

## Segundo Laboratorio Estructura de Datos II

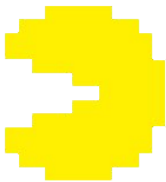
### Descripción:

Elaboración del juego Pacman modificado en el lenguaje de programación Java bajo el IDE NetBeans.

### Requerimientos:

- Debe existir un (1) sólo jugador principal controlado por el usuario mediante las teclas Up, Down, Left y Right del teclado.
- Debe existir un (1) sólo enemigo que persigue todo el tiempo al jugador principal y tiene movimiento propio, es decir, no puede ser controlado de forma externa. Debe implementar algoritmos de camino mínimo para su inteligencia.
- Debe existir un contador de puntos que se muestre todo el tiempo en la ventana de juego.
- Debe colocarse en el tablero de juego un total de diez (10) objetivos, que al ser tomados por el jugador aumentarán su puntaje.
- Al recolectarse los 10 objetivos, el juego habrá terminado y el jugador ganará.
- Si el enemigo colisiona con el jugador principal, el juego habrá terminado y el jugador principal pierde.
- Al finalizar el juego se debe mostrar al usuario si ganó o perdió y tanto el jugador como el enemigo no deberán moverse más.

### Personajes:



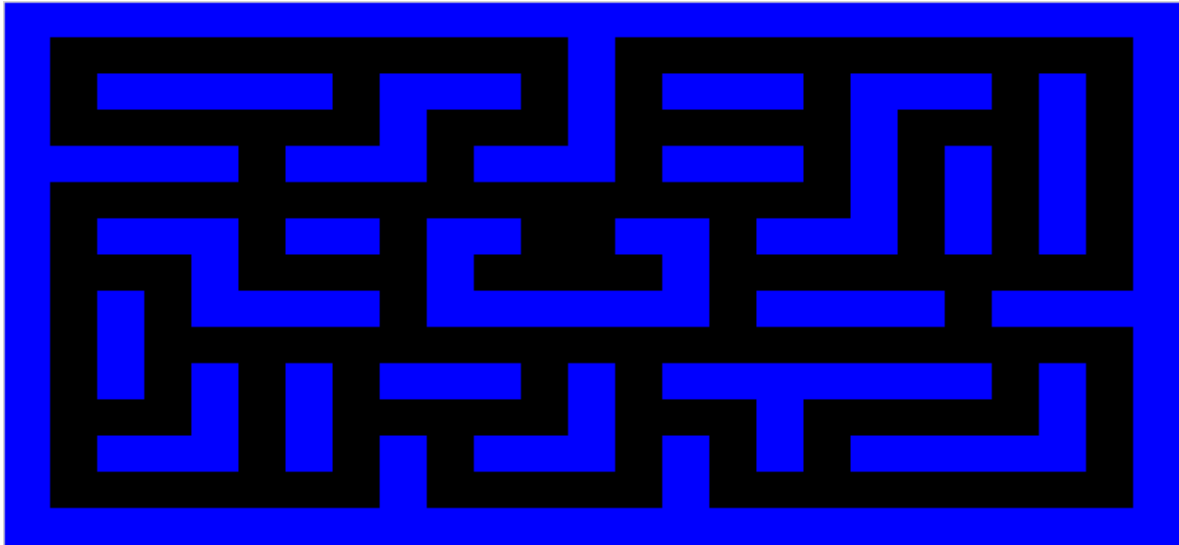
Jugador principal (Pacman): Es controlado por el usuario, debe recoger los objetivos y huir del enemigo. Al recoger todos los objetivos, gana; si es colisionado con el enemigo, pierde.



Enemigo: Tiene movimiento autónomo mediante algoritmos de camino mínimo. Debe perseguir todo el tiempo a Pacman y una vez que lo alcanza, Pacman pierde.

**Mundo:**

TODOS los juegos implementados tendrán un solo mundo y es como se muestra a continuación:



Para la facilidad de su implementación, se adjunta un archivo de texto (.txt) con la codificación del mundo como ceros y unos.

El mundo se compone de vías y muros,

- Vías: de color negro, mediante los cuales se pueden mover tanto el personaje principal como el enemigo.
- Muros: de color azul, sirven de barreras y a través de ellos no se puede transitar.

**Puntaje:**

Por cada segundo que transcurra el juego se le dará al usuario un (1) punto. Además, por cada objetivo alcanzado se le dará al usuario diez (10) puntos. El puntaje debe estar visible todo el tiempo en la ventana de juego.

**Especificaciones técnicas:**

- El juego debe ser desarrollado en Java y el IDE NetBeans.
- Tanto el jugador como el enemigo deben funcionar en un hilo separado del hilo principal.
- Se debe implementar algún algoritmo de camino mínimo para que el enemigo persiga al jugador principal.
- El mundo debe ser modelado como un grafo.

