UNIVERSIDAD ICESI

Departamento de TIC
Algoritmos y Estructuras de Datos
Laboratorio Unidad 3
Concurrencia, Dibujo Básico en 2D y Persistencia

Objetivos

Unidad 3: Concurrencia, Dibujo Básico en 2D y Persistencia

Al finalizar esta unidad, el estudiante estará en capacidad de:

OE3.1 Desarrollar un programa que maneje concurrencia, de manera que sea posible que ejecute más de una parte del programa de manera simultánea, utilizando hilos de ejecución (threads).

OE3.2 Construir interfaces de usuario que incluyan gráficas en 2 dimensiones como una alternativa en la presentación de información al usuario.

OE3.3 Hacer persistir el estado del modelo de solución del problema durante la ejecución de un programa y restaurarlo cuando se requiera.

OE3.4 Manipular archivos de texto y utilizarlos para implementar requerimientos del cliente relacionados con persistencia.

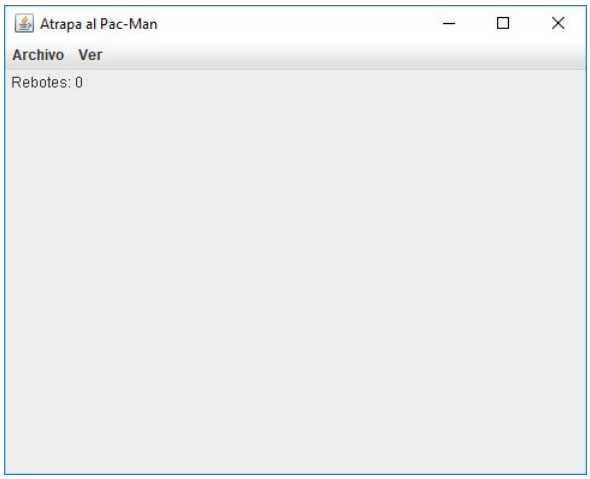
Preparación

- Lea cuidadosamente el enunciado, la documentación suministrada y cada uno de los puntos que debe desarrollar antes de empezar su desarrollo. Pregunte a su profesor cualquier duda respecto al enunciado o a los requerimientos funcionales que debe desarrollar.
- Lea cuidadosamente la rúbrica del laboratorio de la unidad 3 (ver rúbrica).
- El trabajo debe ser realizado individualmente.
- El trabajo será entregado en la fecha y hora establecida en Moodle.

Enunciado

JUEGO ATRAPA AL PAC-MAN

Con el auge de las aplicaciones para móviles y sobre todo los juegos para estos dispositivos, usted ha notado que para que un juego tenga éxito no tiene que ser necesariamente muy elaborado. Usted ha visto juegos bastante sencillos que han tenido gran acogida precisamente gracias a su sencillez. Alguien tiene una idea con videojuegos arcade y quiere desarrollar rápidamente un prototipo para hacer pruebas sobre el juego para ver si es prometedor, por lo tanto ha decidido desarrollar el prototipo utilizando Java y las herramientas conceptuales y técnicas adquiridas en su actual curso de algoritmos.

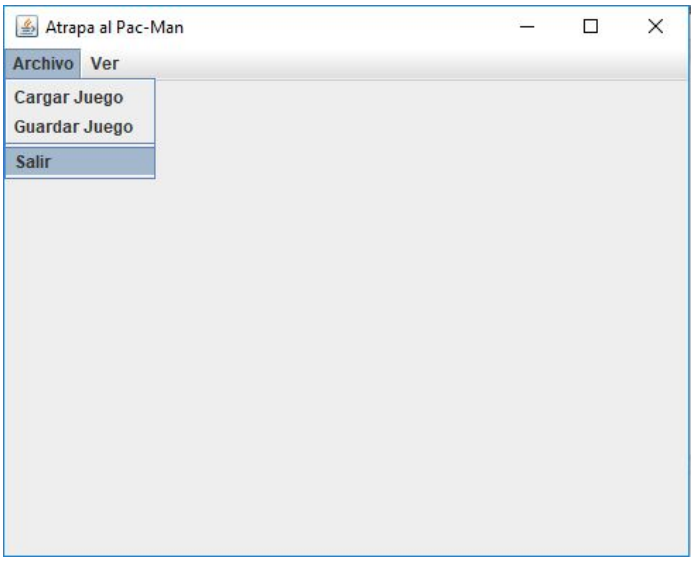


La idea, como se indicó antes, es bastante sencilla y se explica a continuación: aparecen unos Pac-Man en la pantalla moviéndose, algunos horizontal y otros verticalmente. Durante su movimiento, si el Pac-Man alcanza un extremo de la ventana de juego, éste rebotará y se moverá ahora en sentido contrario. El jugador debe detenerlos haciendo clic sobre cada una de los Pac-Man que aparecen en la pantalla, lo más rápido posible y antes de que reboten. Por cada rebote, el contador de rebotes aumentará. El mejor jugador es aquel que detenga todas los Pac-Mac's con la menor cantidad de rebotes, considere que los Pac-Man también deben rebotar entre sí y al momento de rebotar también se deben seguir moviendo en sentido contrario a su movimiento.

