Proyecto Final Algo2

Federico Lucero
Patricio Heduan

Cambios con respecto a la primera presentación

- Utilizar una estructura formal de grafos además del “Diccionario de Esquinas” era Overkill teniendo en cuenta la idea que habíamos planteado
- El diccionario dejó de ser unidimensional para ser bidimensional, ahora es un diccionario de diccionarios

Problemas encontrados:

- Precarga del mapa
- Como analizar las llegadas de las ubicaciones móviles a las ubicaciones fijas
- Encuentro de caminos
- Persistencia de datos
- Ubicaciones móviles

Precarga de Mapa:

¿Diccionario de diccionarios?

Cuando nos referimos a *“diccionario”* nos referimos a los diccionarios nativos de python

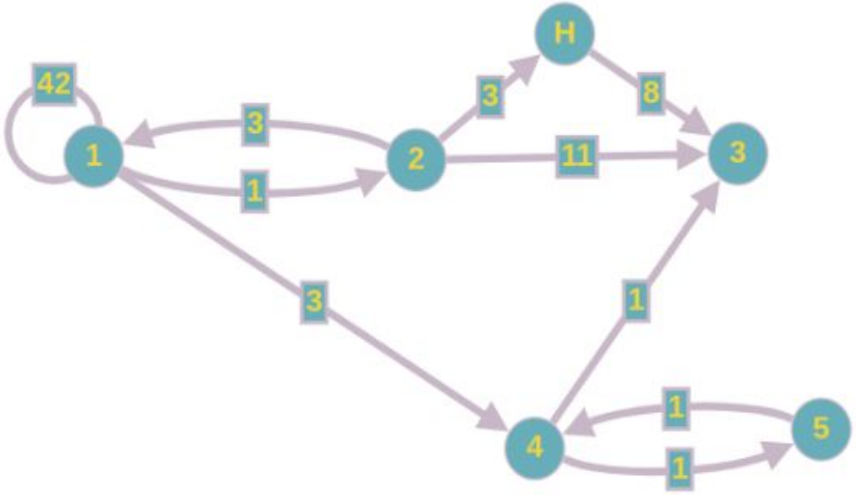
Entonces, construimos el mapa con un diccionario de diccionarios para hacer una semejanza a una matriz de adyacencia, pero un poco más eficiente (en la mayoría de las cosas)

Dentro de cada elemento de la matriz, se guarda la distancia entre 2 nodos, y de no ser una arista directa, se coloca además la arista más cercana al nodo perteneciente al camino más cercano

Un ejemplo visual:

E = {e1,e2,e3,e4,e5}

A = {<e1,e2,1>,<e1,e1,42>,<e1,e4,3>,<e2,e3,11>,<e2,e1,3>,<e4,e3,1>,<e4,e5,1>,<e5,e4,1>}



	e1	e2	e3	e4	e5
e1	42,None				
e2	1,None				
e3	4,e4	11,None		1,None	2,e4
e4	3,None			2,e5	1,None
e5	4,e4			1,None	2,e4
H	4,e2	3,None			

Ventajas

- Todo el resto de algoritmos que tengan que ver con el grafo tienen una complejidad máxima de O(n)

- Los algoritmos de búsqueda de caminos y acceso son muy ligeros y fáciles de entender

Desventajas

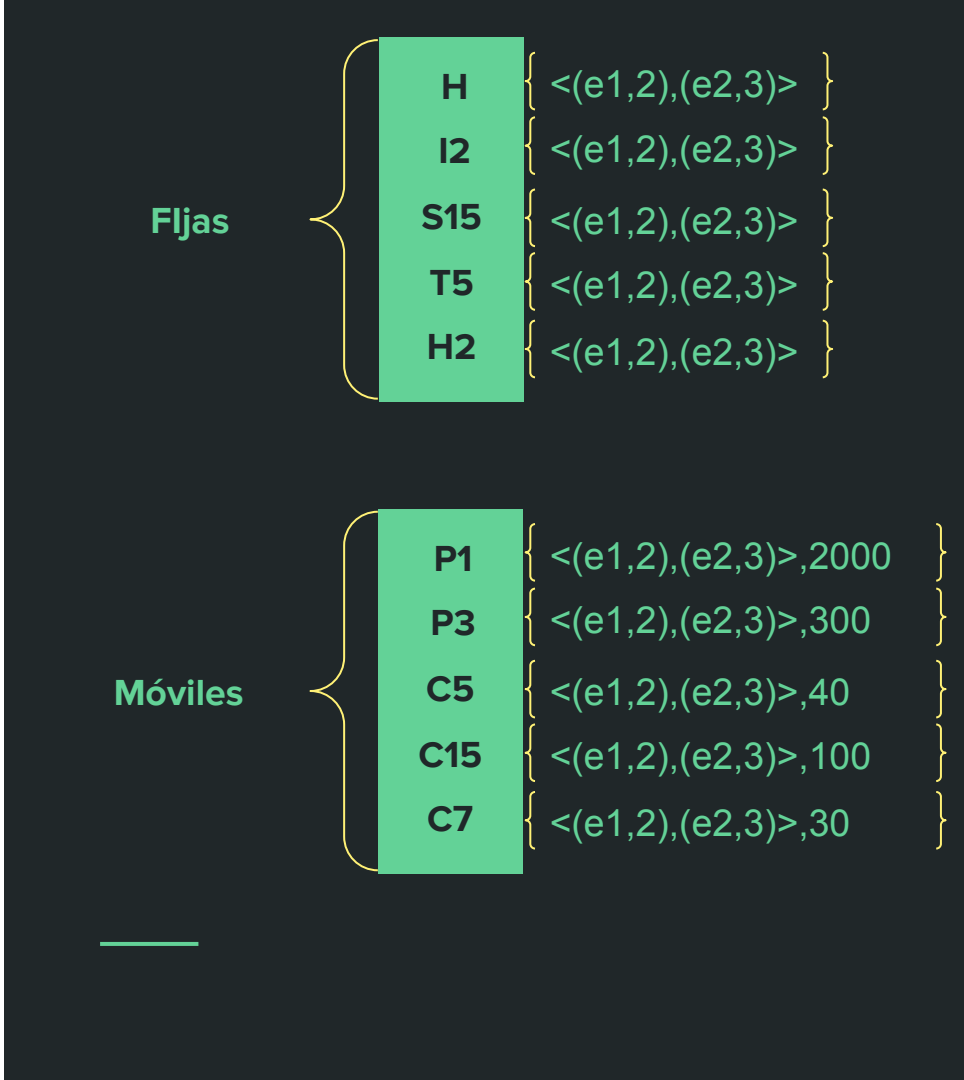
- El algoritmo de carga inicial es muy costoso O (n^3)

- No podemos aplicar algoritmos previamente conocidos de forma sencilla al grafo como BFS, DFS ni Dijkstra

Otras estructuras de datos:

Ubicaciones tanto fijas como móviles:

Más diccionarios nativos de python (para facilitar el acceso) aunque sea costosa la búsqueda



Persistencia de datos

- argparse: Se utiliza para analizar y gestionar los argumentos de línea de comandos en un script de Python.
- re: Permite trabajar con expresiones regulares para buscar y manipular texto en Python.
- pickle: Se utiliza para la serialización y deserialización de objetos en Python, lo que permite guardar y recuperar objetos en forma de bytes.

Posibles mejoras

- La carga del mapa podría haber sido menos costoso
- No utilizamos (de manera directa) nada programado ni implementado en el año
- Estructura para las ubicaciones móviles y su búsqueda