



**Instituto Politécnico Nacional**  
**Escuela Superior de Cómputo**

**Departamento de Ciencias e Ingeniería de la Computación**  
**Paradigmas de Programación**

**Profesor:** Miguel Ángel Rodríguez Castillo

**Grupo:** 3CV2

## **Práctica : 5 Herencia**

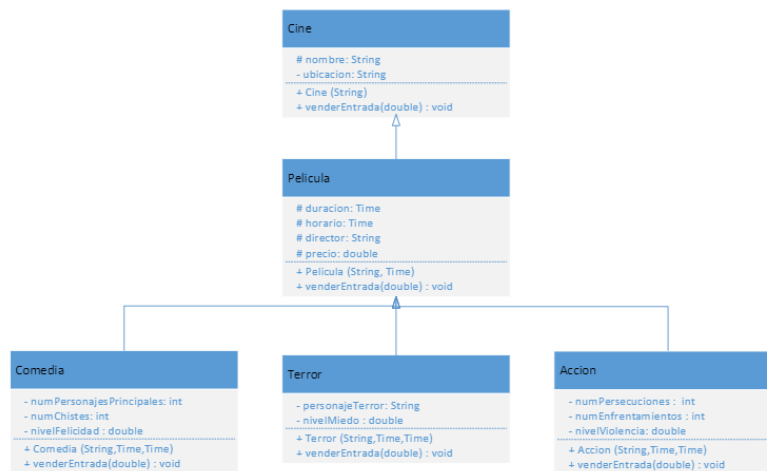
**Equipo:**

- Ramírez Jiménez Itzel Guadalupe
- Colín Ramiro Joel

Fecha de entrega: 17 / 05 / 2021

# Planteamiento del problema

En POO la herencia está en todos lados, de hecho, se podría decir que es casi imposible escribir el más pequeño de los programas sin utilizar herencia. Todas las clases que se crean dentro de la mayoría de los lenguajes de programación orientados a objetos heredan implícitamente de la clase Object y, por ende, se pueden comportar como objetos que es la base de este paradigma. En esta práctica se solicita aplicar los conocimientos adquiridos para el desarrollo de programas en Java en cuanto a este concepto de herencia. Dado el siguiente diagrama UML incompleto (*figura 1*), donde el programa muestre tres películas de diferente género filmográfico, mostrando diferentes tipos de datos como, por ejemplo, la ubicación del cine, el horario de las películas, los precios de entrada, etc. También se solicita el analizarlo mediante este paradigma y completarlo, agregando y/o modificando métodos que se consideren apropiados para su correcto funcionamiento.



*figura 1.*

## Implementación de la solución

### Clase Cine

```
package prac5;
public class Cine {
    protected String nombre;
    private String ubicacion;

    public Cine(String nombre, String ubicacion) {
        this.nombre = nombre;
        this.ubicacion = ubicacion;
    }
}
```

```

public String getUbicacion() {
    return ubicacion;
}

public void venderEntrada(double entrada){
    System.out.println("La entrada ha sido vendida ");
}

public void imprimirCine(){
    System.out.println("Cine: " +nombre+".");
    System.out.println("Ubicación: "+ubicacion+".\n");
    System.out.println("*****Cartelera*****");
}

```

## ***Clase Time***

Antes de crear la clase Película se creo la clase Time la cuál presenta una relación de composición hacia Película, la que dará la estructura del horario y duración de dicha clase.

```

package prac5;
public class Time {
    private int hora;
    private int minuto;
    private int segundo;

    public Time(int hora, int minuto, int segundo) {
        this.hora = hora;
        this.minuto = minuto;
        this.segundo = segundo;
    }

    public int getHora() {
        return hora;
    }

    public int getMinuto() {
        return minuto;
    }

    public int getSegundo() {
        return segundo;
    }

    public void setHora(int hora) {

```

```

        this.hora = hora;
    }

    public void setMinuto(int minuto) {
        this.minuto = minuto;
    }

    public void setSegundo(int segundo) {
        this.segundo = segundo;
    }

    public String imprimirDuracion(){
        return "Duración: "+hora+":"+minuto+":"+segundo;
    }
    public String imprimirHorario(){
        return "Horario: "+hora+":"+minuto;
    }
}

