# Universitat Oberta de Catalunya

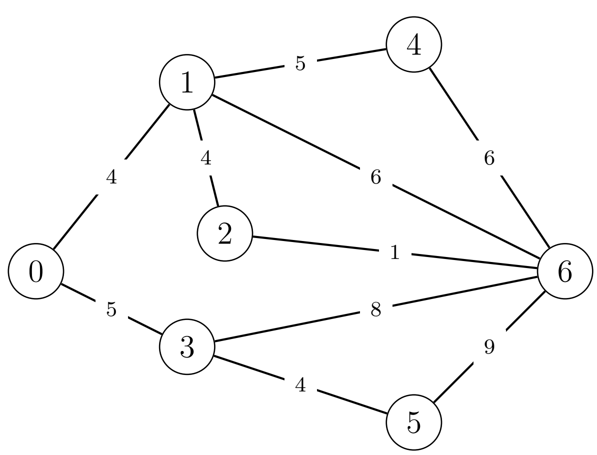
## Estudis d'Informàtica, Multimèdia i Telecomunicació

**ASSIGNATURA**: Grafs i Complexitat
Segona PAC. Mòduls 4 i 5.
Semestre de primavera de 2012 (del 12 d'abril al 2 de maig).

Si us plau, feu cas de les instruccions següents:

- Envieu la solució en un fitxer que haureu d'anomenar:
  **PAC1_Cognom1Cognom2nom.pdf**
- L'heu de lliurar a l'apartat "Lliurament i Registre d'AC" de l'aula.
- Numereu les respostes d'acord amb la numeració de les preguntes i els apartats.
- No us limiteu a donar les respostes als problemes proposats. Doneu, també, una explicació que justifiqui la resposta.

1. (Valoració d'un 20%) El *Spanning Tree Protocol* és un protocol de xarxa que evita la formació de llaços (*loops*) en una xarxa formada per nodes (*bridges*) i segments de la xarxa. Suposem que representem la xarxa com un graf on els vèrtexs representen nodes i les arestes són les connexions entre els nodes. El pes d'una aresta representa el cost de cada segment.



*a*) Utilitzeu l'algorisme més eficient per calcular un arbre generador d'aquest graf. Representeu-lo gràficament.

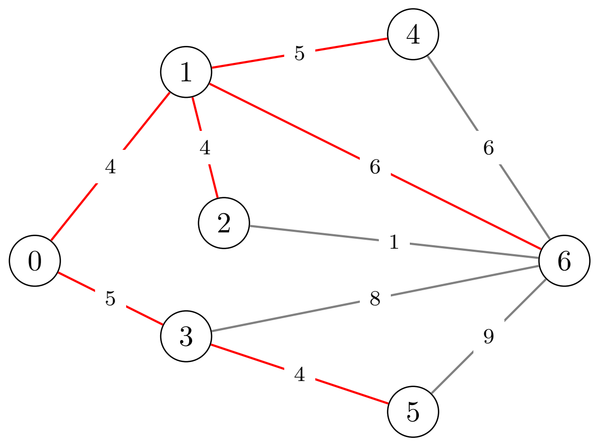
*b*) L'arbre obtingut en l'apartat anterior és un arbre generador minimal? Justifiqueu la resposta.

*c*) Si l'aresta $\{4, 6\}$ volem que formi part de la solució, l'arbre resultant serà un arbre generador minimal? Justifiqueu la reposta.

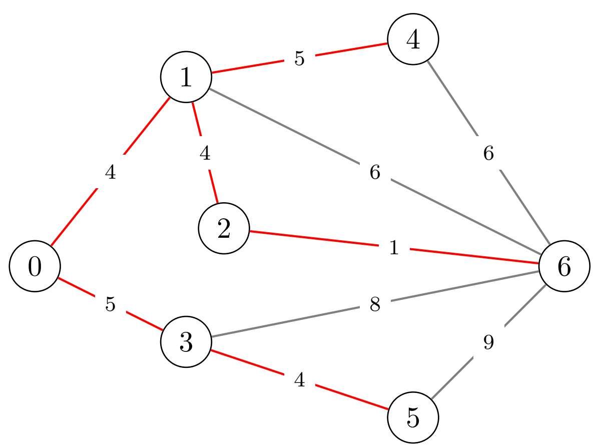
*d*) Demostreu que si les arestes d'un graf connex $G$ d'ordre $n$ totes tenen un pes diferent aleshores $G$ té un únic arbre generador minimal.

