

Tarea 1

Diseño Avanzado de Algoritmos

Uber es un sistema transporte que nos permite movilizarnos desde un punto de la ciudad a otro. Para esto, solicitamos el servicio de transporte poniéndonos en contacto con algún conductor, el cual va hacia donde estamos para llevarnos a donde deseamos, cumpliendo con la solicitud. Se presenta una ciudad que tiene 1.000 cuadras de oeste a este y 1.000 cuadras de sur a norte, donde todas las cuadras tienen igual largo. Los vehículos se puede desplazar por la ciudad a una velocidad de 1[cuadra/unidad de tiempo].

La cantidad y posición de los ubers presentes en la ciudad está determinado por el horario: en la mañana hay 100 ubers, en la tarde 200 ubers y en la noche 50 ubers. Las solicitudes de los ubers van llegando cada ciertas unidades de tiempo de manera no uniforme. En cada horario se comienza a contar desde el tiempo 0, por ejemplo, en la mañana las solicitudes se comienzan a recibir desde el tiempo 0. Cada solicitud indica la coordenada de origen, la coordenada de destino y el tiempo en que ésta llega.

Implemente tres algoritmos para cumplir todas las solicitudes dadas

El primer algoritmo debe resolver las solicitudes de forma greedy.

El segundo algoritmo debe resolver las peticiones de manera aleatoria.

El tercer algoritmo debe resolver las solicitudes mediante el algoritmo Harmonic visto en clases.

La idea es que los Uber recorran la mínima distancia total, considerando las distancias de los trayectos que estos deben recorrer, es decir, la distancia donde se encuentre el Uber hacia donde se le esté solicitando y la distancia del origen de la solicitud hasta donde se tenga que llegar. Tenga en cuenta que para este caso, debe utilizar la distancia de Manhattan.

Formato de Entrada

La posición inicial de los ubers en cada horario está dada en 3 archivos: "manana.dat", "tarde.dat" y "noche.dat", donde cada uno contiene tantas líneas como ubers hayan en la ciudad en ese horario. En cada línea se entregan dos números, el primero indica la posición X dentro de la ciudad desde oeste a este, el segundo la posición Y desde sur a norte.

Un ejemplo particular es el siguiente:

```
2 4
12 780
199 347
```

La configuración anterior indica que inicialmente hay 3 ubers en la ciudad, donde el primero está en la posición (2,4), el segundo en (12,780) y el tercero en (199,347). Además, se entregan 3 archivos diferentes de solicitudes: "requests_1.dat", "requests_2.dat" y "requests_3.dat", en donde cada línea contiene la siguiente estructura:

$$x_i \ y_i - x_f \ y_f - t$$

donde $x_i \ y_i$ es la coordenada de origen, $x_f \ y_f$ la coordenada de destino a la que se quiere llegar y t el tiempo en que se realiza la solicitud. Así, un extracto de ejemplo sería:

```
120 790 - 300 700 - 3
```

La solicitud anterior indica que una persona quiere ir desde la posición (120, 790) a la posición (300,700) y la solicitud es realizada en el instante 3.

Formato de Salida

Como salida se debe generar un archivo en donde cada línea tenga dos números, donde el primero representa el instante de tiempo y el segundo el costo total asociado.

3 754

Así, por ejemplo la línea anterior indica que en el instante 3 los uber han recorrido 754 unidades de longitud. Se debe presentar un gráfico que muestre la función de costo en una ventana de tiempo entre 0 y 12.000 instantes. Este gráfico debe mostrar en el eje x el tiempo y en el eje y la función de costo.

Entrega de la Tarea

- Programe su tarea en Python y muestre su funcionamiento en un jupyter notebook.
- La tarea debe ser entregada a través del sistema moodle de la asignatura. Suba a moodle su notebook. No es necesario que suba los archivos de prueba.
- Por cada día de atraso en la entrega de la tarea se descontarán 10 puntos en la nota.
- Si se detecta COPIA la nota será 0 (CERO).

¿Qué debe mostrar su jupyter notebook?

Muestre como implementó cada algoritmo. Luego muestre las funciones de costo de cada algoritmos para cada archivo de prueba. Muestre la curva costo versus tiempo en su jupyter notebook. Incluya comentarios en su notebook.

Defensa de la Tarea

Debe mostrar su tarea en clases. Para ello dispondrá de 10 minutos en los cuales debe correr su jupyter notebook y explicarlo a sus compañeros.

Entrega del jupyter notebook en moodle: Viernes 24 de Mayo, hasta las 23:55 hrs.
Exposición de la tarea en clases: Lunes 27 de Mayo, bloque 3-4, en sala de clases.