

Optimización de entregas con vehículos eléctricos

Jacobo Rave Londoño

Diego Alejandro Vanegas González

Medellín, Fecha del día de la sustentación

Estructuras de Datos Diseñada

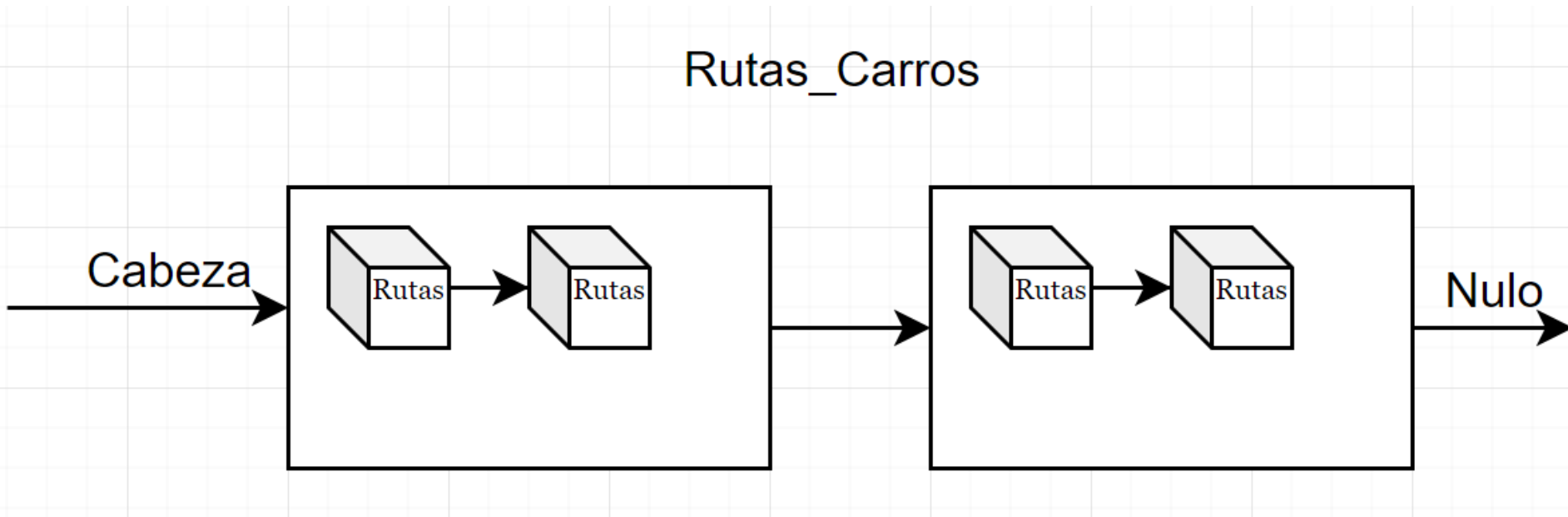


Gráfico 1: Lista de Listas. Una lista, contiene las diferentes rutas para cada carro

