



1 Cruzar pista

Una persona que desea cruzar una pista debe de mirar a ambos lados porque cualquier medio de transporte puede pasar y arrollarla. Si se sabe que los medios de transporte terrestre que encontramos comúnmente transitando por una avenida son el carro, la bicicleta y el bus y éstos pueden transitar de derecha a izquierda o viceversa y a velocidad variada, se le solicita que haciendo uso de la técnica de la programación orientada a objetos, usted elabore una aplicación que permita cruzar a una persona de un lado a otro en una concurrida avenida.

Para realizar la aplicación deberá realizar lo siguiente:

- **Diagrama de Clases**
- **Clase Terrestre**
 - ✓ Crear una clase llamada **Terrestre**, esta clase tiene los atributos que agrupan a todos los medios de transporte (carro, bus y bicicleta) y le permite moverse de un lado a otro en la avenida. En nuestra simulación los medios de transporte se mueven de forma automática.
- **Clase Carro**
 - ✓ Clase **Carro**, un carro es un medio de transporte que se desplaza a una velocidad variable y si atropella a una persona le causa la muerte. Los carros, en la simulación que haremos solo puede moverse de izquierda a derecha. En pantalla un carro se puede representar de la siguiente manera

```

  _~\
 [ _ ]
 ( )   ( )

```

- **Clase Bus**
 - ✓ Clase **Bus**, un bus es un medio de transporte que se desplaza a una velocidad variable mayor a la del carro y si atropella a una persona le causa la muerte. Los buses, en la simulación que haremos solo puede moverse de derecha a izquierda. En pantalla un bus se puede representar de la siguiente manera

```

  _/ _| | | | | | | |
 (  _| | | | | | | |
  ==--OO-----OO==

```

- **Clase Bicicleta**
 - ✓ Clase Bicicleta, una bicicleta es un medio de transporte que se desplaza a una velocidad constante y es más lento que el carro y el bus. Si la bicicleta atropella a una persona no le causa la muerte. Las bicicletas, en la simulación que haremos pueden moverse de derecha a izquierda o viceversa. En pantalla una bicicleta se puede representar de la siguiente manera

```

  .._ \
 (o)(o)

```



/_..
(o)(o)

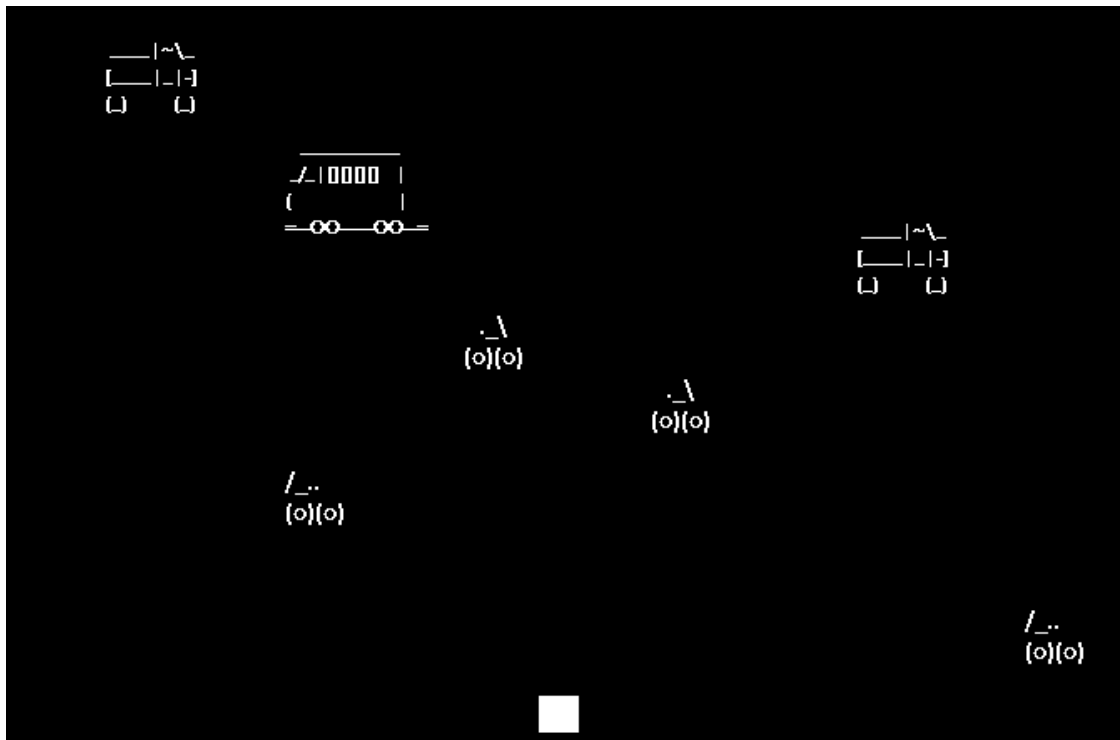
- **Clase Persona**

- ✓ Clase Persona, es aquel ser que desea cruzar la pista y se desplaza únicamente con las teclas direccionales. Si colisiona con un medio de transporte que le causa la muerte la simulación termina automáticamente. La persona será representada por un cuadrado (ASCII 254 ■) verde

