

**Costa Rica Institute of Technology**

**Algorithm Analysis**

**PROJECT #1**

**Professor:** Eng. Jose Angel Campos Aguilar

**Students:**

Brasly Villarebia Morales *2023105915*

Sidney Salazar Jimenez 2023282075

Jeanpooll Domenech Cerdas

**Submission Date:**

April 30, 2025

Group 60  
 First Semester, 2025

Introducción

Los juegos de laberintos representan un desafío mental interesante para el jugador, quien debe encontrar el camino más eficiente desde un punto de entrada hasta un punto de salida, evaluando diversas alternativas posibles. Estos laberintos no solo son interesantes desde el punto de vista lúdico, sino también desde el enfoque de la programación, ya que su resolución implica diseñar algoritmos capaces de encontrar soluciones bajo ciertas condiciones. De esta manera, los juegos de laberintos se convierten en una excelente oportunidad para aplicar y reforzar conocimientos de programación básica, al implementar algoritmos que son fundamentales en el desarrollo de sistemas más complejos.

En este proyecto se abordarán las decisiones tomadas por el equipo en cuanto al manejo de la matriz que representa el laberinto, los algoritmos implementados para su validación y resolución, así como la elección de la interfaz gráfica empleada para representar visualmente el recorrido.

Objetivo General

Desarrollar una aplicación capaz de generar, almacenar, validar y resolver laberintos de diferentes tamaños mediante algoritmos automatizados, implementando técnicas de backtracking y garantizando siempre la existencia de un camino válido entre un punto de inicio y un punto de salida, así como su visualización gráfica.

Objetivos Específicos

Diseñar e implementar un algoritmo generador de laberintos que cree matrices de tamaños entre 5x5 y 25x25, asegurando siempre la existencia de un camino válido y accesible entre el punto de inicio y el de salida.

Desarrollar un sistema de almacenamiento que permita guardar y cargar laberintos generados en archivos de texto o formato JSON, facilitando su reutilización y análisis posterior.

Implementar un algoritmo de backtracking que resuelva automáticamente los laberintos generados, permitiendo la visualización clara del recorrido desde el inicio hasta la salida.

Versions

Python

PyQt 5.15.11

Requerimientos funcionales:

* 1. GENERACION DEL LABERINTO O MATRIZ

- Hacer un algoritmo que genere laberintos(matriz) de tamaño 5x5 y 25x25 (en incrementos de 5)

- Usar algoritmo para la creación del laberinto de forma automática, asegurando que el laberinto tenga un camino valido entre el inicio y la salida

- El algoritmo de generación del laberinto tiene que asegurar que el punto de inicio sea accesible dentro del lab y debe tener un camino

* 2. ALMACENAMIENTO DE LABERINTOS

- Implementar una funcionalidad para poder guardar los laberintos generados y para que puedan ser cargados

- Se tienen que guardar en un archivo de texto (o JSON)

* 3. ASEGURAR EL PUNTO DE INICIO DEL LABERINTO SEA VALIDO

- Verificar el punto de inicio del laberinto sea una celda valida y no una pared y además que sea accesible (que tenga camino)

* 4. ASEGURAR QUE EXISTE UN CAMINO VÁLIDO EN ENTRE EL INICIO Y SALIDA

- Garantizar que siempre haya una conexión entre los 2 puntos, una vez generado el laberinto se tiene que verificar que exista camino y si no lo hay se tiene que generar el laberinto de nuevo hasta que se encuentre un camino

5. CREAR ALGORITMO DE BACKTRACKING

* - Implementar un algoritmo de backtracking para resolver el laberinto de manera automática, el algoritmo tiene que explorar todas las posibles rutas desde el inicio hasta la salida, si hay un callejón sin salida se debe retroceder
* 6. HACER VISIBLE LA SOLUCIÓN DEL LABERINTO