Estructuras de Datos Diseñada

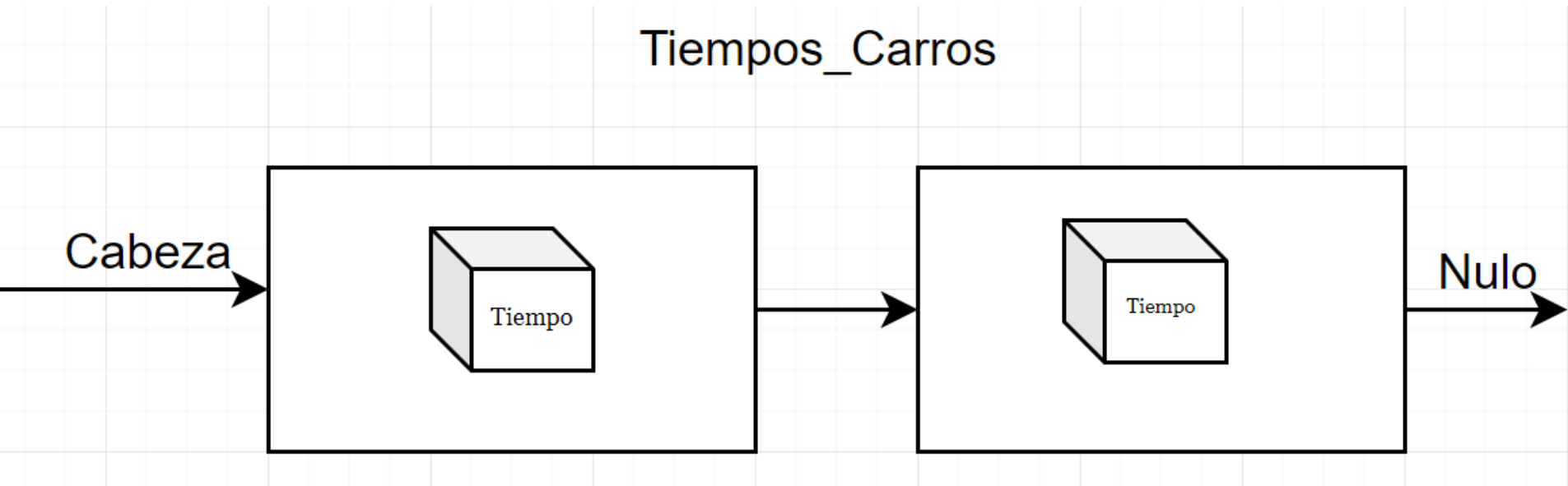


Gráfico 2: Lista de Listas. Una lista, contiene el tiempo que demora cada Carro en sus respectivas rutas

Estructuras de Datos Diseñada

	Deposito	c0	c1
Deposito	null	{D, T}	{D, T}
c0	{D, T}	null	{D, T}
c1	{D, T}	{D, T}	null

Gráfico 3: Matriz de parejas. Una matriz, la cual contiene parejas de Distancia/Tiempo entre los diferentes nodos

