

RUTEO CONTROLADO CON TSP

***Jacobo Rave Londoño
Diego Alejandro Vanegas González
Medellín, 07/06/2021***

Estructuras de Datos Diseñada

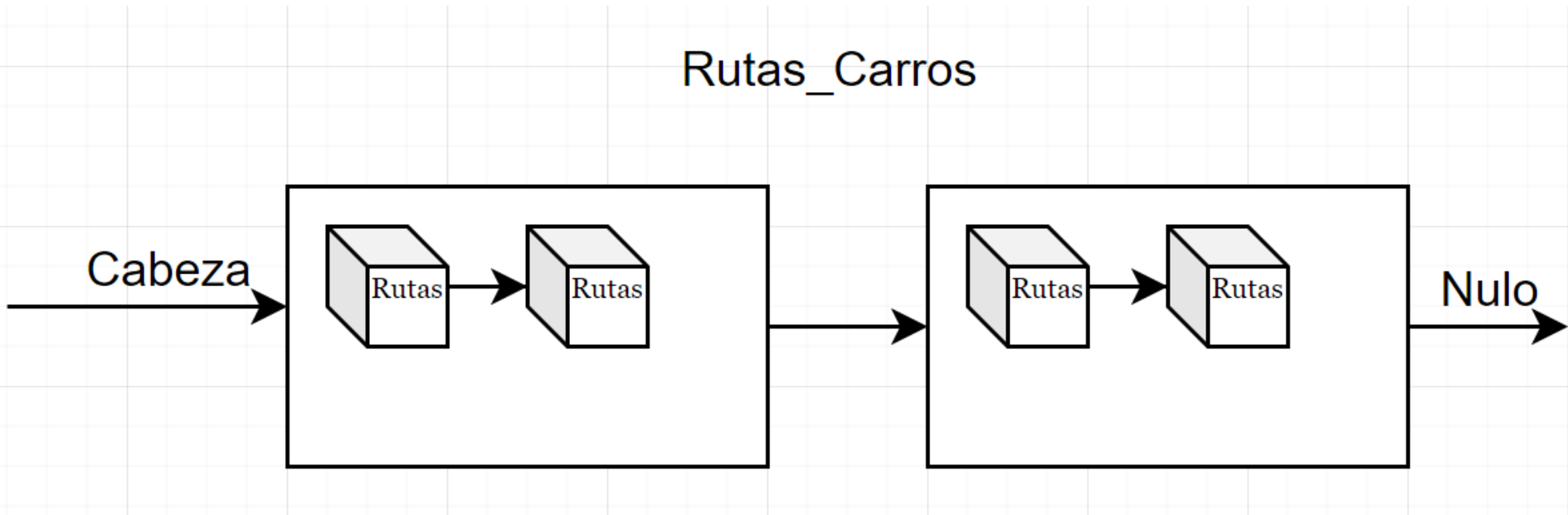


Gráfico 1: Lista de Listas. Una lista, contiene las diferentes rutas para cada carro