### Objetivos:

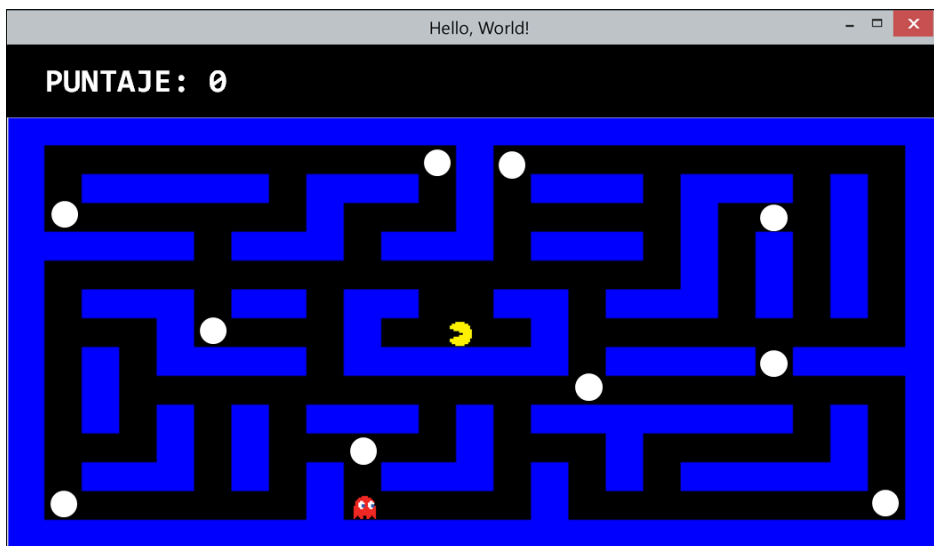
Los objetivos son círculos blancos ubicados en posiciones específicas del tablero de juego. Dan al personaje diez puntos más una vez son alcanzados.

### Posiciones iniciales de los elementos del juego:

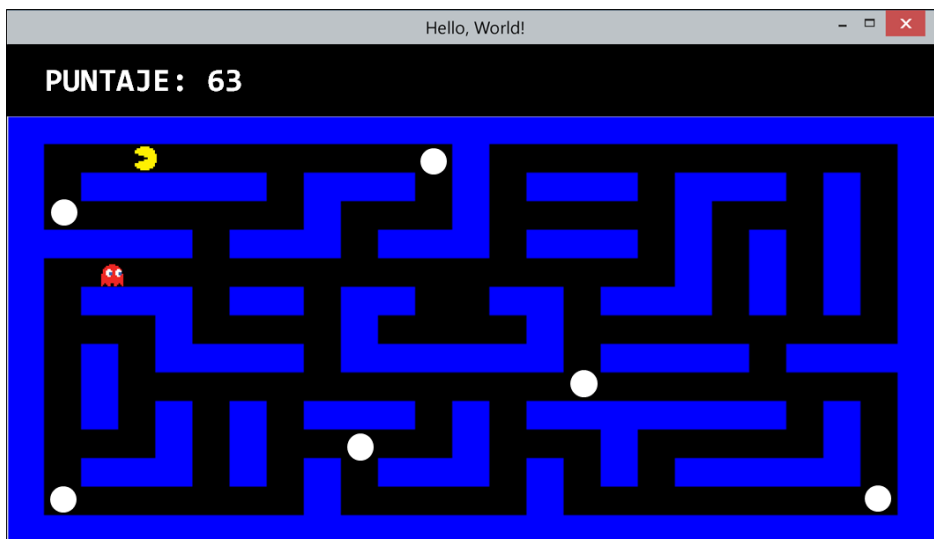
Tanto el jugador principal como el enemigo y los objetivos deben estar distribuidos al iniciar el juego como se muestra en el primer pantallazo mostrado más abajo.

### Pantallazo del juego esperado:

Ventana al momento de iniciar el juego.



Ventana al transcurrir el juego.



### **Requerimientos de trabajo:**

- Equipos: el juego será desarrollado de forma individual. No se admiten grupos.
- Fecha de entrega: 13 de noviembre de 2017.
- Entregables:
  - Código fuente del juego.
  - Documentación, que debe contener:
    - Hoja de presentación.
    - Tabla de contenido.
    - Introducción.
    - Descripción general del juego.
    - Requerimientos funcionales.
    - Requerimientos no funcionales.
    - Descripción técnica de la solución (detalle a nivel de programación de la solución: clases, hilos, algoritmos de grafos, etc.).
    - Pantallazos del juego, 1 al iniciar el juego, 1 durante el juego y 1 al finalizar el juego.
    - Conclusiones.
    - Referencias, si las hay, en formato IEEE.

### **Bonificaciones:**

Los siguientes no son requerimientos. Se darán bonificaciones por cada uno de los siguientes elementos implementados:

- Sonido de fondo reproduciéndose todo el tiempo.
- Sonido por eventos: alcanzar un objetivo, enemigo alcanza a Pacman, el juego inicia, el juego acaba.
- Rellenar los muros con una imagen en vez de un cuadro de color sólido.
- Cuando un enemigo alcanza un objetivo, éste se cambia de posición a un lugar aleatorio del tablero de juego (donde Pacman pueda alcanzarlo igual).

¡ÉXITOS!