```

### ***Clase Película (Padre)***

```

package prac5;
public class Pelicula {
    protected Time duracion;
    protected Time horario;
    protected String director;
    protected double precio;
    protected String nombreP;
    //Constructor
    public Pelicula(String director, double precio, String nombreP) {
        this.nombreP = nombreP;
        this.director = director;
        this.precio = precio;
        duracion = new Time(02,25,32);
        horario = new Time(19,30,00);
    }

    public void imprimirCartelera(){
        System.out.println("Película: "+nombreP);
        System.out.println("Director: "+director);
        System.out.println(horario.imprimirHorario());
        System.out.println("Precio de entrada: "+precio);
    }
}

```

```

        public void venderEntrada(double entrada, String nombreP, double
precio){
            double total = 0;
            System.out.println("Se han vendido "+entrada+" entradas para
"+nombreP);
            total = entrada * precio;
            System.out.println("Recaudando un total de: $" + total + " mxn.\n");
        }
    }
}

```

### ***Clase Comedia (Hija)***

```

package prac5;
public class Comedia extends Pelicula{
    private int numPersonajesPrincipales;
    private int numChistes;
    private double nivelFelicidad;

    public Comedia(int numPersonajesPrincipales, int numChistes, double
nivelFelicidad, String director, double precio, String nombreP) {
        super(director, precio, nombreP);
        this.numPersonajesPrincipales = numPersonajesPrincipales;
        this.numChistes = numChistes;
        this.nivelFelicidad = nivelFelicidad;
        duracion = new Time(02,25,32);
        horario = new Time(19,10,00);
    }
    public int getNumPersonajesPrincipales() {
        return numPersonajesPrincipales;
    }

    public int getNumChistes() {
        return numChistes;
    }

    public double getNivelFelicidad() {
        return nivelFelicidad;
    }

    @Override
    public void imprimirCartelera(){
        precio = 50.00;
        System.out.println("Película: "+nombreP);
        System.out.println("Director: "+director);
    }
}

```

```

        System.out.println("Genero: Comedia");
        System.out.println(horario.imprimirHorario());
        System.out.println(duracion.imprimirDuracion());
        System.out.println("Precio de entrada: $" + precio + " mxn");
        System.out.println("Número de personajes principales:
"+numPersonajesPrincipales);
        System.out.println("Cantidad de chistes: "+numChistes);
        System.out.println("Nivel de Felicidad: "+nivelFelicidad+"\n");

    }
    @Override
    public void venderEntrada(double entrada, String nombreP, double
precio){
        double total = 0;
        System.out.println("Se han vendido "+entrada+" entradas para
"+nombreP);
        total = entrada * precio;
        System.out.println("Recaudando un total de: $" + total + " mxn.\n");
    }
}

```

### ***Clase Terror (Hija)***

```

package prac5;
public class Terror extends Pelicula{
    private String personajeTerror;
    private double nivelMiedo;

    public Terror(String personajeTerror, double nivelMiedo, String director,
double precio, String nombreP){
        super(director, precio, nombreP);
        this.personajeTerror = personajeTerror;
        this.nivelMiedo = nivelMiedo;
        duracion = new Time(01,55,43);
        horario = new Time(20,50,00);
    }

    public String getPersonajeTerror() {
        return personajeTerror;
    }

    public double getNivelMiedo() {
        return nivelMiedo;
    }
}

```

```

@Override
public void imprimirCartelera(){
    precio = 65.00;
    System.out.println("Película: "+nombreP);
    System.out.println("Director: "+director);
    System.out.println("Genero: Terror");
    System.out.println(horario.imprimirHorario());
    System.out.println(duracion.imprimirDuracion());
    System.out.println("Precio de entrada: $" + precio + " mxn");
    System.out.println("Personaje de Terror: "+personajeTerror);
    System.out.println("Nivel de miedo: "+nivelMiedo+"\n");
}
@Override
public void venderEntrada(double entrada, String nombreP, double
precio){
    double total = 0;
    System.out.println("Se han vendido "+entrada+" entradas para
"+nombreP);
    total = entrada * precio;
    System.out.println("Recaudando un total de: $" + total + " mxn.\n");
}
}

```

### ***Clase Accion (Hija)***

```

package prac5;
public class Accion extends Pelicula{
    private int numPersecuciones;
    private int numEnfrentamientos;
    private double nivelViolencia;

    public Accion(int numPersecuciones, int numEnfrentamientos, double
nivelViolencia, String director, double precio, String nombreP){
        super(director, precio, nombreP);
        this.numPersecuciones = numPersecuciones;
        this.numEnfrentamientos = numEnfrentamientos;
        this.nivelViolencia = nivelViolencia;
        duracion = new Time(02,19,11);
        horario = new Time(15,20,00);
    }
    public int getNumPersecuciones() {
        return numPersecuciones;
    }
    public int getNumEnfrentamientos() {