El Pac-Man es un círculo amarillo al que le falta un sector, por lo que parece tener boca, usted debe construir el Pac-Man usando las primitivas que proporciona el lenguaje.

UNIVERSIDAD ICESI

Departamento de TIC
Algoritmos y Estructuras de Datos
Laboratorio Unidad 3
Concurrencia, Dibujo Básico en 2D y Persistencia



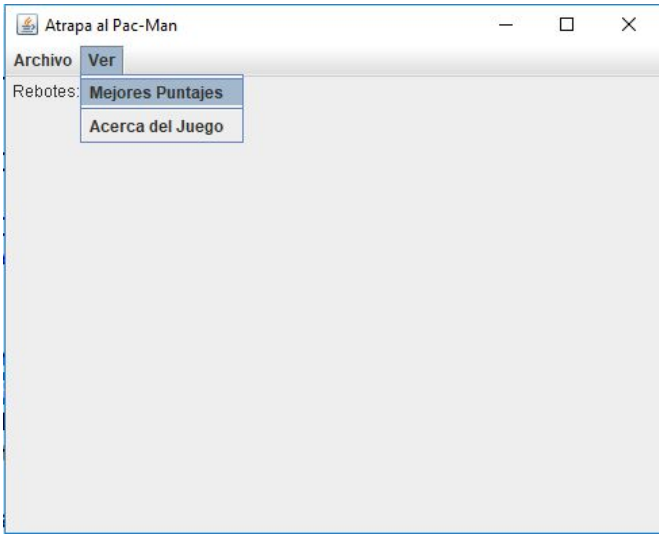
La configuración de un nuevo juego se puede (debe) cargar desde un archivo de texto a través de la opción del menú que se muestra en la imagen de la izquierda. El archivo de la configuración de un (1) posible juego se muestra a continuación:

```
#nivel
0
#radio posX posY espera direccion rebotes
50  70  90  10  IZQUIERDA   0   false
70  120 120 12  DERECHA 0   false
80  350 230 14  ARRIBA  0   false
60  280 10  16  ABAJO   0   false
```

Las líneas en blanco deben ser ignoradas. Las que inicien con el símbolo # son líneas de comentarios, por tanto también deben ser ignoradas. La primera línea válida es un número entero que indica el nivel de dificultad del juego que tiene ese archivo. Son tres niveles posibles 0, 1 y 2. En el resto del archivo, cada línea válida representa un Pac-Mac y tiene los valores de las características del Pac-Mac en el siguiente orden y separados por tabulador (“\t”): diámetro del Pac-Mac, posición en X, posición en Y, tiempo de espera (para su movimiento), dirección inicial del movimiento del Pac-Mac, cantidad de rebotes y si está detenido.

Su programa debe tener la capacidad de leer un archivo como el descrito, el cual debe ser leído y cargado en los objetos del mundo a través del método cargarJuego.

Si en medio del juego se desea guardar el estado actual de éste, el programa debe tener también la posibilidad de guardar el estado actual en un archivo de texto con el mismo formato anterior (pero evidentemente con los valores actuales de los Pac-Mac’s). Esta información será guardada a través del método guardarJuego.



Una vez el jugador ha detenido todos los Pac-Mac’s, si su puntaje (la cantidad de rebotes) se encuentra dentro de las mejores 10 del nivel de dificultad, entonces se le pide su nombre y éste junto con el puntaje queda guardado en el Hall de la Fama. Pero para que esto ocurra de forma permanente se requiere que los datos sean persistentes. Sin embargo, un archivo de texto sería demasiado fácil de modificar fuera del programa, por lo que un usuario podría agregarse fácilmente a través de un editor de texto y ponerse un excelente (o imposible) puntaje, que realmente no es cierto. Por lo anterior, tanto la escritura y la lectura de los puntajes se llevará a cabo a través de serialización. El objeto que se serializará será el objeto de

la clase Puntaje. Cuando el programa se cierra debe guardarse la información del puntaje y al abrirse de nuevo, si se consulta esta información a través de la opción del menú debe visualizarse pues el objeto ha sido deserializado con la información que tenía almacenada.

Entregables. Unidad 1.
1. Requerimientos Funcionales.
2. Diagrama de clases de modelo y control de la interfaz (no generado automáticamente)
3. Implementación completa de todos los requerimientos en Java.
4. Tabla de trazabilidad de requerimientos vs métodos (tabla con una columna de los requerimientos, tal que, por cada requerimiento se indica en la columna siguiente todos los métodos que contribuyen a resolverlo).3

Fecha de Entrega: Martes 19 de marzo de 2019 a las 11:59 AM a través de Moodle. El laboratorio debe trabajarse y entregarse **individualmente**.