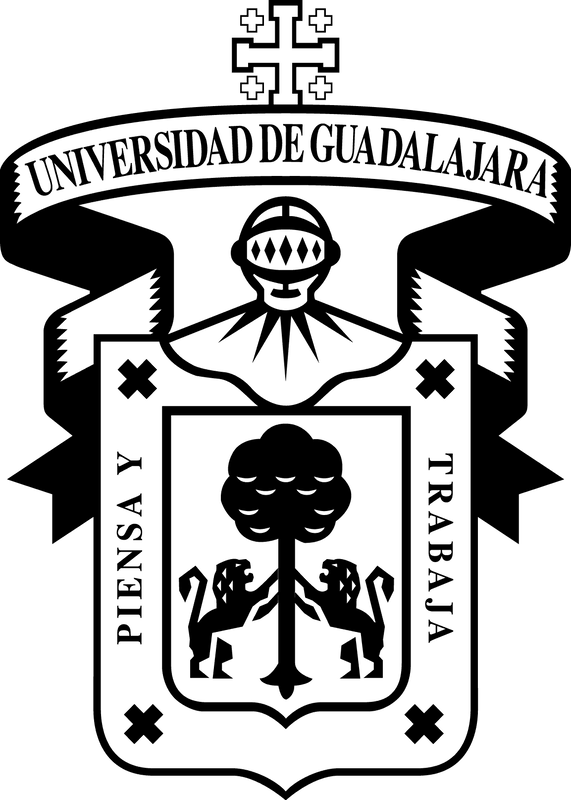
**Laberinto Divide y vencerás**

Omar Alejandro Brizuela Esqueda

Angel Sebastian Garnica Carbajal

Javier Solorzano Razo

Sección D-06  
Análisis de Algoritmia



Introducción

Para este proyecto se estará realizando un laberinto utilizando el método de algoritmos “divide y vencerás” en el cual se tiene que encontrar una solución para el problema de resolver el laberinto.

Divide y vencerás: Divide y vencerás es un paradigma algorítmico similar a los paradigmas de programación Dinámica y Algoritmos ávidos o glotones. Un algoritmo Divide y Vencerás típico resuelve un problema siguiendo estos 3 pasos.

1. Dividir: Descomponer el problema en sub-problemas del mismo tipo. Este paso involucra descomponer el problema original en pequeños sub-problemas. Cada sub-problema debe representar una parte del problema original. Por lo general, este paso emplea un enfoque recursivo para dividir el problema hasta que no es posible crear un sub-problema más.
2. Vencer: Resolver los sub-problemas recursivamente. Este paso recibe un gran conjunto de sub-problemas a ser resueltos. Generalmente a este nivel, los problemas se resuelven por sí solos.
3. Combinar: Combinar las respuestas apropiadamente. Cuando los sub-problemas son resueltos, esta fase los combina recursivamente hasta que estos formulan la solución al problema original. Este enfoque algorítmico trabaja recursivamente y los pasos de conquista y fusión trabajan tan a la par que parece un sólo paso.

Usualmente este método nos permite hacer una reducción bastante significativa en la complejidad tiempo del algoritmo a emplear.

Algunos de los algoritmos mas comunes que utilizan este tipo de paradigma son: Quick Sort, Búsqueda binaria y Merge sort.

Objetivos

Realizar un laberinto con las siguientes especificaciones

Crear matriz cuadrada (6x6)

Definir valores

0 es camino

1 es pared

3 y 4 son teletransportadores

111 es puerta (Trivia o acertijo)

2 es salida

Definir entrada en (0,0)

Se puede avanzar por casillas con valores: 0, 3, 4, 111 y 2

Para avanzar tiene que ser en cruz, no puede volver a una casilla en la que ya se encontraba, solo puede avanzar una casilla a la vez

Si se encuentra con un 0 avanza

Si se encuentra con un 3 se teletransporta a 4 (se define 3 como entrada y 4 como salida para que no se repita en bucle los teletransportes)

Si se encuentra con un 4 se teletransporta a 3 (se define 4 como entrada y 3 como salida para que no se repita en bucle los teletransportes)

Si se encuentra con un 111 se realizará una pregunta o acertijo el cual se tiene que responder correctamente para avanzar

No puede avanzar por casillas con valor 1

Si llega a la casilla con valor 2 se llega al objetivo y se finaliza el laberinto

Para dividir el problema se reducirá la matriz a un espacio de 3x3 para poder validar las casillas para avanzar

Esto realizarlo en Python con interfaz grafica, reporte y presentación.

