### Herencia en C++ (Parte 4) Ejemplo práctico Tipos de PC

### Enunciado

Implementar una aplicación en C++ que permita mediante una jerarquía de Herencia representar los 4 tipos de referencias de computadores que vende una tienda tecnológica.

### Los computadores son:

- Oficina: tiene características mínimas, como: procesador (CPU), RAM, y DD mecánico. Su tarjeta grafica es integrada. Valor = \$1000
- Programador: Oficina + Disco hibrido, es decir DD + SSD (DD de estado sólido).
  - Valor = \$**Oficina** + 350
- Gamer: Programador + tarjeta grafica independiente (GPU).
  - Valor = \$**Programador** + 650
- **Servidor**: **Programador** + *N* cantidad de procesadores.
  - Valor = \$Programador + (450 \* N cantidad de procesadores)
- Utilizar la composición para describir la CPU de todas las referencias.

- marca: string- velocidad: double
- cantNucleos: int
  - + CPU()
  - + setMarca()
- + setVelocidad()
- + setCantNucleos()
- + getMarca(): string
- + getVelocidad(): double
- + getCantNucleos(): int
- + getInfoCPU(): string

- marca: stringvelocidad: double
- cantNucleos: int
  - + CPU()
  - + setMarca()
- + setVelocidad()
- + setCantNucleos()
- + getMarca(): string
- + getVelocidad(): double
- + getCantNucleos(): int
- + getInfoCPU(): string

### Computador

- + CPU objCPU;
- # valor: long
- marca, referencia: string
  - + Computador()
  - + setMarca()
- + setReferencia(), + setCPU()
  - + getMarca(): string
  - + getReferencia(): string
    - + virtual setValor()
  - + virtual getInfo(): string
    - + getValor(): long

- marca: string- velocidad: double- cantNucleos: int
- + CPU() + setMarca()
- + setVelocidad()
- + setCantNucleos()
- + getMarca(): string
- + getVelocidad(): double
- + getCantNucleos(): int
- + getInfoCPU(): string

### Computador

- + CPU objCPU;
- # valor: long
- marca, referencia: string
  - + Computador()
    - + setMarca()
- + setReferencia(), + setCPU()
  - + getMarca(): string
  - + getReferencia(): string
    - + virtual setValor()
  - + virtual getInfo(): string
    - + getValor(): long

### \_\_\_\_T Oficina

- ram, capacidadDD: int
  - + Oficina()
  - + setRam()
  - + setCapacidadDD()
    - + setValor()
    - + getRam(): int
- + getCapacidadDD(): int
  - + getInfo(): string

- marca: string- velocidad: double- cantNucleos: int
- + CPU() + setMarca() + setVelocidad()
- + setCantNucleos()
- + getMarca(): string
- + getVelocidad(): double
- + getCantNucleos(): int
- + getInfoCPU(): string

### Computador

- + CPU objCPU;
- # valor: long
- marca, referencia: string
  - + Computador()
    - + setMarca()
- + setReferencia(), + setCPU()
  - + getMarca(): string
  - + getReferencia(): string
    - + virtual setValor()
  - + virtual getInfo(): string
    - + getValor(): long

### Oficina

- ram, capacidadDD : int
  - + Oficina()
  - + setRam()
  - + setCapacidadDD()
    - + setValor()
    - + getRam(): int
- + getCapacidadDD(): int
  - + getInfo(): string

### **Programador**

- capacidadSDD: int
- + Programador()
- + setCapacidadSDD()
  - + setValor()
- + getCapacidadSDD(): int
  - + getInfo():string

- marca: string- velocidad: double- cantNucleos: int
  - + CPU()
  - + setMarca()
- + setVelocidad()
- + setCantNucleos()
- + getMarca(): string
- + getVelocidad(): double
- + getCantNucleos(): int
- + getInfoCPU(): string

### Computador

- + CPU objCPU;
- # valor: long
- marca, referencia: string
  - + Computador()
    - + setMarca()
- + setReferencia(), + setCPU()
  - + getMarca(): string
  - + getReferencia(): string
    - + virtual setValor()
  - + virtual getInfo(): string
    - + getValor(): long

# Diagrama de Clases

### Oficina

- ram, capacidadDD : int
  - + Oficina()
  - + setRam()
  - + setCapacidadDD()
    - + setValor()
    - + getRam(): int
- + getCapacidadDD(): int
  - + getInfo(): string

### Programador

- capacidadSDD: int
- + Programador()
- + setCapacidadSDD()
  - + setValor()
- + getCapacidadSDD(): int
  - + getInfo():string

### Gamer

- memoriaGPU: int
  - + Gamer()
- + setMemGPU()
- + setValor()
- + getMemGPU(): int
- + getInfo():string

### Computador **CPU** + CPU obiCPU; - marca: string # valor: long - velocidad: double - marca, referencia: string - cantNucleos: int + Computador() + CPU() + setMarca() + setMarca() + setReferencia(), + setCPU() + setVelocidad() + getMarca(): string + setCantNucleos() + getReferencia(): string + getMarca(): string + virtual setValor() + getVelocidad(): double + virtual getInfo(): string + getCantNucleos(): int + getValor(): long + getInfoCPU(): string Oficina - ram, capacidadDD: int + Oficina() + setRam() + setCapacidadDD() + setValor() + getRam(): int + getCapacidadDD(): int + getInfo(): string Servidor **Programador** - cantCPU: int - capacidadSDD: int + Servidor() + Programador() + setCantCPU() + setCapacidadSDD() + setValor() + setValor() + getCantCPU(): int

+ getInfo(): string

+ getCapacidadSDD(): int

+ getInfo():string

### Diagrama de Clases

Gamer

- memoriaGPU: int

+ Gamer()

+ setMemGPU()

+ setValor()

+ getMemGPU(): int

+ getInfo():string

### Declaración de clases (.h)

```
class CPU
   public:
        CPU();
        virtual ~CPU():
        void setMarca();
        void setVelocidad();
        void setCantNucleos();
        string getMarca();
        double getVelocidad();
        int getCantNucleos();
        string getInfoCPU();
   protected:
   private:
        string marca:
        double velocidad:
        int cantNucleos;
};
class Computador
    public:
        CPU objCPU; //objeto de la composicion
        Computador();
        virtual ~Computador();
        void setMarca();
        void setReferencia();
        void setCPU();
        string getMarca();
        string getReferencia();
        virtual void setValor() = 0;
        virtual string getInfo();
        long getValor();
    protected:
        long valor;
    private:
        string marca, referencia;
};
```

```
class Oficina: public Computador
    public:
        Oficina();
        virtual ~Oficina();
        void setRam():
        void setCapacidadDD();
        void setValor();
        int getRam();
        int getCapacidadDD();
        string getInfo();
    protected:
    private:
        int ram, capacidadDD;
};
class Gamer: public Programador
    public:
        Gamer();
        virtual ~Gamer();
        void setMemGPU();
        void setValor();
        int getMemGPU();
        string getInfo();
    protected:
    private:
        int memoriaGPU:
}:
```

```
class Programador: public Oficina
    public:
        Programador();
        virtual ~Programador();
        void setCapacidadSDD();
        void setValor();
        int getCapacidadSDD();
        string getInfo();
    protected:
    private:
        int capacidadSDD;
};
class Servidor: public Programador
    public:
        Servidor();
        virtual ~Servidor();
        void setCantCPU();
        void setValor();
        int getCantCPU();
        string getInfo();
    protected:
    private:
        int cantCPU;
};
```

Ejemplo completo: TiposPC.zip