

**Nombre:** Jeremy Andrey Rojas Cerna

**Carné:** B96804

**Curso:** CI-0116 Análisis de Algoritmos y Estructuras de Datos

## 1. Planteamiento del problema

El Sistema Nacional de Tipificación de Emergencias Potenciales (SiNTiEmPo) requiere analizar la conectividad entre sus centros regionales para planificar de manera eficiente la distribución de recursos y atención a emergencias. Se desea construir un grafo a partir de un archivo CSV que contiene ciudades y tiempos de viaje entre ellas, y luego usar este grafo para responder preguntas relevantes para la logística del sistema.

## 2. Descripción y justificación de la solución

Para resolver el problema, se representó el sistema como un grafo dirigido y ponderado, donde los vértices son las ciudades y las aristas representan rutas con un tiempo de viaje. Se utilizó el algoritmo Floyd-Warshall para calcular las distancias mínimas entre todos los pares de nodos, debido a que es el más eficiente para grafos densos y cuando se requiere información entre todos los pares. Además, este algoritmo se ejecuta una única vez al cargar los datos, mientras que otros algoritmos, como Dijkstra, se ejecutarían cada vez que sea necesario calcular una distancia de un nodo a otro. Se eligió esta solución porque garantiza una matriz completa de distancias y padres, lo que facilita responder todas las preguntas propuestas.

El código fue organizado en módulos separados para:

1. Leer los archivos CSV y construir la matriz de adyacencia.
2. Ejecutar Floyd-Warshall y construir la matriz de padres.
3. Interactuar con el usuario mediante funciones de consulta.

## 3. Resultados

A continuación, se resumen los resultados obtenidos al ejecutar el programa con los archivos input\_small.csv, input\_medium.csv, e input\_large.csv. Los detalles completos están en el archivo anexo.

### 3.1 Resultados para input\_small.csv

	Origen	Destino/Resultado	Peso	Ruta
Buscar hub		San Juan de Dios	15	
Mejor ciudad para enviar ayuda	Zapotal	Mata Redonda	13	Mata Redonda → Zapotal
Ciudades más distantes	Parrita	Zapotal	42	Parrita → Quebradilla → Puerto Carrillo →

				San Juan de Dios →
				Mata Redonda →
				Zapotál
<b>Ciudades más cercanas</b>	San Juan de Dios	Quebradilla	2	San Juan de Dios →
	San Juan de Dios	Mata Redonda	2	Quebradilla
				San Juan de Dios →
				Mata Redonda
<b>Lista de ciudades según tiempo promedio</b>			9.6666	San Juan de Dios
			11.3333	Mata Redonda
			11.6666	Puerto Carrillo
			12	Los Guido
			14.2222	Venado
			15.3333	Zapotál
			15.4444	Jaris
			18.5555	Valle La Estrella
			21.5555	Quebradilla
			25.3333	Parrita

## 3.2 Resultados para input\_medium.csv e input\_large.csv

Para ver los resultados de estos dos archivos, ver los archivos externos en el directorio de este documento.

## 4. Instrucciones de compilación y ejecución

Para facilidad del usuario, se provee un Makefile para compilar y ejecutar el programa. Los comandos en este Makefile son los siguientes:

- make: compila el programa en caso de ser necesario.
- make run: compila (si es necesario) y ejecuta el programa.
- make clean: elimina archivos binarios generados durante la compilación.

Además de poderse ejecutar con el comando ‘make run’, el programa también puede ejecutarse directamente desde la terminal. El binario se encuentra en el directorio ‘bin’, generado durante la compilación. Desde ‘ci0116-25a-b96804/tarea-03’, el comando de ejecución es ‘bin/executable’.

Es necesario colocar los archivos CSV dentro del directorio ‘input’. Además, es necesario asegurarse de que un CSV agregado siga el formato proveído por el encabezado de los archivos de ejemplo:

SourceId,SourceName,TargetId,TargetName,Weight