

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERÍA

MATEMÁTICA PARA COMPUTACIÓN

CATEDRÁTICO: ING. JOSÉ ALFREDO GONZÁLEZ DÍAZ

TUTOR ACADÉMICO: BRAYAN MEJÍA



MANUAL DE **USUARIO**

KEVIN EMMANUEL SALAZAR MONTERROSO

CARNÉ: 202300669

SECCIÓN: A

GUATEMALA, 25 DE OCTUBRE DEL 2,024

ÍNDICE

OBJETIVOS DEL SISTEMA.....	2
GENERAL	2
ESPECÍFICOS.....	2
INTRODUCCIÓN	2
INFORMACIÓN DEL SISTEMA	3
REQUISITOS DEL SISTEMA.....	4
FLUJO DE LAS FUNCIONALIDADES DEL SISTEMA.....	6

OBJETIVOS DEL SISTEMA

GENERAL

El objetivo de este programa es implementar una interfaz gráfica en Python con Tkinter que permita a los usuarios crear, visualizar y manipular grafos no dirigidos con vértices de peso predefinido. Además, el programa permite aplicar y visualizar los algoritmos de búsqueda en anchura (BFS) y profundidad (DFS) sobre el grafo ingresado, mostrando sus resultados gráficos de forma interactiva.

ESPECÍFICOS

- Implementar una interfaz gráfica en Python, mediante Tkinter, que permita al usuario crear grafos dinámicamente mediante la adición de vértices y aristas, facilitando la visualización y manipulación del grafo.
- Desarrollar y aplicar algoritmos de búsqueda en grafos, como BFS y DFS, que recorran los vértices en orden ascendente de peso, proporcionando resultados visuales y detallados en la interfaz.
- Generar representaciones gráficas claras y comprensibles de los grafos creados y de los recorridos BFS y DFS, utilizando Graphviz para mejorar la comprensión y el análisis de los algoritmos de búsqueda implementados.

INTRODUCCIÓN

El propósito de este manual es guiar al usuario en el uso y entendimiento de una aplicación desarrollada en Python para la creación, visualización y manipulación de grafos, así como para la aplicación de algoritmos de búsqueda como BFS (Breadth-First Search) y DFS (Depth-First Search). Este documento describe paso a paso cada función del programa, detalla sus requerimientos de sistema y

proporciona una visión técnica de su funcionamiento. La meta es facilitar el acceso a todas las funcionalidades y asegurar que el usuario pueda operar el software de manera autónoma y eficiente.

La aplicación desarrollada permite a los usuarios construir grafos mediante la adición de vértices y aristas, que son representados visualmente en una interfaz gráfica intuitiva. Al construir el grafo, el usuario puede aplicar algoritmos de búsqueda que se ejecutan en función del orden de peso asignado a los vértices. Estos recorridos se muestran de forma gráfica y detallada en la misma interfaz, proporcionando una herramienta interactiva y educativa para explorar estructuras y algoritmos de grafos.

INFORMACIÓN DEL SISTEMA

El programa permite al usuario crear y manipular grafos mediante una interfaz gráfica amigable desarrollada con Tkinter. Al iniciar la aplicación, el usuario tiene a su disposición un área de dibujo para visualizar el grafo y secciones con controles específicos: campos de texto y botones para agregar vértices y aristas, y apartados que listan los vértices y aristas agregados. Los vértices son representados por letras en orden alfabético, con un peso fijo que va de menor a mayor desde 'a' hasta 'z'.

Al añadir vértices y aristas, el grafo se dibuja automáticamente en el área de visualización, proporcionando una representación visual clara y actualizada en tiempo real. Los usuarios pueden ejecutar recorridos en anchura (BFS) y profundidad (DFS) haciendo clic en los botones correspondientes. Los algoritmos de búsqueda procesan los vértices de acuerdo con su peso, mostrando el orden de los nodos visitados en el área de visualización y resaltándolos.

Por último, el programa emplea Graphviz para representar los grafos y los resultados de los recorridos de búsqueda. Estos resultados pueden exportarse a un archivo o visualizarse directamente en la interfaz, lo cual facilita al usuario la exploración y análisis de las estructuras de grafos y de los algoritmos de búsqueda implementados.

REQUISITOS DEL SISTEMA

Para ejecutar esta aplicación de manera óptima, el usuario debe contar con los siguientes requisitos en su sistema:

1. Sistema Operativo: Se recomienda Windows 10 o superior, aunque también puede ejecutarse en sistemas macOS o distribuciones de Linux compatibles con Python y Tkinter.
2. Python: Versión 3.8 o superior. Python es esencial para ejecutar la interfaz gráfica desarrollada con Tkinter y realizar los procesos de back-end en el programa. Puede descargarse e instalarse desde [python.org](https://www.python.org/).
3. Tkinter: La biblioteca Tkinter suele instalarse automáticamente con Python, pero en algunas configuraciones, puede requerir instalación manual. Para verificar su instalación, se puede ejecutar ``import tkinter`` en la consola de Python.
4. Graphviz: La herramienta Graphviz es necesaria para generar y visualizar los grafos creados. El usuario debe descargar e instalar Graphviz desde graphviz.org, asegurándose de agregar su ruta al PATH del sistema para facilitar su integración.