Estructuras de Datos Diseñada

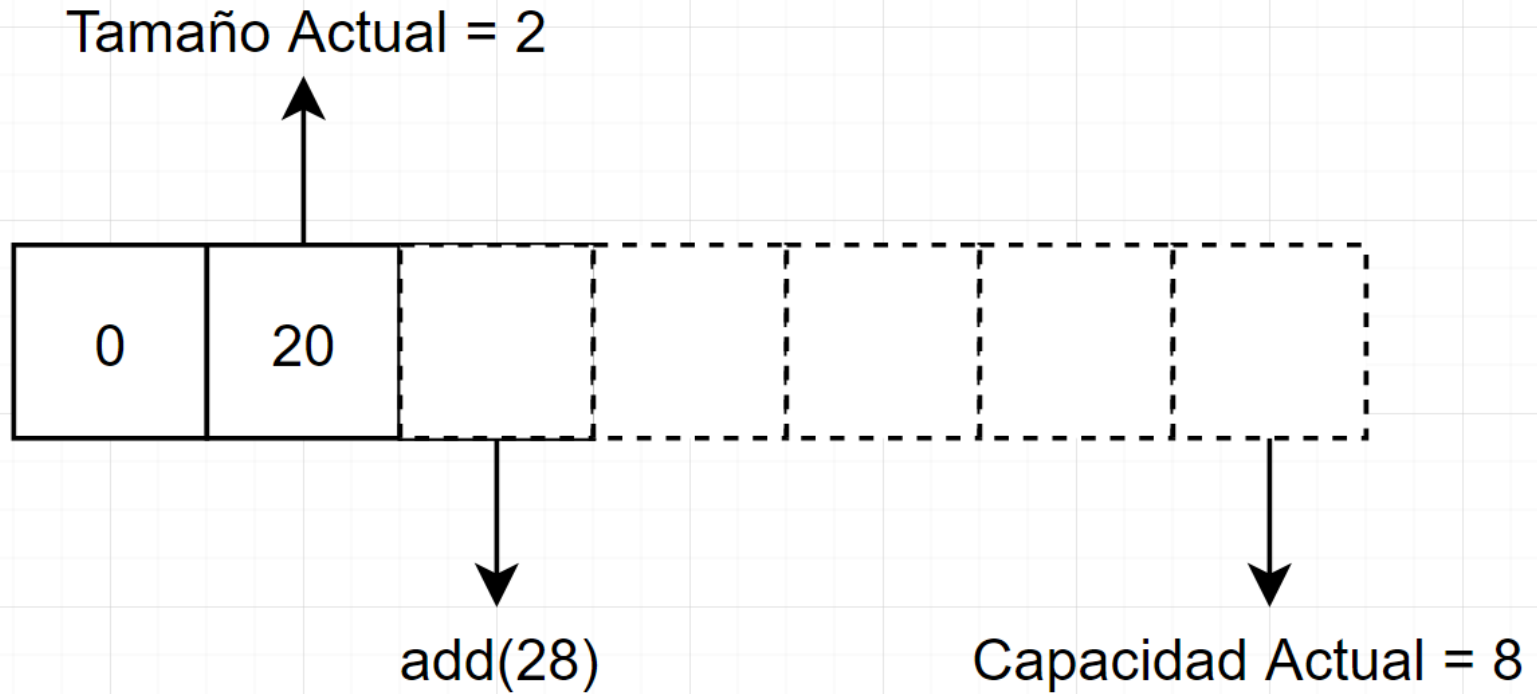


Gráfico 4: Array List. Una lista de arreglos para guardar la “ruta” más optima para todo el recorrido

