

2.6. CoopC és una companyia de lloguer de cotxes que vol diversificar el seu negoci per tal de competir en els desplaçaments curts. La companyia té oficines distribuïdes a Barcelona i un dipòsit central de cotxes al Prat. CoopC vol un mètode que li permeti processar les peticions de trajectes curts rebuts pel dia següent. Per això, vol determinar el nombre de cotxes que ha de deixar a cada oficina a l'inici del dia dedicats a trajectes curts. Aquesta assignació ha de permetre tenir suficients cotxes disponibles al llarg del dia (en cadascuna de les oficines) per tal de poder cobrir tots els trajectes curts del dia. A més, CoopC vol minimitzar el nombre de cotxes destinats durant el dia a desplaçaments curts.

La informació de la qual disposa CoopC per a cada sol·licitud de trajecte curt per al proper dia és la següent: una tupla  $(l, t, l', t')$  on el parell  $(l, t)$  indica l'oficina i l'hora de recollida del vehicle i el parell  $(l', t')$  indica l'oficina i l'hora de devolució.

A efecte d'aquesta aproximació CoopC assumeix que tots els cotxes són idèntics i que disposa d'una flota prou gran per a poder cobrir qualsevol nivell de demanda de trajectes curts. A més, assumeix també que els cotxes seran recollits i retornats puntualment a les hores i locals estipulats.

Dissenyeu un algorisme voraç que permeti obtenir el nombre de cotxes que s'han d'assignar a cada oficina, de manera que, al llarg del dia, sempre hi hagi almenys un cotxe disponible en una oficina a l'hora en què algun client l'ha de recollir d'allà. Tenint en compte que mantenir immobilitzat un cotxe té un cost alt, l'assignació obtinguda ha de garantir el requisit de disponibilitat i ha d'assignar el menor nombre possible de cotxes a cada oficina.

## Solució

1-Dividir les tuples  $(l, t, l', t')$  en dues parts derents  $(l, t)$   $(l', t')$

2- Ordenar tots els parells anteriors. El criteri de prioritat és Lloc (t), temps (t) i primer les tornades de cotxe, en aquest ordre.

3- Ordenar en cada lloc les recollides i tornades de cotxes per hora de llegada / salida

En cada lloc diferent executar el següent algorisme

reservaDeCotxes = 0

cotxesADemanar = 0

Mentres ( assignacionsPendants ){

    if tornada de cotxe: reservaDeCotxes = reservaDeCotxes + 1

    Si es una Recollida: {

        Si reservaDeCotxes es 0: cotxesADemanar = cotxesADemanar +1

        Si no: reservaDeCotxes = reservaDeCotxes - 1

    }

El valor final de cotxesADemanar serà el nombre de cotxes que cada lloc diferent necessitarà. El cost de dividir les tuples i executar el bucle es lineal. Ordenar tots els parells  $(t, l)$ , té cost  $O(n \log n)$  amb un merge sort. Per tant, el cost total es  $O(n \log n)$  on  $n$  és el nombre de cotxes que llogarà l'empresa en el dia.