**Solució:**

*a*) Utilitzant l'algorisme BFS començant pel vèrtex 0 s'obté l'arbre generador:




*b*) L'arbre obtingut amb l'algorisme BFS té un pes de 28. Si apliquem, per exemple, l'algorisme de Kruskal obtindrem l'arbre,



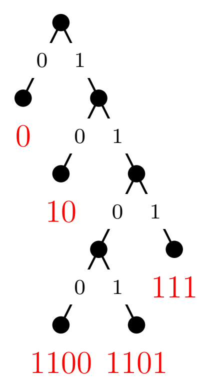
amb un pes de 23. Per tant, l'arbre obtingut amb l'algorisme BFS no és un arbre generador minimal.

*c*) Si afegim l'aresta $\{4, 6\}$ a l'arbre generador minimal obtingut en l'apartat anterior, llavors formarem el cicle $\{4, 6, 2, 1, 4\}$. Per tant, per obtenir un arbre generador hauríem de treure alguna de les arestes del cicle i, com que totes tenen un pes més petit que l'aresta $\{4, 6\}$, l'arbre obtingut no podrà ser un arbre generador minimal.

*d*) Si totes les arestes tenen un pes diferent llavors, aplicant l'algorisme de Kruskal, sempre triarem les $n - 1$ arestes de pes més petit que determina l'arbre generador minimal de forma única.

2. (Valoració d'un 20%) Un codi $C$ es diu *prefixe* si cap succesió de símbols de $C$ forma part d'un altra successió de símbols de $C$. Els codis prefixos s'utilitzen en la codificació i en la compressió de dades (mètode de Huffman) ja que permeten una descodificació eficient. Per exemple, el codi següent $C = \{0, 111, 1100, 1101, 10\}$ és un codi prefixe mentre que el codi $D = \{0, 10, 01, 101, 010\}$ no és un codi prefixe.

   Els codis prefixos es poden representar mitjançant arbres amb arrel. Per exemple, l'arbre següent representa el codi prefixe $C$:




   Les fulles de l'arbre s'identifiquen amb els símbols (paraules-codis) del codi. Els codis representats poden ser sobre qualsevol alfabet: binaris, ternaris,...

   *a*) Indiqueu quins dels codis següents són prefixos i en cas afirmatiu, representeu l'arbre corresponent a cada codi.

      1) $C_1 = \{00, 01, 11, 100, 101\}$.
      2) $C_2 = \{0, 01, 011, 0111\}$.
      3) $C_3 = \{02, 12, 01, 120, 012\}$.

4) $C_4 = \{00, 01, 02, 10, 11, 12, 20, 21, 22\}$.

b) La longitud mitjana d'un codi prefixe d'$n$ paraules-codi es calcula per la fórmula

$$\bar{L} = \frac{1}{n}\sum_{i=1}^{n} L_i$$

on $L_i$ és la longitud (nombre de símbols) de la paraula-codi $i$. Per exemple, pel codi $C$ la longitud mitjana és $\bar{L} = \frac{1+2+3+4+4}{5} = 2.8$. Els codis millors són els que tenen la menor longitud mitjana.

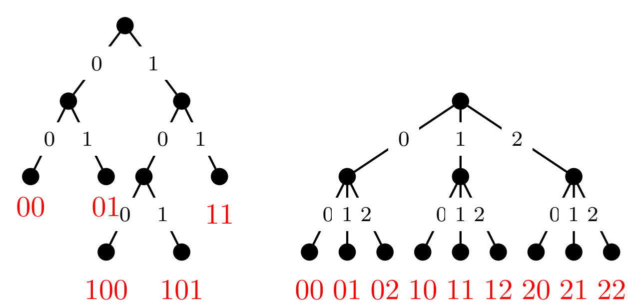
Relacioneu la longitud mitjana d'un codi prefixe amb les propietats de l'arbre que el representa.

c) En el codi ASCII cada caràcter està representat per un codi de 8 bits. És un codi prefixe? En cas afirmatiu digueu quina altura tindria l'arbre que el representa, quantes fulles tindria i quina classe d'arbre és.

d) Volem construir un codi ternari per representar 243 caràcters. Volem que sigui un codi prefixe de la menor longitud mitjana possible. Utilitzant la seva representació en forma d'arbre, indiqueu com es construiria aquest codi i quina seria la seva longitud mijana.