5. Editor de Texto o IDE: Si el usuario desea revisar o modificar el código fuente, es recomendable utilizar un editor de texto como Visual Studio Code, PyCharm o cualquier otro IDE compatible con Python.

Una vez cumplidos estos requisitos, el usuario podrá ejecutar la aplicación desde la terminal de Python o directamente desde el IDE elegido para explorar todas sus funcionalidades.

FLUJO DE LAS FUNCIONALIDADES DEL SISTEMA

Para ejecutar y comprender cada funcionalidad del programa, se describen a continuación los pasos desde su apertura hasta el cierre, junto con las acciones que debe realizar el usuario en cada etapa. A continuación, se incluye una guía visual y funcional paso a paso.

Paso 1: Ejecutar el Programa

1. **Abrir el archivo principal del programa:** Asegúrese de tener el archivo principal (en este caso, main.py o el archivo ejecutable del programa) en su directorio de trabajo.
 2. **Ejecutar desde la terminal:**
 - Abra una terminal de comandos en el directorio donde se encuentra el archivo.
 - Ejecute el programa con el comando:

```
python graphiz.py
```
 3. Esto abrirá la interfaz gráfica en Python utilizando Tkinter y cargará las funcionalidades de creación de grafos y algoritmos de búsqueda.
-

Paso 2: Interfaz de Opciones Principales

- Al iniciar el programa, la interfaz mostrará un menú con opciones básicas como:
 - **Agregar Vértice:** Permite al usuario ingresar un nuevo vértice.
 - **Agregar Arista:** Añade una arista entre dos vértices existentes.
 - **Generar Grafo:** Genera un grafo visual en el área de dibujo.
 - **Recorridos BFS y DFS:** Permite al usuario realizar búsquedas de anchura (BFS) y profundidad (DFS).

Cada opción está acompañada de un botón en la interfaz, facilitando la navegación del usuario.

Paso 3: Agregar Vértice

- **Botón "Agregar Vértice":** Al hacer clic, se abre un cuadro de diálogo que permite al usuario ingresar el nombre de un vértice.
- **Visualización del Vértice:** Una vez añadido, el vértice aparecerá en la lista de vértices de la interfaz, y se mostrará en el área de dibujo.

Nota: Los vértices se generan con pesos predefinidos (según su posición en el abecedario, de 'a' a 'z').

Paso 4: Agregar Arista

- **Botón "Agregar Arista":** Solicita al usuario especificar los vértices de inicio y fin para crear la arista.
- **Visualización de la Arista:** La arista se mostrará como una línea conectando los vértices en el área de dibujo, y la lista de aristas actualizadas se muestra en la interfaz.

Nota: Las aristas no incluyen peso y son dirigidas de acuerdo con el orden de ingreso.

Paso 5: Generar Grafo Visual

- **Botón "Generar Grafo":** Una vez que los vértices y aristas están definidos, el usuario puede generar el grafo visual.
 - **Visualización del Grafo:** El grafo se renderiza en el área de dibujo de la interfaz, mostrando los vértices y conexiones.
-

Paso 6: Realizar Recorridos (BFS y DFS)

- **Botones "Recorrido BFS" y "Recorrido DFS":** Al seleccionar cualquiera de estas opciones, se generará el recorrido según el algoritmo seleccionado.
- **Ejecución del Recorrido:**

- **BFS (d)**: Este recorrido explora los vértices en anchura y muestra el camino seguido, comenzando desde el vértice de menor peso.
- **DFS (p)**: Realiza el recorrido en profundidad, mostrando los vértices visitados en este orden, también comenzando desde el vértice de menor peso.

El resultado de cada recorrido se muestra en el área de salida de la interfaz, junto con una representación visual que resalta los nodos visitados.

Paso 7: Cerrar el Programa

- **Opción "Salir"**: Para finalizar, el usuario selecciona la opción "Salir" en el menú principal.
- **Confirmación de Cierre**: Al cerrar, la interfaz muestra un mensaje de confirmación, indicando que el programa ha finalizado con éxito.
- **Salida en la Terminal**: La terminal confirmará el cierre y el fin de la ejecución del programa.

Este flujo garantiza que el usuario pueda comprender y utilizar las funcionalidades clave de creación, visualización y recorrido de grafos.