- **Clase Juego**

- ✓ Clase Juego, esta clase tiene una colección de medios de transporte y a la persona que desea cruzar la avenida. En su colección debe de tener al menos uno de los medios de transporte arriba mencionados y que reflejen todos sus movimientos y de esta forma podamos apreciar correctamente la simulación.

EJEMPLO



- **Funcionamiento del programa**

- ✓ El programa debe tener las clases definidas e implementadas y debe funcionar correctamente sin errores de compilación.



2 Gato y ratón

Es conocido en el mundo animal que los gatos son buenos cazadores de ratones, por lo que simularlo en una computadora puede resultar muy divertido. Para ello se le solicita elaborar un programa que, haciendo uso de POO, simule un gato que persigue a ratones.

La temática es que se tiene un Gato que persigue a 10 ratones y debe de atraparlos. Para esto, se debe de definir una clase llamada **Ratón** que se desplaza por pantalla y que huye del Gato. El **Gato** es otra clase que busca atrapar a los ratones pero que se desplaza únicamente cuando el usuario presiona una tecla para hacerlo.

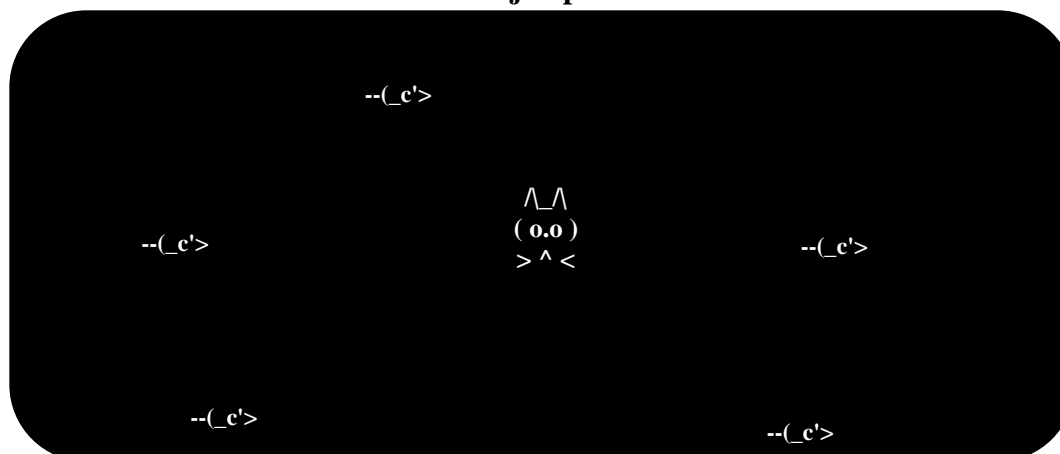
ASCII ART	
$\begin{array}{c} \wedge \wedge \\ (\text{ o.o }) \\ > ^ < \end{array}$	--(_c'>

Utilice los Ascii art para representar al gato y ratón

Tenga en cuenta presente lo siguiente:

- ✓ Definir las clases necesarias para representar las relaciones de herencia
- ✓ Definir una clase para el Ratón
- ✓ Definir una clase para el Gato
- ✓ Definir una clase para el arreglo de ratones e insertar los “n” ratones que tiene el juego.
El valor “n” es el número máximo de ratones y es aleatorio en el intervalo siguiente:
7 <= n <= 15.
- ✓ Cuando el gato atrapa a un ratón este se elimina del arreglo y desaparece de la pantalla.
- ✓ El gato se mueve únicamente con las teclas direccionales o con las teclas “AW S D “
- ✓ El movimiento de los ratones debe ser siempre huyendo del gato.
- ✓ Los ratones aparecen cada 2 segundos y tienen una posición inicial aleatoria
- ✓ La simulación termina cuando el gato atrapa a los “n/2” ratones. Considere el valor entero.

Ejemplo





UPC
Universidad Peruana
de Ciencias Aplicadas

PROGRAMACIÓN 2
Hoja de Ejercicios Nro. 7a
CICLO 2019-2

Temas:
Relaciones entre clases