Explicación del algoritmo y su complejidad

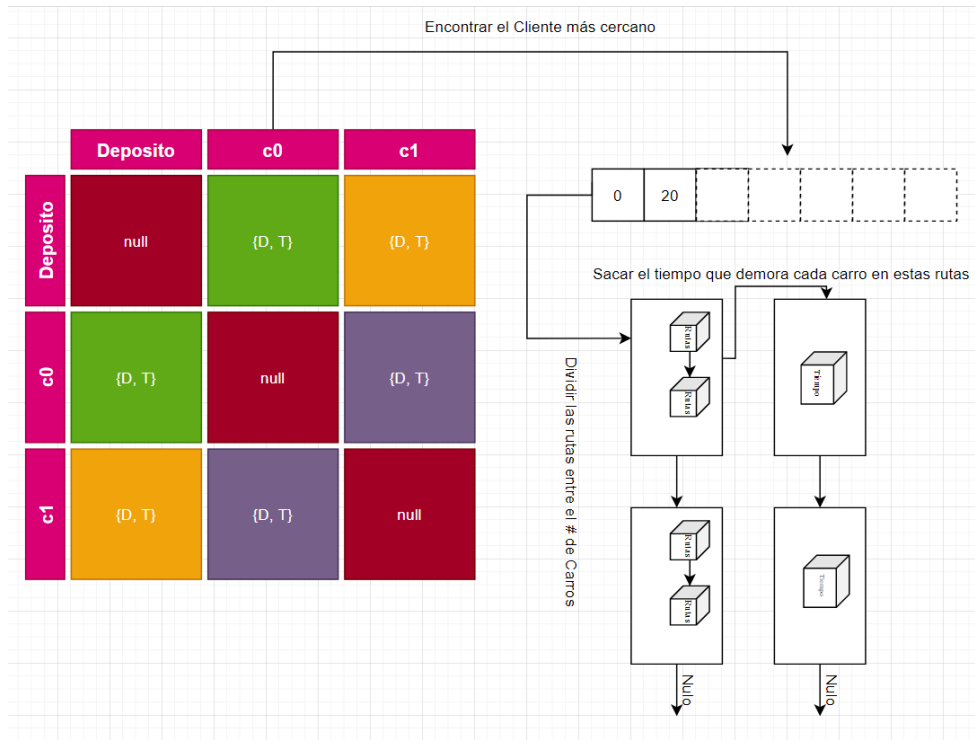


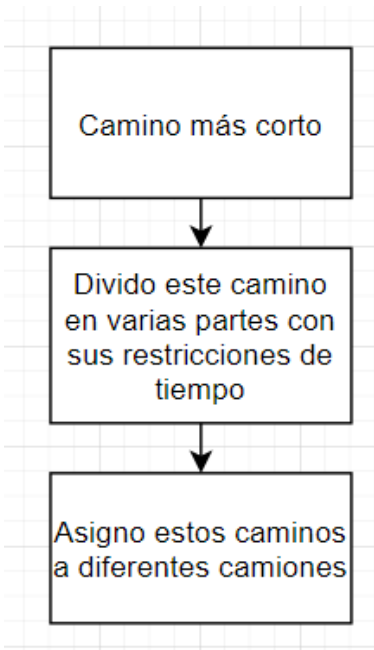
Gráfico 5: De la matriz, encuentra el mejor cliente y crea una ruta general, de ahí divide esa ruta dependiendo del número de carros y calcula el tiempo que demorará cada carro en su ruta

Método 	Complejidad 
Leer	$O(N^2)$
asignarCor	$O(N)$
generarGrafoDT	$O(N * M)$
calcularRuta	$O(N * M)$
calcularTiempo	$O(N)$
generarTiempoTotal	$O(N)$
generarCarros	$O(1)$
bestClient	$O(N)$
asignarRutas	$O(N^3)$

Tabla 1: Complejidad del algoritmo para los métodos

Criterios de Diseño del Algoritmo

Después de analizar diferentes soluciones al problema, nosotros concluimos que una solución basada en un algoritmo, el cuál se centra en encontrar el camino más corto y distribuye este camino entre una cantidad especifica de carros.



De esta forma, logramos hallar un camino optimo y lo distribuimos cumpliendo las restricciones en tiempo

Consumo de Tiempo y Memoria



	Tiempo (ms) 	Mejor Memoria 
Ejecución ruta general	65	123731968
Ejecución tiempo para ruta	69	143654912
Ejecución número de carros	66	173314904
Ejecución rutas carro individualmente	93	137594168
Ejecución matriz distancia tiempo	9618	152856464
Ejecucion de todo el programa de seguido	9911	257949696

Gráfico 3: consumo de memoria de un algoritmo de ruteo y optimización de tiempo para entregas de pedidos a Domicilios

Consumo de Tiempo y Memoria

	Peor Memoria 	Memoria Promedio 
Ejecución ruta general	186067528	154899748
Ejecución tiempo para ruta	186222728	164938820
Ejecución número de carros	186377928	179846416
Ejecución rutas carro individualmente	186377852	161986010
Ejecución matriz distancia tiempo	174246672	163551568
Ejecucion de todo el programa de seguido	258864934	258407315

