

Algoritmo para optimización de las rutas en la ciudad

Santiago Albisser

Juan Pablo Leal

Medellín, 16/05/2019

Diseño de operación de la estructura

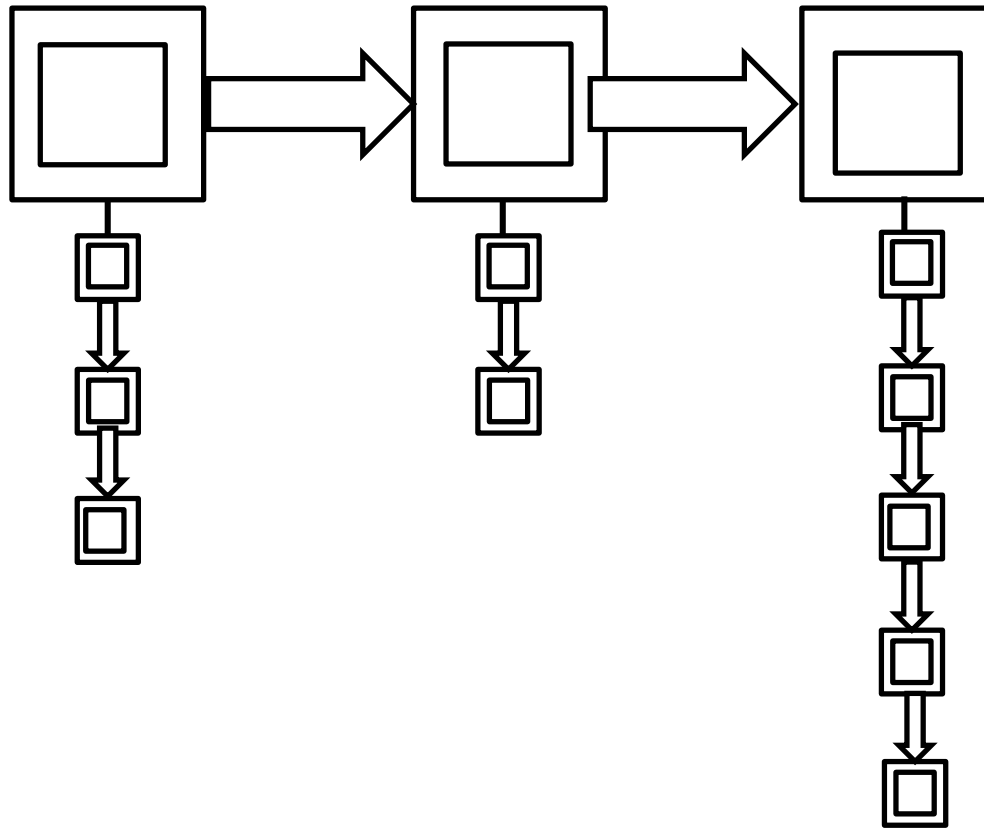


Figura 1: LinkedList de LinkedList para la asignación de vehículos.

Algoritmo y complejidad

SUBPROBLEMA	COMPLEJIDAD
Leer el archivo	$O(n)$
Hallar la distancia desde un nodo cualquiera hasta la universidad	c
Hallar la distancia mayor que se encuentra en una lista	$O(n^2)$
Asignar nodos a los vehículos	$O(n^2)$
Guardar el archivo	$O(m^2)$

Tabla 1: Complejidad del algoritmo para cada método utilizado

Criterio de diseño del algoritmo

Seleccionamos este algoritmo con esta estructura de datos debido a su complejidad.

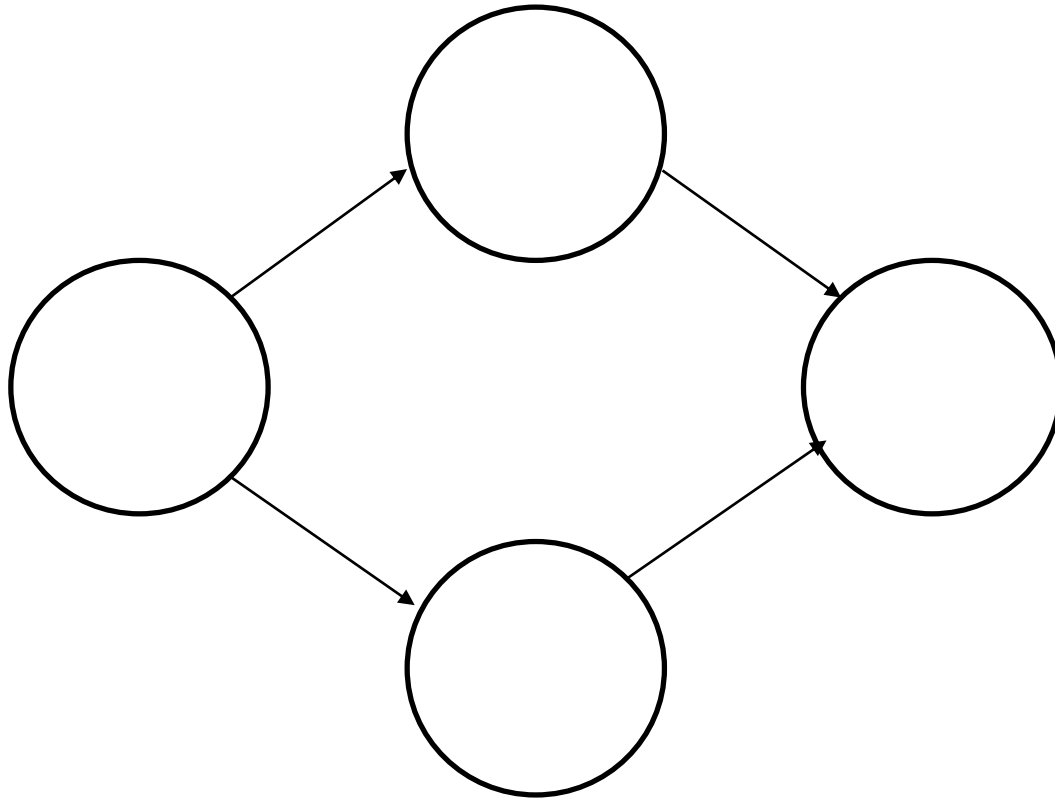
Ya que con las linkedlist enlazadas, al usar el método de agregar nos iba a dar una complejidad constante y no se iba a seguir aumentando el tiempo de ejecución del programa.

Tuvimos en cuenta que la suma del camino nuevo no fuera mayor a la distancia que había desde el nodo inicial hasta la universidad multiplicada por la constante.

Tiempo de ejecución

	Mejor tiempo	Peor tiempo	Tiempo promedio
P=1.1	11 ms	24 ms	17 ms
P=1.2	14 ms	21 ms	18,2 ms
p=1.3	15 ms	22 ms	17,66 ms

Tabla 2: Tiempo de ejecución del algoritmo para optimización de rutas en la ciudad para un grafo de 205 nodos con diferentes p



Resultados

Numero de nodos	p	Numero de carros
205	1.1	79
205	1.2	71
205	1.3	60

Tabla 3: Resultados del algoritmo para 205 nodos con distintos valores de la p.