**Solució:**

a) Només $C_1$ i $C_4$ són codis prefixos. La seva representació gràfica és:




b) La longitud de cada paraula-codi coincideix amb el nivell de la fulla que el representa. Així, la longitud mitjana del codi seria igual a la suma dels nivells de totes les fulles dividit pel nombre de fulles.

c) El codi ASCII és un codi prefixe perquè cada paraula-codi té 8 bits i cap paraula-codi pot coincidir amb una altra paraula-codi. L'arbre tindria altura 8, 256 fulles i seria un arbre binari complet.

*d*) Hauríem de construir un arbre ternari de 243 fulles i amb la mínima altura. És a dir, $243 \leq 3^h$ on $h$ és l'altura de l'arbre. D'aquí, $h \geq \log_3 243 = 5$. Per tant, agafant $h = 5$ podem construir un arbre ternari complet d'exactament 243 fulles. El codi corresponent tindria longitud mitjana 5.

3. (Valoració d'un 20%) Quatre amics, *Anna*, *Bob*, *Carles* i *Diana*, van de vacances a una illa del carib. La taula següent representa els 9 punts d'interès cultural de l'illa juntament amb les carreteres que els uneixen (de doble sentit) i la distància en quilòmetres entre ells:

| | $P0$ | $P1$ | $P2$ | $P3$ | $P4$ | $P5$ | $P6$ | $P7$ | $P8$ |
|---|---|---|---|---|---|---|---|---|---|
| $P0$ | | 12 | | 6 | | 5 | | 4 | |
| $P1$ | | | 7 | | | | 8 | | 2 |
| $P2$ | | | | 7 | | | | 5 | |
| $P3$ | | | | | 2 | | | | 1 |
| $P4$ | | | | | | 9 | | | |
| $P5$ | | | | | | | 6 | | 15 |
| $P6$ | | | | | | | | 3 | |
| $P7$ | | | | | | | | | 5 |

*a*) Dibuixeu el mapa de carreteres de l'illa.

*b*) L'*Anna* vol fer diferents circuits (començant i acabant en el mateix punt) que incloguin sempre un nombre parell de punts diferents. Ho podrà fer?

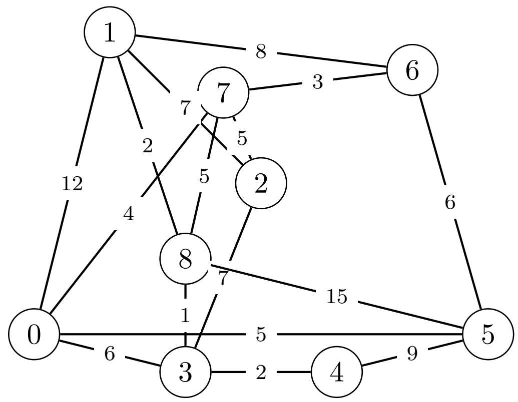
*c*) El *Bob* vol visitar tots els punts d'interès sense repetir-ne cap i tornar al punt de partida. Podrà fer-ho? En cas afirmatiu doneu el recorregut que haurà de seguir.

*d*) El *Carles* vol recórrer totes les carreteres de la illa, independentment del sentit, una sola vegada i tornar al punt de partida. Ho podrà fer? En cas afirmatiu doneu el recorregut que haurà de seguir. En cas negatiu, digueu si ho podrà fer encara que el punt d'arribada sigui diferent del de partida.

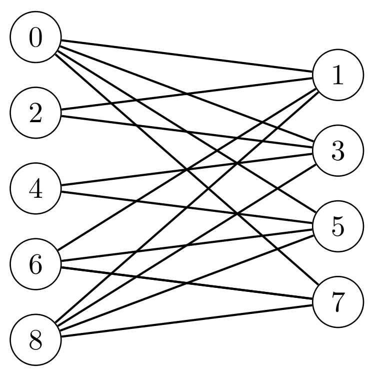
*e*) La *Diana*, com el Bob, vol visitar tots els punts d'interès sense repetir-ne cap i tornar al punt de partida però recorrent el menor nombre de quilòmetres possible. Ho podrà fer? En cas afirmatiu expliqueu com es trobaria la solució. En cas negatiu, calculeu el nombre mínim de quilòmetres que hauria de recòrrer per visitar tots els punts.