Software en funcionamiento

The image displays four terminal windows from a BlueJ IDE, showing the step-by-step execution of a Java program titled 'davanegasg_jravel_Proyecto_Entrega2'. The program prompts the user to enter a file with coordinates, which is 'dummy'. It then presents a menu of options: 1) Show the general route, 2) Print the time for this route, 3) Print the number of cars we will use to make this route, 4) Show the routes of each car individually, 5) Show the Distance-Time Matrix, 6) Save coordinates, and 0) Exit. The user selects option 1, and the program outputs 'La mejor ruta es: {0 -> 1 -> 2 -> 0}'. The user then selects option 3, and the program outputs 'El número de carros que usaremos para las entregas es: 1'. The terminal windows are arranged in a grid, with the first window at the top left, the second at the top right, the third at the bottom left, and the fourth at the bottom right. Each window has a title bar with the BlueJ logo and the file name. The first window shows the initial prompt. The second window shows the menu after the 'dummy' file is entered. The third window shows the result of selecting option 1. The fourth window shows the result of selecting option 3. Blue annotations include a box around the first window's title bar, a box around the first window's content area, a box around the first window's input field, a box around the second window's title bar, a box around the second window's content area, a box around the second window's input field, a box around the third window's title bar, a box around the third window's content area, a box around the third window's input field, and a box around the fourth window's title bar, a box around the fourth window's content area, and a box around the fourth window's input field.

```
BlueJ: Ventana de Terminal - davanegasg_jravel_Proyecto_Entrega2
Opciones
Por favor, ingrese un archivo con las coordenadas

Type input and press Enter to send to program

BlueJ: Ventana de Terminal - davanegasg_jravel_Proyecto_Entrega2
Opciones
Por favor, ingrese un archivo con las coordenadas
dummy
-----¿Qué quieres hacer?-----
1) Mostrar la ruta General
2) Imprimir el tiempo para esta ruta
3) Imprimir el número de carros que usaremos para hacer esta ruta
4) Mostrar las rutas de cada carro individualmente
5) Mostrar la Matriz Distancia-Tiempo
6) Guardar coordenadas
0) Salir
-----Selecciona una opción-----

to progr

BlueJ: Ventana de Terminal - davanegasg_jravel_Proyecto_Entrega2
Opciones
La mejor ruta es: {0 -> 1 -> 2 -> 0}
-----¿Qué quieres hacer?-----
1) Mostrar la ruta General
2) Imprimir el tiempo para esta ruta
3) Imprimir el número de carros que usaremos para hacer esta ruta
4) Mostrar las rutas de cada carro individualmente
5) Mostrar la Matriz Distancia-Tiempo
6) Guardar coordenadas
0) Salir
-----Selecciona una opción-----

Type input and press Enter to send to program

BlueJ: Ventana de Terminal - davanegasg_jravel_Proyecto_Entrega2
Opciones
El número de carros que usaremos para las entregas es: 1
-----¿Qué quieres hacer?-----
1) Mostrar la ruta General
2) Imprimir el tiempo para esta ruta
3) Imprimir el número de carros que usaremos para hacer esta ruta
4) Mostrar las rutas de cada carro individualmente
5) Mostrar la Matriz Distancia-Tiempo
6) Guardar coordenadas
0) Salir
-----Selecciona una opción-----

Type input and press Enter to send to program
```

Gráfico 4: Sistema de planificación óptima de domicilios

Software en funcionamiento

```
Blue: Ventana de Terminal - davanegasg_jravel_Proyecto_Entrega2
Opciones
La mejor ruta para el carro 1 es: {Deposito -> 1 -> 2 -> Deposito} y su tiempo es: [7.868721245533427] Horas

-----¿Qué quieres hacer?-----
1) Mostrar la ruta General
2) Imprimir el tiempo para esta ruta
3) Imprimir el número de carros que usaremos para hacer esta ruta
4) Mostrar las rutas de cada carro individualmente
5) Mostrar la Matriz Distancia-Tiempo
6) Guardar coordenadas
0) Salir
-----Selecciona una opción-----

Blue: Ventana de Terminal - davanegasg_jravel_Proyecto_Entrega2
Opciones
null {66.72156847676769 , 1.6688392119191922} {115.22641927960792 , 2.880660481990198} {108.3
|{66.72156847676769 , 1.6688392119191922} null {92.80086206496145 , 2.3200215516240363} {89.92
|{115.22641927960792 , 2.880660481990198} {92.80086206496145 , 2.3200215516240363} null {8.234
|{108.31345345800771 , 2.707836336450193} {89.92486474829974 , 2.2481216187074935} {8.23415447996
|{58.60928680678516 , 1.465232170169629} {9.407486380537577 , 0.23518715951343944} {97.7491728865

-----¿Qué quieres hacer?-----
1) Mostrar la ruta General
2) Imprimir el tiempo para esta ruta
3) Imprimir el número de carros que usaremos para hacer esta ruta
4) Mostrar las rutas de cada carro individualmente
5) Mostrar la Matriz Distancia-Tiempo
6) Guardar coordenadas
0) Salir
-----Selecciona una opción-----

Blue: Ventana de Terminal - davanegasg_jrav...
Opciones
Muchas gracias-----
-----por usar-----
-----el programa :D
-----Adios-----

Can only enter input while your programmi
```

Gráfico 4: Sistema de planificación óptima de domicilios