Cognoms, Nom DNI

# Algorísmia QT 2017–2018

**Examen final** **15 de Gener de 2018**

Durada: 2h 50m

---

Instruccions generals:

- L'exercici 1 s'ha de resoldre fent servir l'espai reservat per a cada resposta.
- Heu d'argumentar la correctesa i eficiència dels algorismes que proposeu. Per això podeu donar una descripció d'alt nivell de l'algorisme amb les explicacions i aclariments oportuns que permetin concloure que l'algorisme és correcte i té el cost indicat.
- Heu de justificar totes les vostres afirmacions, en cas contrari la nota de la pregunta serà 0.
- Podeu fer crides a algorismes que s'han vist a classe, però si la solució és una variació n'haureu de donar els detalls.
- Es valorarà especialment la claredat i concisió de la presentació.
- Entregueu per separat la solució de cadascun dels exercicis.
- La puntuació total d'aquest examen és de **6 punts** (nota E).

**Exercici 1 (3 punts)**

(a) (0.3 punts) Utilitzant l'algorisme d'ordenació RADIX es poden ordenar en temps $O(n \lg n \lg n)$, $n$ enters diferents amb valors entre 1 i $n^{\lg n}$.

(b) (0.4 punts) Donat un vector $A$ amb $n$ elements, és possible posar en ordre creixent els $\sqrt{n}$ elements més petits i fer-ho en $O(n)$ passos?

(c) (0.3 punts) Antoni escull un enter entre 1 i $10^6$, Bartomeu ha d'endevinar quin és l'enter escollit fent el mínim nombre de preguntes que tinguin com a resposta (sí/no). Doneu una fita superior a aquest mínim nombre de preguntes.

(d) (0.3 punts) És cert que si a una xarxa $(s-t)$ totes les capacitats són racionals aleshores el flux màxim també serà un racional? (recordeu que un nombre racional és el quocient de dos enters, per ex, 3/4 o 7/5)

(e) (0.3 punts) Sigui $G=(V,E)$ un graf amb pes $w: E \to \mathbb{Z}^+$, aleshores l'aresta amb menys pes a un cicle ha de ser al MST, i la aresta de més pes al cicle NO ha de ser al MST. Digueu si aquesta afirmació és certa o falsa.

(f) (0.3 punts) Suposem que volem el canvi de $n$ cèntims, utilitzant el mínim de monedes de denominacions 1, 10 i 25 cèntims. Considereu la següent estratègia golafre: si la quantitat que em queda és m; agafem la moneda més gran que no sigui més gran que m; restem el valor d'aquesta moneda de m, i repetim.

Aquesta estratègia resol el problema plantejat?

(g) (0.3 punts) Recordeu el problema del recobriment de vèrtexs amb $k$-centres a $\mathbb{R}^2$: *Donat com entrada un conjunt de punts a $\mathbb{R}^2$, $X = \{x_1, \ldots, x_n\}$, on els punts venen representats per les seves coordenades cartesianes, donada una taula amb totes les $\frac{n(n-1)}{2}$ distancies Euclidianes $D = \{d(x_i, x_j)\}$, i donat un enter $k < n$, seleccionar un conjunt de $k$ centres $C \subseteq X$ i el mínim radi $r$ tal que els $k$ cercles centrats al punts a $C$ siguen un recobriment d'$X$.* Vam comentar a classe que el problema era NP-complet per $k > 1$. Doneu un algorisme determinista que resolgui exactament el problema del $k$-centre per el cas de que $k = 1$, i ho faci en temps polinòmic.

(h) (0.3 punts) Tenim una taula de hash $T$, amb grandària $2n$, on resolem les col·lisions amb llista d'adjacència. Suposem que introduïm $n/2$ claus a la taula $T$ utilitzant una funció d'hashing que és simple uniforme. Es cert que el nombre esperat de claus per a cada registre és $1/4$?

(i) (0.5 punts) Considerem una cadena amb longitud 62, formada amb l'alfabet $\Sigma = \{A, B, C, D, E\}$. Sabem que els cops que apareix cada lletra són: $A = 24, B = 12, C = 10, D = 8, E = 8$. Noteu que si comprimim la cadena, utilitzant un mètode de compressió amb longitud fixada, cada lletra necessitarà 3 bits i la longitud de la cadena comprimida és 186 bits. Per a comprimir més eficientment utilitzem el mètode de Huffman:

i. (0.1 punts) Quina tècnica algorísmica utilitza Huffman?

ii. (0.3 punts) Dibuixa l'arbre de Huffman.

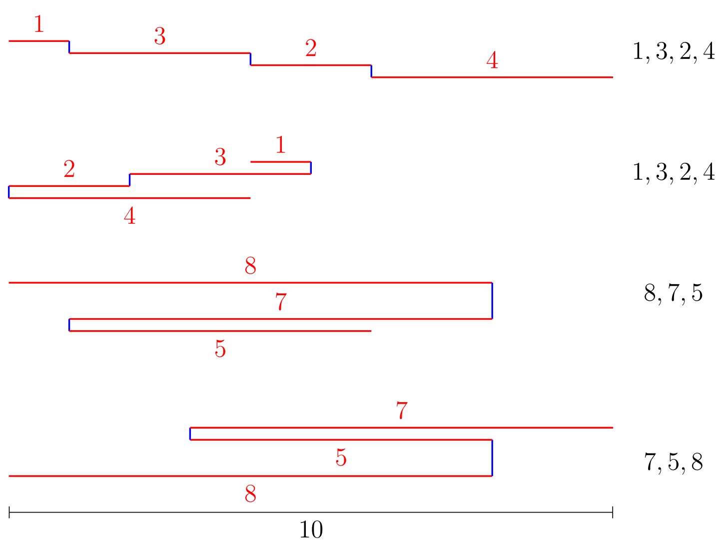
iii. (0.1 punts) Quina es la longitud total de la cadena comprimida amb Huffman?