Estructuras de Datos Diseñada

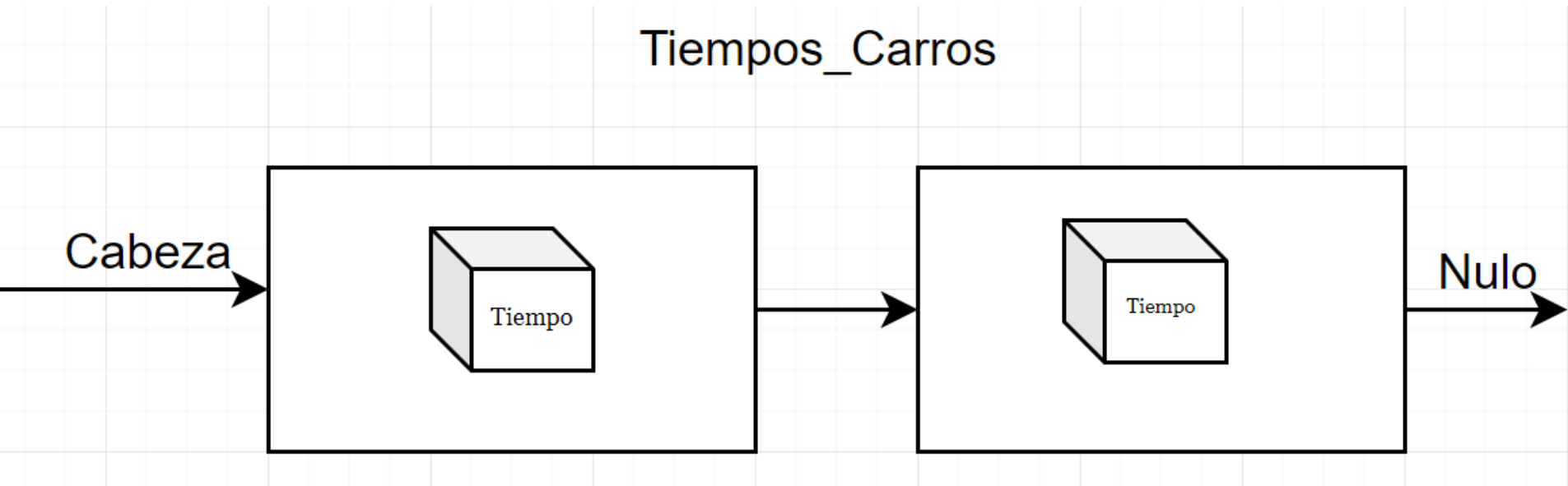


Gráfico 2: Lista de Listas. Una lista, contiene el tiempo que demora cada Carro en sus respectivas rutas

