dtor
}

void Prueba::setCoord(int a, int b) {
    x = a;
    y = b;
}

void Prueba::mostrarCoord() {
    cout<<x<<" "<<y<<endl;
}
```

Constructor copia.
Recibe por referencia
un objeto de la clase
Punto