---

**Exercici 2 (1.5 punts)** Un metre de fuster (com el de la figura de sota) està format per uns quants segments de fusta habitualment iguals. Cada segment és rígid i s'uneix al previ i/o al següent pels extrems de manera que es pot rotar completament a les unions.


En aquest problema considerarem una generalització de metre de fuster en el que els segments poden tenir longituds diferents encara que tots tenen la mateixa amplada. A més cada segment té com a molt 100cm de llargada. Així un metre de fuster està format per $n$ segments de llargades $l_1, \ldots, l_n$ (en aquest ordre) on totes les longituds dels segments son enters al interval $[0, 100]$. Per simplificar la notació considerarem també els extrems $A_0, \ldots A_n$, on $A_0$ és l'extrem lliure del primer segment, $A_1$ és l'extrem comú al primer i segon segment, etc, i $A_n$ és l'extrem lliure del segment $n$-ésim.

Volem analitzar el problema de plegar el metre per tal de ficar-lo a dintre d'una caixa. Per exemple, si els segments són de longitud 1, 3, 2 i 4, el metre es pot guardar en una caixa de longitud 10 (plegar a l'interval [0,10]), però podem fer-ho també en una caixa de longitud 5 (a l'interval [0,5]). Si els segments tenen longitud 8, 7 i 5 en aquest ordre, es pot guardar plegat en una caixa de 8, però si els segments son 7, 5 i 8, llavors la caixa més petita en la que es pot plegar té longitud 10 metres. A la figura de sota teniu una representació estilitzada i bidimensional d'aquests plegaments.



(a) (0.5 punts) Considereu l'algorisme següent

**procedure** FOLD INSIDE INTERVAL($L(n)$)
    Let $m = \max L[i]$
    Place $A_0$ at position 0
    **for** $i = 1, \ldots, n$ **do**
        **if** it is possible to place $A_i$ to the left of $A_{i-1}$ inside $[0, 2m]$ **then**
            place $A_i$ to the left of $A_{i-1}$
        **else**
            place $A_i$ to the right of $A_{i-1}$
        **end if**
    **end for**
**end procedure**

Demostreu que Fold inside interval determina un plegat que ens permet ficar el metre dintre de l'interval $[0, 2m]$ on $m$ es la longitud del segment més llarg i analitzeu-ne el seu cost.

(b) (0.5 punts) Considereu el problema Min-Fold: Donat un metre de fuster, format per $n$ segments de llargades $l_1, \ldots, l_n \in \mathbb{N}$ (en aquest ordre), $0 < l_i \leq 100$, trobar la llargada $\ell$ més petita que ens permeti ficar el metre dintre de l'interval $[0, \ell]$.

Dissenyeu un algorisme que ens permeti resoldre Min-Fold i analitzeu-ne el seu cost temporal i espacial.

Ajut: Penseu en com resoldre recursivament el problema de determinar si un metre de fuster es pot ficar a dintre de l'interval $[0, k]$ posant-hi l'extrem $A_0$ a la posició $j \in [0, k]$, per valors raonables de $k$.

(c) (0.25 punts) Analitza el cost de l'algorisme proposat a l'apartat (b) en el cas que els segments del metre de fuster poden tenir qualsevol longitud.

(d) (0.25 punts) És Fold inside interval una 2-aproximació a Min-Fold?

**Exercici 3 (1.5 punts)** El Rector de la UPC de cara a fomentar la germanor entre el personal de la UPC ha decidit establir activitats mensuals per incentivar la interacció entre els diferents estaments de PDI i PAS. Les festes se celebraran normalment un cop al mes. Suposem que hi han $k$ grups en el personal, $C_1, \ldots, C_k$, no necessàriament disjunts. El rectorat produeix una llista $L$ d'activitats que tindran lloc en el curs acadèmic, i per a cada activitat $a \in L$, hi ha un nombre màxim $M(a)$ i un nombre mínim $m(a)$ de gent que pot assistir a $a$. A més, com que els grups tenen diferents nivells d'influència, el rectorat estima, per a cada $j$, el nombre mínim $s(j)$ de persones del grup $C_j$ que s'han de convidar a cada activitat. Per una altra banda és ben conegut que alguns membres de la UPC no poden participar sovint en activitats lúdiques. Tenint en compte aquest aspecte el rectorat ha determinat per cada membre $i$ del personal de la UPC un valor $t(i)$ indicant el nombre màxim d'activitats a les què $i$ pot participar al llarg del curs. El problema consisteix en, en cas que suigui possible, decidir a qui convidar a cada activitat de manera que es compleixin les restriccions prèvies, o sigui:

- el nombre d'assistents a l'activitat $a$ és $\leq M(a)$ i $\geq m(a)$,
- per a cada grup $j$ hi ha d'haver com a mínim $s(j)$ persones assistint a cada activitat.
- la persona $i$ mai es convidada a més de $t(i)$ activitats.

(a) (0.25 punts) Indiqueu com modelaríeu aquest problema com un problema de programació entera. Una solució factible és el conjunt de invitacions que satisfan les restriccions, i volem minimitzar el nombre total de personal convidat.

(b) (1 punt) Com resoldríeu aquest problema utilitzant un model de xarxa de fluxos?

(c) (0.25 punts) Compareu la complexitat de resoldre el problema en cadascun dels dos models.