```

```

        return numEnfrentamientos;
    }

    public double getNivelViolencia() {
        return nivelViolencia;
    }
    @Override
    public void imprimirCartelera(){
        precio = 80.00;
        System.out.println("Película: "+nombreP);
        System.out.println("Director: "+director);
        System.out.println("Genero: Accion");
        System.out.println(horario.imprimirHorario());
        System.out.println(duracion.imprimirDuracion());
        System.out.println("Precio de entrada: $" + precio + " mxn");
        System.out.println("Cantidad de persecuciones: " + numPersecuciones);
        System.out.println("Cantidad de enfrentamientos: " + numEnfrentamientos);
        System.out.println("Nivel de violencia: " + nivelViolencia + "\n");
    }
    @Override
    public void venderEntrada(double entrada, String nombreP, double precio){
        double total = 0;
        System.out.println("Se han vendido " + entrada + " entradas para " + nombreP);
        total = entrada * precio;
        System.out.println("Recaudando un total de: $" + total + " mxn.\n");
    }
}

```

## Clase Main

```

package prac5;

public class Main {

    public static void main(String[] args){

        Cine c1= new Cine("Cine Regis","Av de la luz no. 23");

        c1.imprimirCine();

        Pelicula pc1 = new Comedia(3,80,79.98,"Nima Nourizadeh",50.00,"Proyecto X");
    }
}

```



```

Película pt1 = new Terror("Jason",89.99,"Sean S.
Cunningham",60.00,"Viernes 13");

Película pa1 = new Accion(4,7,80.56,"Brian De Palma",80.00,"Scarface");

pc1.imprimirCartelera();

pt1.imprimirCartelera();

pa1.imprimirCartelera();

pc1.venderEntrada(5, "Proyecto X", 50.00);

pt1.venderEntrada(2, "Viernes 13", 60.00);

pa1.venderEntrada(10, "Duro de matar",80.00);

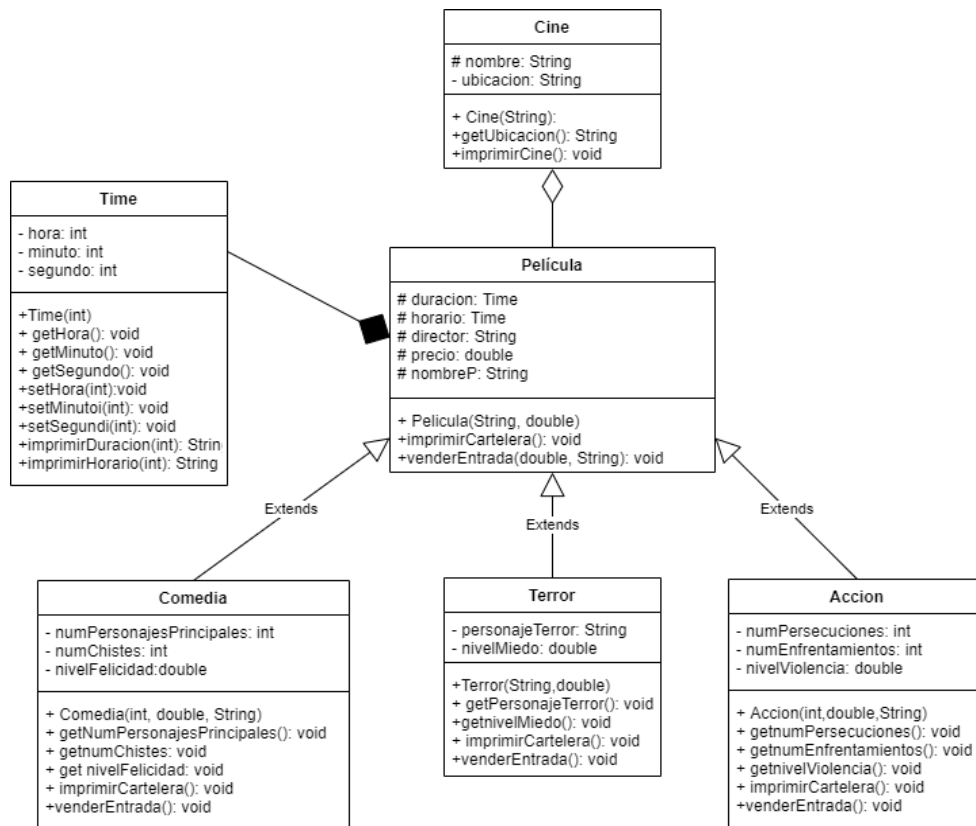
}

}

```

## Funcionamiento

### Diagrama UML final



## **Salida final del programa**

Cine: Cine Regis.

