

Algorítmica (11593)

Recuperación del Primer Parcial 24 de enero de 2017



NOMBRE.	
NUMBRE	
TACAMETATA.	
MOMENTAL.	

1 3 puntos

Un empresario alquila una sala para dar recitales. La oferta es que el precio es lo mismo sea cual sea la duración del concierto. Recibe N peticiones y quiere maximizar su beneficio. Su socio está convencido de que la mejor estrategia es elegir primero las solicitudes de menor duración.

Se pide que hagas una función Python que elija los conciertos utilizando tanto la estrategia que indica el socio como la óptima (que debes conocer) y que diga si ambas estrategias coinciden o no en número de conciertos para la instancia recibida como argumento (es decir, la función devuelve un booleano). Ejemplo de llamada:

```
dos\_voraces([(1,3),(2,5),(4,7),(4,6)])
```

La tupla (1,3) indica que el concierto empieza en el instante 1 y termina en el instante 3. Son compatibles dos conciertos tal que uno termina y el otro empieza en el mismo instante. Puedes asumir que existe una función auxiliar solapan que recibe 2 tuplas y nos devuelve True si solapan y False en otro caso. También se pide: ¿Cuál es el coste temporal?

 $\mathbf{2}$ 1.5 puntos

Se ha resuelto, mediante programación dinámica, el cálculo del coste del camino más barato en el río Ebro. Hay E embarcaderos numerados de 0 a E-1 y queremos ir del primero al último sabiendo que solamente podemos llegar a un embarcaderos desde los 2 anteriores. La función c(i,j) nos devuelve el coste de ir de un embarcadero i a otro j. Nos dan este código:

```
def rio_Ebro(E, c):
C = [None] *E
C[0] = 0
C[1] = c(0,1)
for i in range(2, E):
  C[i] = min(C[i-1]+c(i-1,i), C[i-2]+c(i-2,i))
return C[E-1]
```

Y se pide que lo adaptes para recuperar la secuencia de embarcaderos utilizado el vector C (es decir, sin utilizar punteros hacia atrás).

3 5.5 puntos

Deseamos asfaltar los K kilómetros de carretera que separan dos ciudades. Tenemos N tipos de tramo con los que construir la carretera. Para cada tipo de tramos i $(1 \le i \le N)$ nos indican:

- su longitud en kilómetros L_i ,
- su coste en euros C_i ,
- el número de tramos en stock S_i .

Los tramos deben ser utilizados enteros y nos hemos de limitar al stock. Queremos conseguir tramos que sumen exactamente los K kilómetros minimizando el coste total. Se pide:

- 1. Especificar formalmente el conjunto de soluciones factibles X, la función objetivo a minimizar f y la solución óptima buscada \hat{x} .
- 2. Una ecuación recursiva que resuelva el problema y la llamada inicial.
- 3. El algoritmo iterativo asociado a la ecuación recursiva anterior para calcular el menor coste (no hace falta determinar el número de tramos necesarios de cada tipo).
- 4. El coste temporal y espacial del algoritmo iterativo.