

## **Documento LAB2 EDD2**

### **Integrantes:**

Santy Baza

Darlen Arias

Daniel Pedroza

### **Documentación a detalle:**

Programa de Grafos de Aeropuertos

#### Estructuras de Datos

- VUELO: Registro que contiene información de un vuelo (códigos de aeropuertos, nombres, ciudades, países, coordenadas)
- GRAFO: Lista de listas de tuplas (índice\_destino, distancia)
- INFO\_AEROPUERTO: Diccionario con detalles del aeropuerto (código, nombre, ciudad, país, latitud, longitud)

#### Constantes

- RADIO\_TIERRA\_KM = 6371
- RUTA\_ARCHIVO\_CSV = 'ruta/al/archivo/flights\_final.csv'
- RUTA\_SALIDA\_MAPA = 'mapa\_aeropuertos.png'

#### Funciones Principales

distancia\_haversine(coord1, coord2)

Entrada: coord1, coord2 (tuplas de latitud y longitud)

Salida: distancia en kilómetros

Convertir coordenadas a radianes

Calcular diferencias de latitud y longitud

Aplicar fórmula de Haversine

Retornar distancia \* RADIO\_TIERRA\_KM

cargar\_datos\_aeropuertos(ruta\_archivo)

Entrada: ruta\_archivo (string)

Salida: codigos\_aeropuertos, coordenadas\_aeropuertos, grafo,  
info\_aeropuertos

Inicializar listas y diccionarios vacíos

Abrir archivo CSV

Para cada fila en CSV:

Extraer información de aeropuertos origen y destino

Si aeropuerto no existe en listas:

Agregar a codigos\_aeropuertos, coordenadas\_aeropuertos,  
info\_aeropuertos

Agregar nueva lista vacía a grafo

Calcular distancia entre aeropuertos

Agregar conexión al grafo (en ambas direcciones)

Retornar estructuras de datos

dijkstra(grafo, inicio)

Entrada: grafo, índice de vértice inicial

Salida: distancias, predecesores

Inicializar distancias a infinito y predecesores a -1

Establecer distancia del nodo inicial a 0

Crear cola de prioridad con nodo inicial

Mientras la cola no esté vacía:

    Extraer nodo con menor distancia

    Para cada vecino del nodo actual:

        Calcular nueva distancia

        Si nueva distancia es menor que la almacenada:

            Actualizar distancia y predecesor

            Agregar vecino a la cola de prioridad

Retornar distancias y predecesores

mostrar\_aeropuertos\_lejanos\_misma\_componente(codigo, ...)

Entrada: código de aeropuerto, estructuras de datos

Salida: Ninguna (imprime información)

Verificar si el aeropuerto existe

Mostrar información del aeropuerto de inicio

Encontrar componente conexas del aeropuerto

Calcular distancias mínimas desde el aeropuerto

Ordenar aeropuertos de la misma componente por distancia

Mostrar los 10 aeropuertos más lejanos con detalles

`mostrar_camino_minimo(codigo_inicio, codigo_fin, ...)`

Entrada: códigos de aeropuertos inicio y fin, estructuras de datos

Salida: Ninguna (imprime información y genera visualización)

Verificar si los aeropuertos existen

Aplicar Dijkstra desde aeropuerto inicial

Si no hay camino, informar y terminar

Reconstruir camino usando predecesores

Mostrar detalles de cada tramo del camino

Generar visualización del camino

`analizar_conectividad(grafo)`

Entrada: grafo

Salida: lista de componentes conexas

Inicializar lista de visitados

Para cada vértice no visitado:

Realizar búsqueda en profundidad

Agregar componente encontrada a la lista

Retornar lista de componentes

`arbol_expansion_minima_prim(grafo, inicio)`

Entrada: grafo, vértice inicial

Salida: peso total del árbol de expansión mínima

Inicializar estructuras para Prim

Mientras no se hayan visitado todos los vértices:

    Seleccionar arista de menor peso a vértice no visitado

    Agregar peso al total

    Marcar nuevo vértice como visitado

    Actualizar lista de candidatos

Retornar peso total

`generar_mapa_aeropuertos(codigos_aeropuertos,  
coordenadas_aeropuertos)`

Entrada: lista de códigos y coordenadas de aeropuertos

Salida: Ninguna (genera archivo de imagen)

Configurar figura de matplotlib

Para cada aeropuerto:

    Dibujar punto en el mapa

    Agregar etiqueta con código

Configurar título, etiquetas y rejilla

Ajustar límites del mapa

Guardar mapa como archivo PNG

main()

Cargar datos de aeropuertos

Mientras verdadero:

Mostrar menú de opciones

Leer opción del usuario

Según la opción seleccionada:

1: Mostrar aeropuertos lejanos

2: Mostrar camino mínimo

3: Analizar conectividad

4: Calcular MST para componentes

5: Generar mapa

6: Salir del programa

Si opción es salir, terminar bucle