

UNIVERSIDAD DEL VALLE DE GUATEMALA
*Departamento de Ingeniería en Ciencias de la Computación
y Tecnologías de la Información*
Ing. Kimberly Marisol Barrera González



Proyecto 1

Snake

Integrantes:

Adriana Martínez - 24086

Mishell Ciprian -231169

Belén Monterroso - 231497

Pablo Andrés Cabrera Argüello - 231156

Fase 1: Análisis del Juego Snake

Resumen del proyecto

El proyecto consiste en el desarrollo del juego Snake implementando C++ con hilos POSIX (pthreads) y control grafico en consola mediante ncurses.

En la fase 1 se definió el análisis del juego (funcionamiento, elementos, interacción, cálculos necesarios.) En la fase 2, el objetivo fue diseñar e implementar el entorno grafico en consola, integrando menús, pantallas de instrucciones, puntajes destacados y representación en tiempo real del tablero de juego ASCII-Art.

Este entorno permite que el jugador tenga una interacción fluida con el sistema, seleccionando modos de juego y controlando la serpiente en tiempo real.

Diseño del entorno grafico:

Estilo visual:

- ASCII-Art para representar la serpiente, comida y trampas.
- Tablero con dimensiones fijas de 60x24 celdas, delimitado con caracteres | y -.
- HUD (Head-Up Display) que muestra el nivel, velocidad, tiempo restante y puntajes de cada jugador.
- El menú principal tiene un titulo decorado con arte ASCII para dar estilo al juego.

Pantallas implementadas:

1. Menú principal:
 - a. Opciones:
 - i. 1.Un jugador.
 - ii. 2. Dos jugadores.
 - iii. 3. Seleccionar dificultad.
 - iv. 4. Puntajes destacados,
 - v. 5. Instrucciones.
 - vi. 6. Salir.
 - b. Incluye titulo con ASCII y diseño centrado.
2. Pantalla de instrucciones.
 - a. Explicaciones de controles:
 - i. Jugador 1: WASD
 - ii. Jugador 2: flechas de dirección.

- iii. Explicación de símbolos visuales. @ (comida, +7 pts), X (trampas, muerte inmediata).
- 3. Pantalla de puntajes destacados:
 - a. Puntajes almacenados en el archivo scores.txt.
 - b. Lista de las partidas recientes con fecha y hora.
- 4. Tablero de juego:
 - a. Campo de juego de 60x24 con bordes y serpientes.
 - b. Comida generada en posiciones aleatorias.
 - c. Trampas que aumentan en número según nivel.
 - d. HUD dinámico abajo del campo de juego que muestra el nivel, velocidad, tiempo, modo y puntaje.
- 5. Pantalla de Game over.
 - a. ASCII “game over “.
 - b. Puntajes finales.
 - c. Opción de volver al menú principal o salir del juego.
- 6. Pantalla para seleccionar nivel de dificultad.
 - a. Fácil 200 ms
 - b. Media 150 ms
 - c. Difícil 100 ms

Justificación de decisiones graficas:

- Uso de ASCII-Art: Se eligió por su compatibilidad multiplataforma y estética retro coherente con juegos clásicos.
- HUD fijo: Garantiza que el jugador tenga información crítica siempre visible (tiempo, nivel, puntaje).
- Separación de pantallas: Facilita la navegación y comprensión del flujo del juego.
- alineación (ncurses): Se utilizó alineación consistente para que el diseño sea ordenado.

Funciones y librerías utilizadas:

librerías:

- <ncurses.h> → Control de pantalla y entrada en tiempo real.
- <pthread.h> → Creación y manejo de hilos POSIX.
- <unistd.h> → Funciones de control de tiempo (usleep, sleep).
- <cstdlib> y <ctime> → Generación de posiciones aleatorias.

- <vector>, <string>, <fstream>, <algorithm> → Estructuras de datos y manejo de archivos para puntajes.

Funciones principales de ncurses:

- initscr(), endwin() → Inicialización y cierre.
- cbreak(), noecho(), keypad(), nodelay() → Configuración de entrada en tiempo real.
- mvaddch(), mvprintw(), refresh(), clear() → Dibujo de elementos.
- getch() → Captura de teclas presionadas.

