MANUAL TECNICO

librerías necesarias para el funcionamiento del proyecto.

1. **networkx (nx):** Utilizado para la creación y manipulación de estructuras de grafos complejas. Permite realizar operaciones comunes de grafos como añadir nodos, aristas, y realizar algoritmos de grafos.
2. **customtkinter (ctk):** Biblioteca que extiende las capacidades de Tkinter con un estilo visual mejorado y widgets adicionales.
3. **matplotlib:** Usada para visualizar datos en forma gráfica dentro de Python. En este contexto, se utiliza para dibujar los grafos y visualizar las estructuras de datos.

```python
# Importa las bibliotecas necesarias para trabajar con grafos, interfaces de usuario personalizadas y gráficos.
import networkx as nx
import customtkinter as ctk
from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg
from matplotlib.figure import Figure
import matplotlib.pyplot as plt
```

Creación del grafo

G: Es un objeto de grafo de NetworkX que se inicia vacío y puede ser modificado dinámicamente añadiendo nodos y aristas a través de la interfaz gráfica.

```python
# Crea un grafo vacío usando NetworkX.
G = nx.Graph()
```

Configuración de la ventana principal

root: Es la ventana principal de la aplicación creada con ctk.CTk(), donde se configura el título y el comportamiento de ajuste automático de tamaño.

main_frame: Marco principal dentro de root que organiza visualmente el espacio de la aplicación y contiene otros widgets.

```python
# Crea la ventana principal de la aplicación usando CustomTkinter.
root = ctk.CTk()
root.title("Algoritmo de busqueda en Anchura")  # Establece el título de la ventana.
```

Creación de widgets

Etiquetas y entradas para vértices y aristas: Permiten al usuario introducir y enviar datos al grafo.

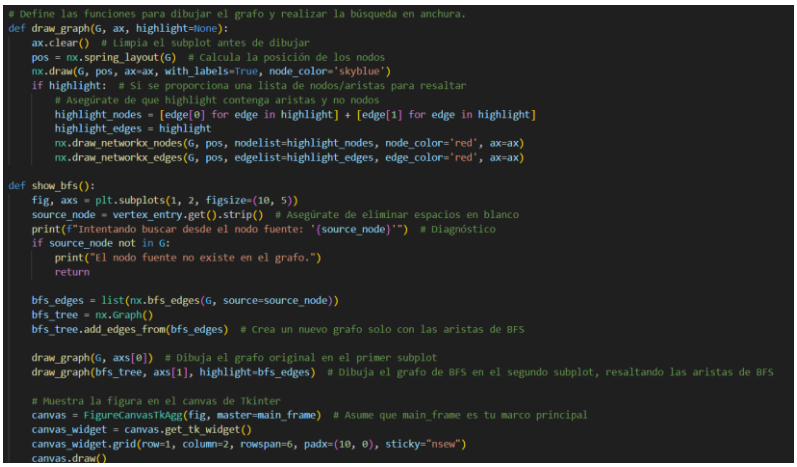
Botones: Incluyen funcionalidades para agregar vértices y aristas, y botones para operaciones como dibujar el grafo y ejecutar algoritmos de búsqueda.

print_info_button: Botón que imprime información básica del grafo en la consola, como el número de nodos y aristas.

Funciones para manipulación y visualización del grafo

draw_graph(G, ax, highlight=None): Esta función dibuja el grafo en un subplot de matplotlib. Utiliza la posición calculada por nx.spring_layout para el acomodo visual de los nodos. Los nodos o aristas pasados al parámetro highlight se resaltan en color rojo.

show_bfs() y show_dfs(): Estas funciones ejecutan los algoritmos de Búsqueda en Anchura y Profundidad desde un nodo fuente. Después de ejecutar el algoritmo correspondiente, se crea y se muestra un grafo que representa el árbol de búsqueda con las aristas del árbol resaltadas. Los grafos resultantes se visualizan en paneles adyacentes para comparar con el grafo original.



Configuración de Matplotlib

Se prepara un FigureCanvasTkAgg para mostrar gráficos de matplotlib dentro de la interfaz de Tkinter.

Ejecución de la aplicación

Se configura el mainloop de Tkinter, que es esencial para que la interfaz de usuario permanezca activa y responda a eventos del usuario.

```python
# Ahora que show_bfs está definida, crea los botones que la utilizan.
draw_button = ctk.CTkButton(main_frame, text="Dibujar grafo", command=show_bfs)
draw_button.grid(row=7, column=0, padx=(0,5), sticky="ew")

bfs_button = ctk.CTkButton(main_frame, text="Busqueda en anchura", command=lambda: show_bfs())
bfs_button.grid(row=7, column=1, padx=(5,0), sticky="ew")

dfs_button = ctk.CTkButton(main_frame, text="Búsqueda en profundidad", command=show_dfs)
dfs_button.grid(row=8, column=0, columnspan=2, sticky="ew", pady=(5,0))

# Configura Matplotlib para la visualización del grafo.
figure = Figure(figsize=(5, 5))
ax = figure.add_subplot(111)
canvas = FigureCanvasTkAgg(figure, main_frame)
canvas_widget = canvas.get_tk_widget()
canvas_widget.grid(row=1, column=2, rowspan=6, padx=(10, 0), sticky="nsew")

# Configura la ventana principal para ajustarse automáticamente al tamaño de sus contenidos.
root.grid_rowconfigure(1, weight=1)
root.grid_columnconfigure(2, weight=1)

# Inicia el bucle principal de la aplicación Tkinter.
root.mainloop()
```

DIAGRAMA DE FLUJO