**Solució:**

*a*) El mapa de carreteres de l'illa serà:



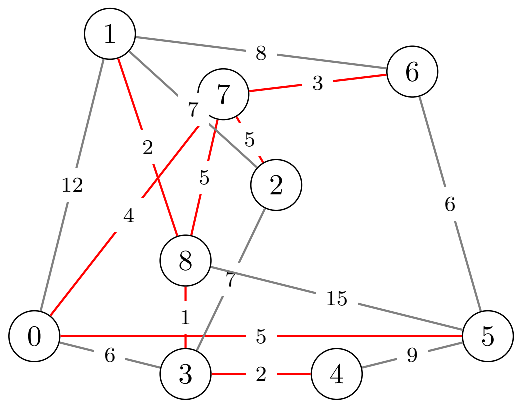
*b*) L'*Anna* ho podrà fer si tots els llocs pertànyen a cicles de longitud parella. Aquest és el cas del nostre graf ja que és bipartit i, per tant, no conté cicles de longitud senar.



*c*) El *Bob* vol trobar un cicle hamiltonià en el graf. Això no és possible ja que el graf és bipartit i els conjunts de vèrtexs de la partició no tenen el mateix cardinal.

*d*) En *Carles* vol fer un circuit eulerià. El graf és connex i té exactament dos vèrtexs de grau senar. Per tant, no és possible trobar un circuit eulerià però sí un recorregut eulerià.

*e*) La *Diana* vol resoldre el problema TSP sobre el graf. Però com que el graf no és hamiltonià no podrà resoldre el problema. El nombre mínim de quilòmetres que ha de recórrer serà el pes de l'arbre generador minimal,




que té un pes mínim 27.

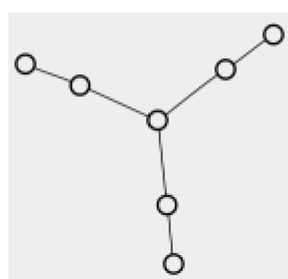
4. (Valoració d'un 20%)

   *a*) Demostreu que un arbre amb màxim grau $k$ té, com a mínim, $k$ fulles.

   *b*) Un hidrocarbur saturat és una mol·lècula $C_iH_j$, on cada carboni ($C$) té quatre enllaços i cada hidrògen ($H$) en té un, i que no conté cap cicle. Demostreu que una mol·lècula d'aquest tipus només pot existir si es verifica que $j = 2i + 2$.

   *c*) Demostreu que un graf $G$ és un arbre si i només si no conté cicles i té un únic arbre generador.

   *d*) Un arbre $A$ es diu que és una *eruga* quan conté com a subgraf un graf trajecte $T$, de manera que tot vèrtex de $A$, o bé pertany a $T$, o bé és adjacent a un vèrtex de $T$. Demostreu que un arbre és una eruga si i només si no conté com a subgraf al graf $Y$ de la figura.



**Solució:**

*a*) Ho veurem per inducció sobre l'ordre $n$. Evidentment per $n = 1$ és cert. Per veure el cas inductiu, considerem un arbre $T$ d'ordre $n + 1$.
En primer lloc observem que $T$ té alguna fulla: eliminant un vèrtex qualsevol obtenim

un graf amb $n$ vèrtexs, que per hipòtesi d'inducció té almenys $n$ fulles. Per tant, $T$ té també almenys $n$ fulles, on $n \geq 1$.
Ara eliminem una fulla qualsevol de $T$. L'arbre resultant $T'$ és un arbre amb un cert grau màxim $k$. Per hipòtesi d'inducció conté, almenys, $k$ fulles. Llavors $T$ té, com a mínim, $k+1$ fulles, mentre que el grau màxim serà $k$ o $k+1$ (en funció de si la fulla que té de més $T$ que $T'$ és o no adjacent a un vèrtex de grau $k$). En conclusió, hem vist que l'arbre $T$ té grau màxim $k$ o $k+1$, i $k+1$ fulles com a mínim, per tant, té almenys tantes fulles com el grau màxim.
També pot raonar-se de manera més directa considerant un vèrtex de grau màxim $k$ i els seus veïns: aquells que no siguin fulles tindran, a la seva vegada, veïns que seran fulles o no. De nou, considerem els veïns dels que no ho siguin. Aquest procés acabarà trobant $k$ fulles. (Observem que cal tenir un arbre, en particular que no hi ha cicles: en un graf qualsevol podríem tenir branques que s'unissin).

*b*) Aplicant el lema de les encaixades, $2m = 4i + j$. Com $m = n - 1$ per ser un arbre, i $n = i + j$, tenim que $2m = 2(n-1) = 2(i+j-1)$. De la primera i última de les igualtats obtenim $4i + j = 2(i+j-1)$. Operant s'obté $j = 2i + 2$.