Funciones graficas implementadas:

- drawBorders(): Dibuja los límites del tablero.
- drawHUD(): Imprime información de nivel, tiempo, puntajes.
- drawSnake(): Representa la serpiente en movimiento.
- drawFood(): Coloca la comida con símbolo @.
- drawTraps(): Representa trampas con símbolo X.
- showMenu(): Muestra menú principal.
- showInstructions(): Pantalla de controles.
- showScores(): Lee y muestra puntajes.
- gameOverScreen(): Pantalla final con resultados.

Conclusiones:

En esta fase 2 se consolidó el diseño visual del juego Snake en consola. Se lograron implementar todas las pantallas requeridas, con el menú navegable, la visualización de puntajes y la representación en tiempo real de la serpiente, comida y trampas.

Tomando este entorno como base para integrar la lógica completa en la fase 3, donde se va a profundizar el uso de múltiples hilos para paralelismo.

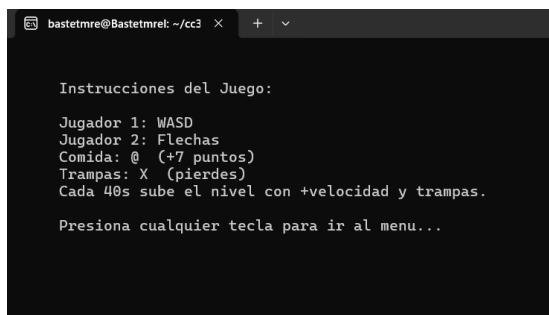
Fase 1:

<https://github.com/bastmre44/Proyecto-Microprocesadores/blob/main/Proyecto%201%20Micros%20%202025.docx>

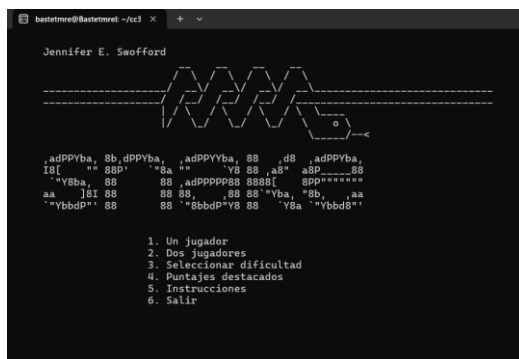
Snake.cpp:

<https://github.com/bastmre44/Proyecto-Microprocesadores/blob/main/Snake.cpp>

Imágenes de pantallas.



Instrucciones del juego.



Menú principal(nombre de la autora del dibujo de serpiente.)



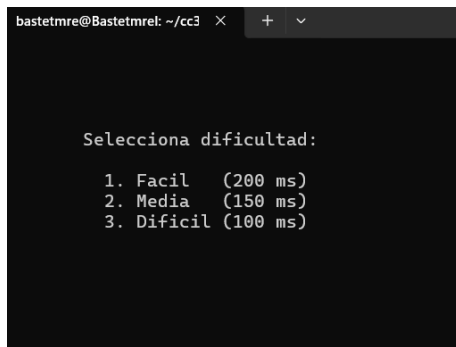
Cuadro de un solo jugador.



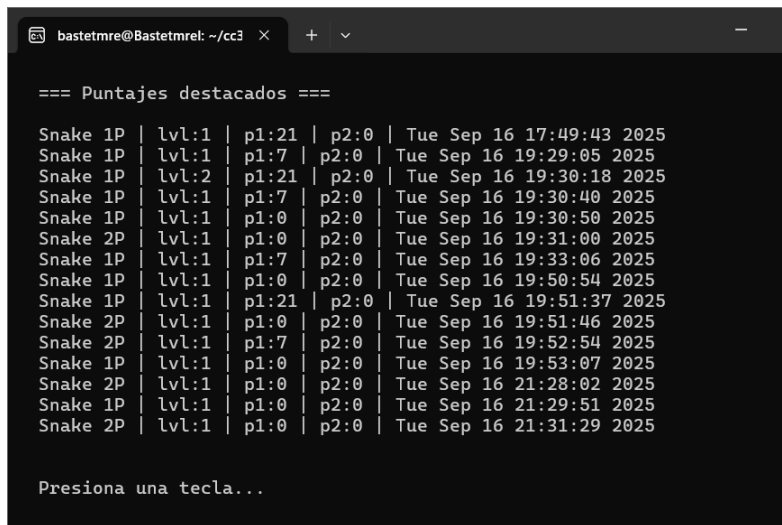
Pantalla de Game Over.



Pantalla con 2 jugadores.



Nivel de dificultad.



Puntajes destacados.