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

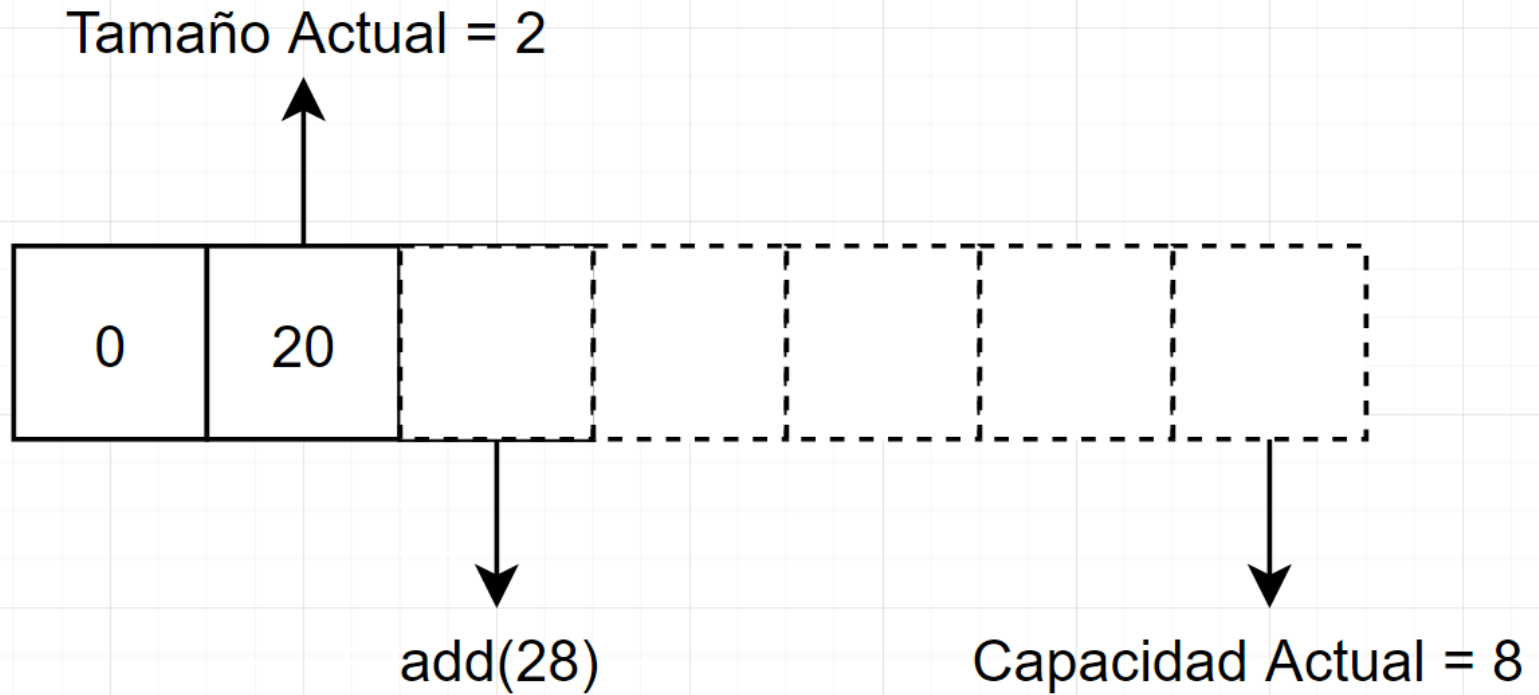
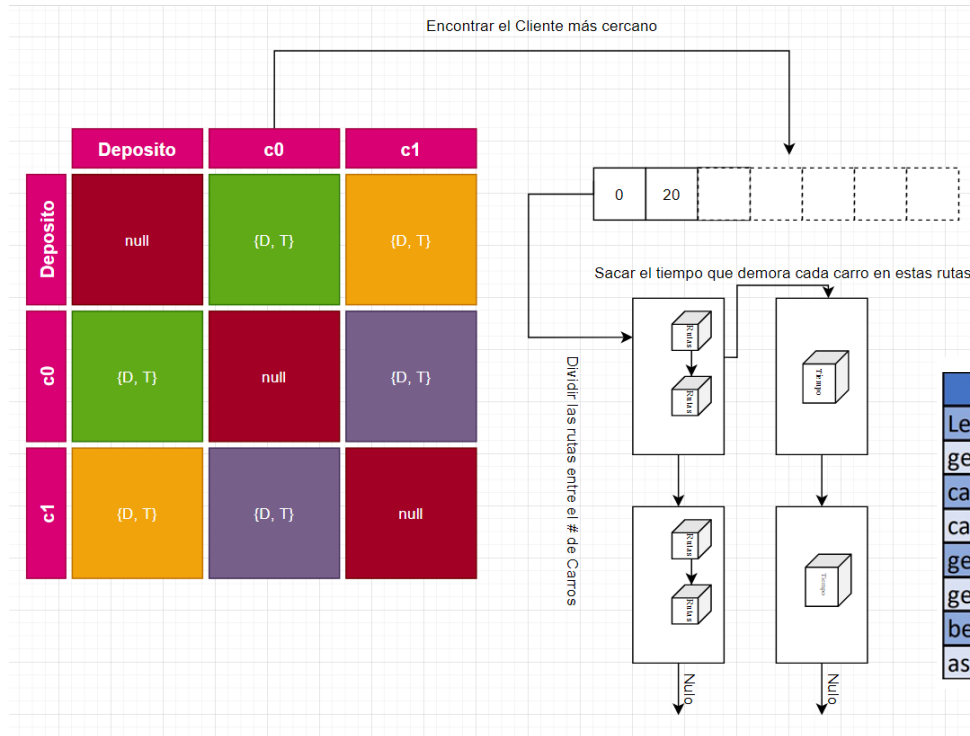


Gráfico 4: Array List. Una lista de arreglos para guardar la “ruta” más óptima para todo el recorrido

Explicación del algoritmo y su complejidad



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

Método	Complejidad
asignarCargaBateria	$O(n)$
permutaciones	$O(m)$
dfs	$O(n!)$
permute	$O(n!)$
bestPermu	$O(n)$

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

Tabla 1: Complejidad del algoritmo para los métodos


Criterios de Diseño del Algoritmo

Al analizar diferentes soluciones nos dimos cuenta que podíamos llegar a una solución óptima y factible la cuál es dividir el grafo en varios subgrafos, con estos lo que haremos es hacer permutaciones de las diferentes posibles rutas del subgrafo y cogemos la que menor coste en distancia tenga y esta la utilizamos para la ruta



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



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

Ejecución ruta carros invidualmente	10	90000000
-------------------------------------	----	----------

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

asignarCargaBateria	98000000	90000000
---------------------	----------	----------

Software en funcionamiento

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

Blue: 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-----

Blue: 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-----

Blue: 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-----
```

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}, {8.23415447990...
|{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