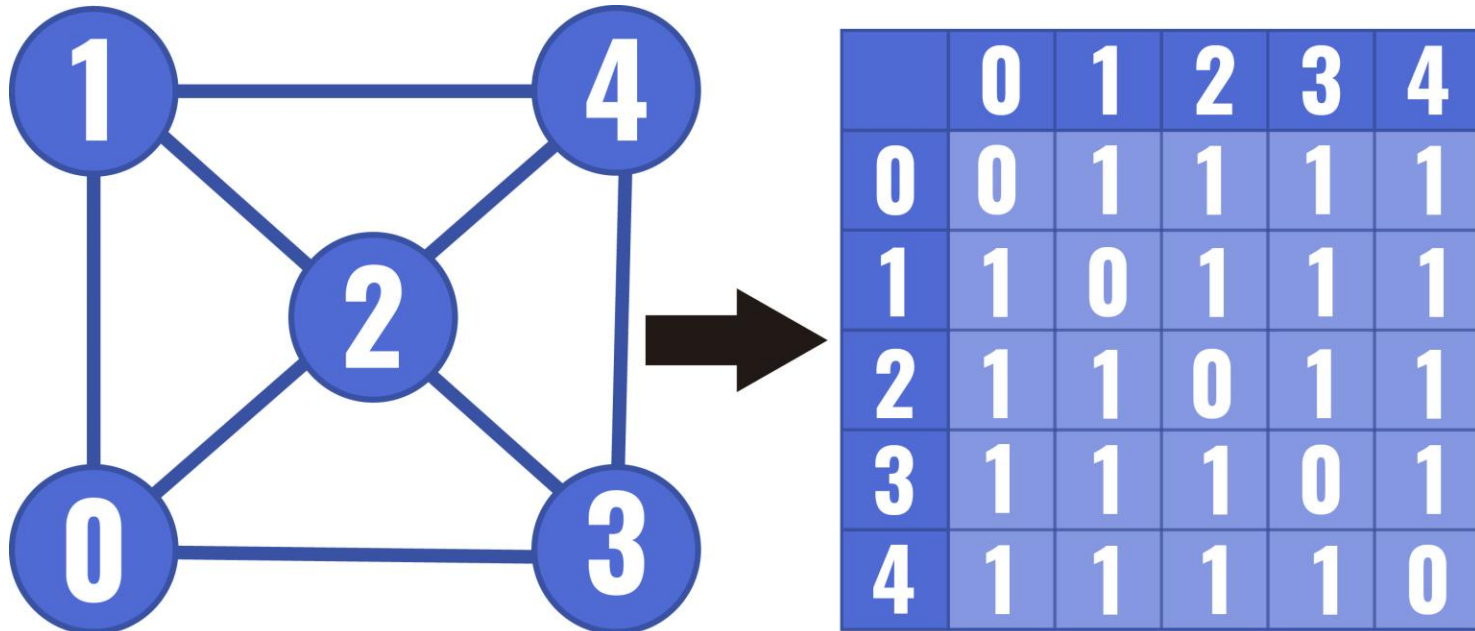


# ***Ruteo de vehículos eléctricos en una empresa repartidora de mercancía***

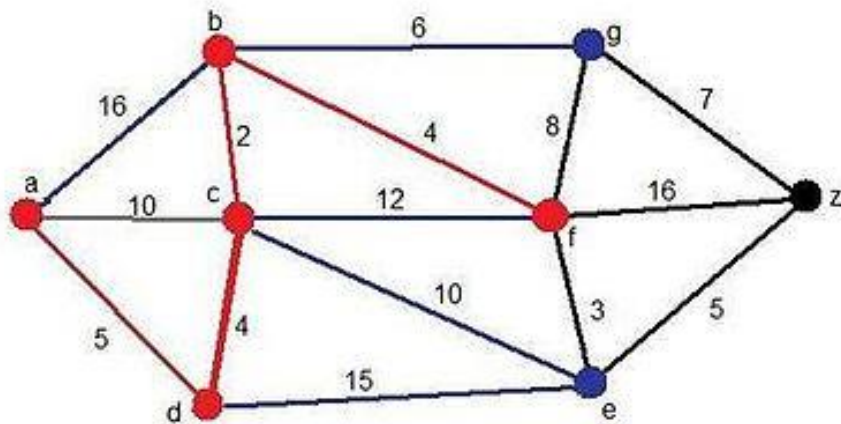
***Julian Gomez Benitez  
Juan Pablo Rincon Usma  
Medellín, 28/03/2021***

## *Estructuras de Datos Diseñada*



**Gráfico 1:** Representacion con una matriz de adyacencia de un grafo.

# Explicación del algoritmo y su complejidad



**Gráfico 2:** Algoritmo del vecino más cercano para hallar una solución óptima

Sub problema	Complejidad
Crear matriz de adyacencia	$O(N/2)$
Solucion Greedy	$O(M^2)$
Optimizacion Tabu Search	$O(I*N*V^2)$
Complejidad Total	$O(N/2 + M^2 + I*N*V^2)$

**Tabla 1:** Complejidad del algoritmo dividido en sub-problemas

# ***Criterios de Diseño del Algoritmo***

En nuestro programa decidimos usar dos algoritmos que se complementan. El primero es un algoritmo tipo Greedy (también conocido como el algoritmo del vecino más cercano) que se guía por una heurística que consiste en elegir la opción local óptima con la esperanza de llegar a una solución general lo más óptima posible, la razón por la que decidimos usar un Greedy para la solución del problema

es más que todo por la rapidez con la que devuelve una solución, aunque la mayoría de las veces esas soluciones no son las más eficientes esto no es un problema porque estas soluciones nos sirven de base para el segundo algoritmo que implementamos en nuestro programa.

