

Informe Técnico: Juego del Ahorcado en C++

Juan Manuel Cuervo Gaitan

Samuel Marín Vanegas

Fecha: 15/10/2025

1. Introducción

El juego del Ahorcado es un clásico juego de adivinación de palabras, donde un jugador intenta descubrir una palabra secreta proponiendo letras. Si la letra está en la palabra, se revela su posición; si no, se incrementa un contador de errores hasta que el jugador gana o pierde.

Los objetivos del proyecto incluyen:

- Desarrollar un programa que seleccione aleatoriamente una palabra de una lista predefinida y gestione el estado del juego.
- Garantizar una interfaz simple y amigable, con visualización del progreso y un dibujo del ahorcado
- Manejar errores comunes, como entradas inválidas o letras repetidas, para mejorar el programa.

2. Diseño de la Solución

En esta sección se describe el enfoque general para resolver el problema, incluyendo las estructuras de datos seleccionadas, su justificación y el flujo principal del programa.

2.1 Explicación de las Estructuras de Datos Utilizadas

El programa utiliza principalmente vectores de la biblioteca estándar de C++ (`std::vector`), que son contenedores dinámicos para almacenar colecciones de datos.

- `vector<string> palabras`: Almacena la lista de palabras posibles para el juego.
- `vector<char> estado`: Representa el estado actual de la palabra secreta, inicializado con guiones bajos.
- `vector<char> letrasUsadas`: Registra las letras que el jugador ha intentado,

evitando repeticiones.

También se utiliza una cadena de texto (string palabraSecreta) para almacenar la palabra seleccionada, y variables simples como int intentos para controlar los errores.

2.2 Justificación del Uso de Arreglos de Caracteres

Se optó por vectores de caracteres debido a su eficiencia y simplicidad para manejar secuencias de letras. Los vectores permiten un tamaño dinámico, son seguros en C++ moderno y facilitan el acceso rápido por índice.

2.3 Descripción del Flujo Principal del Programa

El flujo del programa sigue una estructura secuencial:

1. Inicialización del juego y palabra aleatoria.
2. Bucle principal: muestra estado, valida letra, actualiza progreso.
3. Finalización: muestra resultado y determina si el jugador ganó o perdió.

El bucle while dentro de main() controla el juego de forma modular y clara.

3. Implementación

Esta sección detalla cómo se llevaron a cabo los aspectos clave de la programación.

3.1 Algoritmo de Verificación de Letras

La función verificarLetra recorre la palabra secreta y actualiza el vector de estado cuando hay coincidencias. Complejidad: $O(n)$.

3.2 Manejo de la Validación de Entrada

La función validarEntrada asegura que la entrada sea una sola letra alfabética y no repetida, lanzando excepciones si no cumple las condiciones.

3.3 Estrategia para Evitar Letras Repetidas

El vector letrasUsadas registra las letras ya intentadas, evitando repeticiones con una búsqueda lineal $O(m)$.

4. Conclusiones: Desafíos Enfrentados y Soluciones Implementadas

El proyecto permitió aplicar estructuras dinámicas y manejo de errores en C++.

4.1 Desafíos Enfrentados

- Alineación del dibujo.
- Manejo de entradas inválidas.
- Generación verdaderamente aleatoria de palabras.

4.2 Soluciones Implementadas

- Corrección del dibujo mediante cadenas fijas.
- Manejo de excepciones en validarEntrada.
- Uso de `srand(time(0))` para aleatoriedad controlada.