Desarrollo

En este programa se utilizo la librería de Tkinter para el apartado de la interfaz grafica y el código principal se realizo en el lenguaje de programación de Python.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteEl programa al empezar dará una pequeña bienvenida al usuario indicando si este quiere empezar a jugar directamente o si en su defecto prefiere conocer los créditos del laberinto

En caso de seleccionar la opción de créditos mostrara una pestaña como la siguiente

Interfaz de usuario gráfica, Texto, Sitio web

Descripción generada automáticamente

El laberinto se creo con la base de una matriz cuadrada con dimensiones proporcionadas por el usuario, por regla genera el usuario no puede mostrar una matriz menor a 6x6 y esta estrictamente tiene que ser cuadrada para que empiece el laberinto; el programa estará presentando una serie de alertas en caso de que estas condiciones no se cumplan

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

Imagen que contiene Gráfico

Descripción generada automáticamenteUna vez seleccionadas las medidas las medidas correspondientes se generara un laberinto de forma aleatoria el cual se va a empezar a resolver una vez seleccionada la opción de “Resolver Laberinto”

Interfaz de usuario gráfica

Descripción generada automáticamenteAl seleccionar la opción de “Resolver laberinto” este empezara a buscar el camino hacia la salida, una vez que encuentra la salida mostrar una alerta indicando que el laberinto se ha resuelto correctamente

En el programa se tiene un botón con el que se puede acceder a las reglas que se tienen en el laberinto

Interfaz de usuario gráfica

Descripción generada automáticamente

Reglas del laberinto:

Trivia: Si en el camino al resolverse este se encuentra con una casilla en color morado aparecerá una pequeña trivia que tiene que ser contestada correctamente para poder avanzar

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

En caso de que esta sea contestada de manera incorrecta saltara una alerta la cual mencionara que no se ha contestado correctamente

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

Una vez que se contesta de manera correcta el recorrido vuelve a tomar su curso hasta encontrar la salida

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

Teletransporte:

Cuando el recorrido se encuentra con la casilla en color amarillo se teletransporta a la casilla mas cercana a la salida para con esto resolver el laberinto lo mas rápido posible

Código QR

Descripción generada automáticamente

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

Código

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Pantalla de computadora

Descripción generada automáticamente con confianza media

Conclusiones

Brizuela Esqueda Omar Alejandro: En esta actividad me toco ser el administrador del proyecto, al principio se me complico bastante todo el tema de como llevar el proyecto a cabo debido a que no contaba con la experiencia en este tipo de roles de trabajo, con el paso del tiempo en el desarrollo de la actividad y con el apoyo del profesor logré entender mas las funciones de un desarrollador, al menos en el ámbito de proyectos tecnológicos, con respecto a la forma de trabajo estuve cómodo debido a que mis compañeros de trabajo cumplieron con sus actividades en tiempo y forma por lo que no hubieron retrasos o fallas con el desarrollo del proyecto. Para finalizar este proyecto fue muy interesante desde el hecho de realizar un tipo “juego” con el paradigma de divide y vencerás el cual puede ser bastante útil cuando no se utilizan muchos valores, con esta actividad también aprendí mas sobre el uso de github, los commits, realizar organigramas y otras herramientas bastante útiles a la hora de realizar proyectos de este ámbito.

Garnica Carbajal Ángel Sebastián: En esta actividad tuvimos que cambiar de roles a los que estábamos anteriormente, en este caso me toco hacer el Frontend, se me hizo un tanto complicado ya que aún no termino de familiarizarme con Tkinter, además de que no soy muy bueno en el aspecto de diseñar, también creamos commits y los fuimos actualizando y esto me parece un recurso bastante útil para trabajar en equipo y mantener un control sobre el código. Conocimos y aplicamos un algoritmo con el paradigma "Divide y vencerás", en esta actividad del laberinto justamente se implementa este algoritmo, en ciertos casos resulta bastante optimo implementar estos, pero no en todos los problemas ya que se vuelve ineficiente ante ciertos problemas, por ejemplo, cuando el tamaño de la entrada del arreglo es grande, estos algoritmos pueden tardar mucho tiempo en ejecutarse.

Solorzano Razo Javier: Una conclusión sobre este algoritmo que hicimos sobre divide y vencerás para resolver un laberinto es que, se aplica bien cuando se trata para laberintos grandes o complejos. Y yo que trabaje en el back end, fue algo difícil hacer el código, pero se logró poder aplicar el divide y vencerás, en donde lo que se hizo es dividir el laberinto en secciones más pequeñas para encontrar el camino en cada sección para llegar a la salida, fue algo complicado, pero se logró.

Bibliografía

*Algoritmos de divide y vencerás (artículo)*. (s/f). Khan Academy. Recuperado el 11 de marzo de 2024, de <https://es.khanacademy.org/computing/computer-science/algorithms/merge-sort/a/divide-and-conquer-algorithms>

*Davidcabreraygarcia. (2023, 12 abril). La recursividad y el algoritmo de «divide y vencerás». Medium.* <https://medium.com/@davidcabreraygarcia/la-recursividad-y-el-algoritmo-de-divide-y-vencer%C3%A1s-9418325e55b5>