La optimización matemática Tabu Search, es un algoritmo metaheurístico que toma como punto de inicio una solución, en este caso toma la solución tipo Greedy que se realizó anteriormente, y genera nuevas soluciones a partir de esta. Para generar las nuevas soluciones, el orden en el que dos nodos son visitados es intercambiado y se usa la distancia total de la ruta para juzgar cual de todas es la más óptima. Esta búsqueda se repite un número definido de veces. Decidimos usar este algoritmo debido a que se complementa muy bien con la solución Greedy que planteamos, ya que mientras el Greedy saca una posible solución lo más rápido posible el Tabu Search la optimiza para asegurarse de que sea mucho más eficaz, generando así un algoritmo que es eficaz y a la vez preciso.

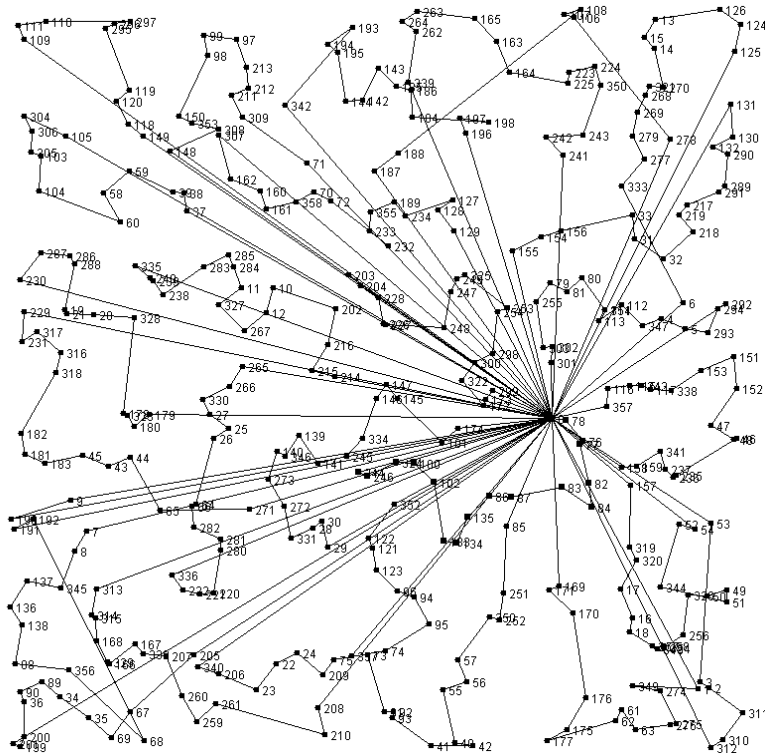
# Consumo de Tiempo y Memoria

	<i>Conjunto de Datos 1 (5 nodos)</i>	<i>Conjunto de Datos 2(345 nodos)</i>	<i>Conjunto de Datos 3(359 nodos)</i>
<i>Mejor caso</i>	222 ms	1193 ms	1196 ms
<i>Caso promedio</i>	227.8 ms	1333 ms	1313 ms
<i>Peor caso</i>	239 ms	1369 ms	1482 ms

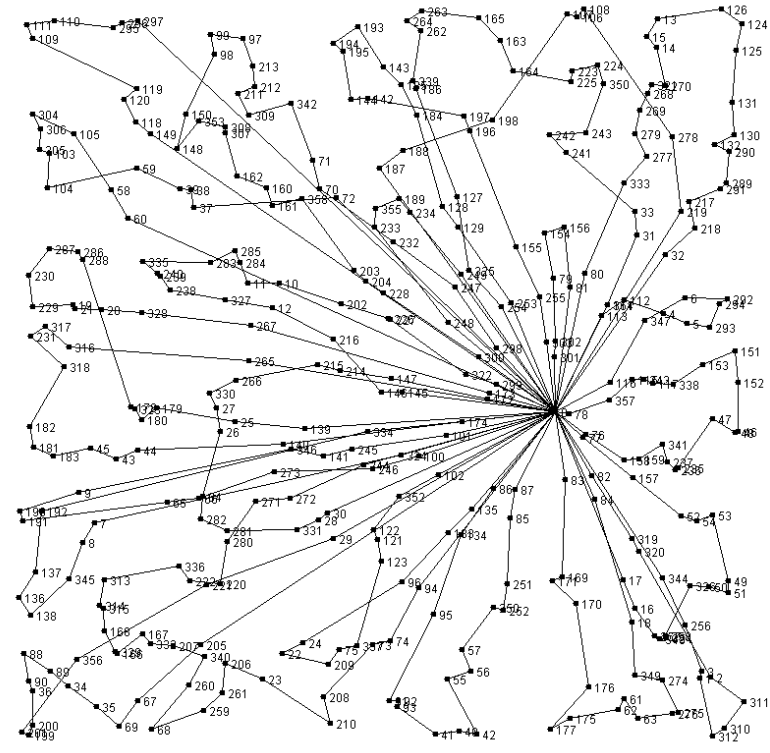
**Gráfico 3:** consumo de tiempo en diferentes datasets

# Software en funcionamiento

VRP solution for 358 customers with Cost: 4575.92



VRP solution for 358 customers with Cost: 3828.7000000000007



**Gráfico 4:** El gráfico de la izquierda representa la solución greedy mientras el de la derecha representa la optimización tabu search