

# A-PROBLEMES-3.pdf



**Arnau\_FIB**



**Algorítmica**



**3º Grado en Ingeniería Informática**



**Facultad de Informática de Barcelona (FIB)  
Universidad Politécnica de Catalunya**



**Que no te escriban poemas de amor  
cuando terminen la carrera**



*(a nosotros por  
suerte nos pasa)*

**WUOLAH**

Que no te escriban poemas de amor  
cuando terminen la carrera ▶▶▶▶▶▶▶▶



WUOLAH

(la nosotros por suerte nos pasa)

No si antes decirte  
Lo mucho que te voy a recordar

Pero me voy a graduar.  
Mañana mi diploma y título he de  
pagar

Llegó mi momento de despedirte  
Tras años en los que has estado mi  
lado.

Siempre me has ayudado  
Cuando por exámenes me he  
agobiado

Oh Wuolah wuolilah  
Tu que eres tan bonita

(64)

$n$  cordes indexades de  $0, 2n-1$

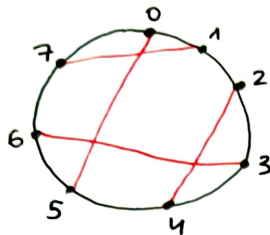
Seqüència  $S$  de parells  $\langle a_i, b_j \rangle$   $i, j \in \{0, 2n-1\}$

Cada  $i$  només apareix una vegada +  $n$  elements

Ex:

$n = 4$

$S = \{\langle 0, 5 \rangle, \langle 1, 7 \rangle, \langle 2, 4 \rangle, \langle 3, 6 \rangle\}$



a) Recurrència de  $T(i, j)$  on  
 $T(i, j)$  és el màxim de cordes  $\langle a, b \rangle$   
amb  $i \leq a, b \leq j$

b) Algorisme de cost polinòmic

a)

$$T(i, j) = \begin{cases} 0 & , i+1 = j \\ \max \{ T(i+1, j), T(i, j-1) \} & , \text{si } \nexists (i, j) \\ 1 + \max \{ T(i+1, j) + T(i, j-1) \} & \text{si } \exists (i, j) \end{cases}$$

$\exists (i, j)$  vol dir que hi ha una corda entre  $i$  i  $j$

Ens hem d'anar guardant les dades en una matriu

b) Inicialitzem  $T[2n][2n] = -1$

int calcule  $T$  ( int  $i$ , int  $j$  ) {

if (  $i+1 == j$  )  $T[i+1, j] = 0$ ; return 0;

if (  $T[i, j] \neq -1$  ) return  $T[i, j]$ ;

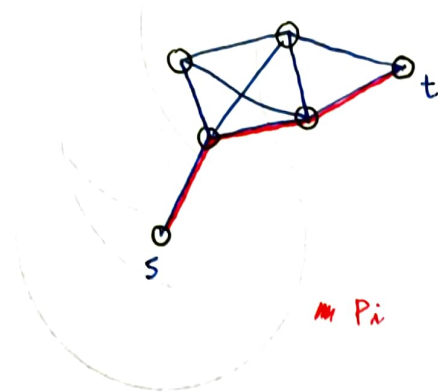
sum = (  $S$ .contains  $(i, j)$  ) ? 1 : 0;

return sum + calcule  $T(i+1, j)$  + calcule  $T(i, j-1)$ ;

}

WUOLAH

- 76) Xarxa agents mòbils. Distància de comunicació  $r$ .  
Representat com random geometric graph.



Volem mantenir dos nodes  $s, t \in V$  connectats.

$P_i$  és camí mínim de  $s$  a  $t$

$|P_i|$  és la longitud del camí  $\sim$  #arestes

$\mathcal{C}(P_1, \dots, P_m)$  canvis que es fan d'un  $P_i$  a

un  $P_{i+1} \sim$  # dif de índexs

$\left\{ \begin{array}{l} \text{Afegir 1 node} \rightarrow 1 \\ \text{Canviar 1 node} \rightarrow 2 \\ \dots \end{array} \right.$

Objectiu: minimitzar  $CT(P_0 \dots P_m) = \sum_{i=0}^m |P_i| + K \cdot \mathcal{C}(P_0 \dots P_m)$

- a) Suposem que  $\exists$  camí únic que és igual a  $G_0, G_1, \dots, G_m$   
Alg. polinòmic per trobar-lo de  $s$  a  $t$

Algorisme consisteix en: cada iteració calculem tots els possibles camins de  $s$  a  $t$  i ens guardem només la intersecció amb els de la iteració anterior. Si el camí és únic, la intersecció a la última iteració serà el camí buscat.

Cost:  $m$  vegades fer un BFS (modificat, que no vagi als pares) per guardar els camins possibles i després fer la intersecció  $\Rightarrow \Theta(m \cdot (|V| + |E| + |V|^2))$

- b) Trobar seqüència de camins amb cost mínim