*c*) Sigui $G$ un arbre, cal veure que l'únic arbre generador és ell mateix: $G$ té $n-1$ arestes, i tot arbre generador ha de tenir aquest mateix nombre d'arestes, per la qual cosa ha de contenir-les totes.
D'altre banda, suposem que un graf $G$ d'ordre $n$ té almenys dos arbres generadors. Com cadascun té $n-1$ arestes, i almenys tenen alguna aresta diferent l'un de l'altre, ajuntant les arestes d'ambdós tenim almenys $n$ arestes de $G$, per la qual cosa $G$ no pot ser un arbre.

*d*) Si $A$ conté a $Y$, no pot ser una eruga, ja que, independentment del trajecte $T$ que considerem, una de les fulles de $Y$ no pertanyerà al trajecte ni serà adjacent a ell.
D'altra banda, suposem que $A$ no és una eruga. Veurem que necessàriament ha de contenir al graf $Y$. Considerem qualsevol subgraf $T$ de $A$ que sigui un trajecte. Sigui $(x_1, ..., x_k)$ la successió de vèrtexs d'aquest trajecte. Com $A$ no és una eruga, hi ha un vèrtex $y_0$ que no pertany a $T$, ni és adjacent. Com $A$ és connex, hi ha un camí des de $y_0$ fins a $T$, format per una successió de vèrtexs $(y_0, y_1, y_2, ..., y_l, x_s)$, on $x_s$ pertany al trajecte $T$, i $l \geq 1$. Aleshores els vèrtexs $\{x_{s-2}, x_{s-1}, x_s, x_{s+1}, x_{s+2}, y_l, y_{l-1}\}$ ens determinen un subgraf que és $Y$. Per garantir que existeixen $x_{s-2}$ i $x_{s+2}$ cal que $s$ sigui diferent de $1, 2, k-1$ i $k$, però si no existís un $y_0$ que verifiqués això, aleshores $A$ seria una eruga.

5. (Valoració d'un 20%)

*a*) Un hongarès vol visitar les sis ciutats europees de la taula partint i acabant a Budapest. Justifiqueu que podem usar $TSP - aproximat$ i apliqueu-lo.

*b*) Trobeu una fita inferior del circuit òptim. Demostreu que el circuit trobat a l'apartat anterior no és lòptim.

*c*) Apliqueu l'algorisme de Hierholzer a $K_7$.

| | *Bar* | *Bel* | *Ber* | *Bru* | *Buc* | *Bud* |
|---|---|---|---|---|---|---|
| Barcelona | | 1528 | 1498 | 1063 | 1968 | 1499 |
| Belgrad | | | 999 | 1373 | 447 | 316 |
| Berlín | | | | 652 | 1293 | 689 |
| Brussel·les | | | | | 1770 | 1132 |
| Bucarest | | | | | | 640 |
| Budapest | | | | | | |

**Solució:**

*a*) Aplicant l'algorisme de Prim des de Budapest, obtenim aquestes arestes (en l'ordre en què s'indiquen): $Bud-Bel, Bel-Buc, Bud-Ber, Ber-Bru, Bru-Bar$. Recorrent l'arbre obtingut en preordre obtenim: $Bud, Bel, Buc, Ber, Bru, Bar$. Afegint Bud al final obtenim el circuit hamiltonià: $Bud, Bel, Buc, Ber, Bru, Bar, Bud$, amb un total de 5270 Km.

*b*) La meitat del TSP-aproximat ens dóna una fita inferior, o sigui 2635 Km (també seria una fita inferior la suma de les arestes de l'arbre generador minimal). Un circuit més curt al trobat seria, per exemple, $Bud, Ber, Bru, Bar, Buc, Bel, Bud$, amb un total de 5135 Km.

*c*) Si numerem els vèrtexs de l'1 al 7, una possibilitat seria la següent:

| Iteració | $v$ | $C'$ | $C$ |
|---|---|---|---|
| 0 | 1 | | {1} |
| 1 | 1 | {1,2,3,4,5,6,7,1} | {1,2,3,4,5,6,7,1} |
| 2 | 1 | {1,3,5,7,2,4,6,1 } | {1,3,5,7,2,4,6,1,2,3,4,5,6,7,1 } |
| 3 | 1 | {1,4,7,3,6,2,5,1 } | {1,4,7,3,6,2,5,1,3,5,7,2,4,6,1,2,3,4,5,6,7,1 } |