Ubicación: Av de la luz no. 23.

\*\*\*\*\*Cartelera\*\*\*\*\*

Película: Proyecto X

Director: Nima Nourizadeh

Genero: Comedia

Horario: 19:10

Duración: 2:25:32

Precio de entrada: \$50.0 mxn

Número de personajes principales: 3

Cantidad de chistes: 80

Nivel de Felicidad: 79.98

Película: Viernes 13

Director: Sean S. Cunningham

Genero: Terror

Horario: 20:50

Duración: 1:55:43

Precio de entrada: \$65.0 mxn

Personaje de Terror: Jason

Nivel de miedo: 89.99

Película: Scarface

Director: Brian De Palma

Genero: Accion

Horario: 15:20

Duración: 2:19:11

Precio de entrada: \$80.0 mxn

Cantidad de persecuciones: 4

Cantidad de enfrentamientos: 7

Nivel de violencia: 80.56

Se han vendido 5.0 entradas para Proyecto X

Recaudando un total de: \$250.0 mxn.

Se han vendido 2.0 entradas para Viernes 13

Recaudando un total de: \$120.0 mxn.

Se han vendido 10.0 entradas para Duro de matar

Recaudando un total de: \$800.0 mxn.

BUILD SUCCESSFUL (total time: 0 seconds)

### Pruebas de escritorio

Datos	VenderEntrada (Comedia)	venderEntrada (Accion)	venderEntrada (Terror)
Entradas vendidas	5	10	2
Pelicula	Proyecto X	Duro de matar	Viernes 13
Recaudado total	\$250	\$800	\$120

Datos	imprimirCartelera (Comedia)	imprimirCartelera (Accion)	imprimirCartelera (Terror)
Pelicula	Proyecto x	Scarface	Viernes 13
Director	Nima Nourizadeh	Brian de Palma	Sean S. Cunningham
Duración	2:25:32	2:19:11	1:55:43
Horario	19:10hrs	15:20hrs	20:50hrs
Precio de entrada	\$50	\$80	\$89.99

imprimirCartelera (Comedia)	imprimirCartelera (Terror)	imprimirCartelera (Acción)
Proyecto X	Viernes 13	Scarface
Nima Nourizadeh	Sean S. Cunningham	Brian De Palma
Comedia	Terror	Acción
19:10	20:50	15:20
02:25:32	01:55:43	2:19:11
\$50.00 mxn	\$65.00 mxn	\$80.00 mxn
3	Jason	4
80	89.99	7
79.98		80.56

<i>venderEntrada(Comedia)</i>	<i>venderEntrada(Terror)</i>	<i>venderEntrada(Acción)</i>
<b>2 (100)</b>	1 (65)	4 (320)
<b>3 (150)</b>	2 (130)	3 (240)
<b>1 (50)</b>	3 (195)	1 (80)
<b>4 (200)</b>	5 (325)	4 (320)
<b>5 (250)</b>	2 (130)	3 (240)
<b>2 (100)</b>	2 (130)	5 (400)
<b>Total: 17 (\$850.00 mxn)</b>	Total: 15 (\$975.00 mxn)	Total: 20 (\$1600.00 mxn)

## Conclusiones

Al realizar esta práctica nos percatamos que la herencia es muy útil en la POO y en general en las ciencias de la computación, ya que está presente en casi toda la programación. Nos resultó sencillo el entender todo este tema y claro tuvimos que investigar un poco para poder establecer la estructura del horario.

Se deben comprender a fondo todas las características y como es que se implementan los métodos, las clases para su correcto funcionamiento.

## Bibliografía:

[http://odin.fi-b.unam.mx/salac/practicaspoo/P07-POO\\_Herencia.pdf](http://odin.fi-b.unam.mx/salac/practicaspoo/P07-POO_Herencia.pdf)

[https://www.ctr.unican.es/asignaturas/pp/apuntes/03-herencia\\_3en1.pdf](https://www.ctr.unican.es/asignaturas/pp/apuntes/03-herencia_3en1.pdf)

<https://www.codingame.com/playgrounds/50747/herencia-en-c-practica-3>