PAC1

Presentació

Aquesta PAC és una introducció a la teoria de grafs que cobreix els continguts estudiats en els 3 primers mòduls de l'assignatura. Els exercicis treballen tant els conceptes previs sobre funcions i algorismes, els fonaments de la teoria de grafs i els problemes de recorreguts i connectivitats sobre grafs.

Competències

En aquesta PAC es treballen les següents competències del Grau en Enginyeria Informàtica:

- Capacitat per utilitzar els fonaments matemàtics, estadístics i físics per comprendre els sistemes TIC.
- Capacitat per analitzar un problema en el nivell d'abstracció adequat a cada situació i aplicar les habilitats i coneixements adquirits per a resoldre'l.

Objectius

Els objectius concrets d'aquesta PAC són:

- Conèixer el concepte de complexitat temporal i espacial d'un algorisme i saber analitzar-la en algorismes concrets.
- Conèixer el concepte de graf i els diferents tipus de graf (grafs orientats, grafs ponderats, pseudografs, multigrafs, ...).
- Conèixer les principals propietats dels grafs i saber analitzar-les en un graf concret.
- Conèixer els problemes de connectivitat més usuals sobre grafs, els algorismes que els resolen i saber-los aplicar en un graf concret.
- Ser capaç de representar i analitzar un problema en termes de la teoria de grafs.

Descripció de la PAC

1. (Valoració d'un 20%) Disposem d'un teclat numèric per accedir a un laboratori. Cada treballador del laboratori ha de tenir una paraula d'accés $x_1 x_2 \dots x_r$ amb una longitud r mínima de 6 dígits i màxima de 8.

   (a) Quantes paraules d'accés es poden formar?

   (b) Quantes paraules d'accés es poden formar que continguin tots els dígits diferents?

   (c) Expresseu les solucions dels dos apartats anteriors a través de funcions $f : N_r \longrightarrow X$, detallant el tipus de funció, el paràmetre r i el conjunt X que es considera en cada cas.

   (d) Quantes paraules d'accés es poden formar de longitud exactament 7, si el 0 no pot aparèixer en la primera posició, i les paraules han de ser nombres parells?

Solució:

(a) Es poden formar $VR(10,6) = 10^6$ paraules d'accés de longitud 6, $VR(10,7) = 10^7$ de longitud 7 i $VR(10,8) = 10^8$ de longitud 8. Per tant, en total, es poden formar 111.000.000 paraules d'accés.

(b) Amb tots els dígits diferents es poden formar $V(10,6) = 10 \cdot 9 \cdot 8 \cdot 7 \cdot 6 \cdot 5 = 151200$ de longitud 6, $V(10,7) = 10 \cdot 9 \cdot 8 \cdot 7 \cdot 6 \cdot 5 \cdot 4 = 604800$ de longitud 7 i $V(10,8) = 10 \cdot 9 \cdot 8 \cdot 7 \cdot 6 \cdot 5 \cdot 4 \cdot 3 = 1814400$ de longitud 8. Per tant, en total, es poden formar $V(10,6) + V(10,7) + V(10,8) = 2570400$ paraules d'accés.

(c) Sigui $X = \{0, 1, \dots, 9\}$. Per al primer apartat, la solució coincideix amb la suma del nombre de funcions que hi ha del conjunt N_6 al conjunt X, del conjunt N_7 al conjunt X i del conjunt N_8 al conjunt X. Per al segon apartat, la solució representa el nombre de funcions considerades en el primer apartat que a més a més són injectives.

(d) En aquest cas, si el 0 no pot aparèixer en la primera posició, només hi haurà 9 opcions per al primer dígit. Per a cadascun dels dígits x_2, x_3, x_4, x_5, x_6 hi haurà 10 opcions. Finalment, com que en la darrera posició només poden aparèixer dígits del conjunt $\{0, 2, 4, 6, 8\}$, només hi haurà 5 opcions. Per tant, en total, es poden formar $9 \cdot 10^5 \cdot 5 = 4.500.000$ paraules d'accés.

---

2. (Valoració d'un 20%) Considereu l'algorisme següent on T és una llista d'enters de longitud $n > 1$.

```
funció Insercio(T)
  inici
    n ← Longitud(T)
    per i = 2 fins n
        element ← T[i]
        j ← i − 1
        mentre j ≠ 0 ∧ element < T[j] do
                T[j + 1] ← T[j]
                j ← j − 1
        fimentre
        T[j + 1] ← element
    fiper
    retorn T
  fi
```

   (a) Calculeu el resultat de les següents crides: *Insercio*([1,2,3,4,5,6,7]), *Insercio*([7,6,5,4,3,2,1]), *Insercio*([7,2,5,4,3,6,1]), *Insercio*([1,6,3,4,5,2,7]).

(b) Calculeu, en el pitjor dels casos, el nombre d'operacions elementals que efectua l'algorisme. Suposeu que obtenir la longitud a partir d'una llista només representa una operació elemental. I en el millor dels casos possibles?

(c) Determineu, en funció de la longitud de la llista, n, la complexitat de l'algorisme.

(d) Proposeu una alternativa que millori l'algorisme.

Solució:

(a) Els resultats són: [1,2,3,4,5,6,7], [1,2,3,4,5,6,7], [1,2,3,4,5,6,7], [1,2,3,4,5,6,7], respectivament.

(b) Les línies 5, 8 i 11 efectuen una operació elemental cada una. La línia 3 efectua dues operacions elementals, una per obtenir la longitud de la llista, i l'altra per l'assignació. Les línies 6 i 9 efectuen dues operacions elementals, una per la resta, i l'altra per l'assignació. La línia 4 efectua una inicialització, n comparacions i $n-1$ increments de la variable, per tant un total de $2n$ operacions elementals.

La condició de la línia 7 representa 3 operacions elementals (les dues comparacions i la connectiva lògica $\wedge$). En el pitjor dels casos, o sigui quan la llista T estigui inversament ordenada (o dit d'una altra manera, quan l'element i-èssim sigui més petit que tots els que van de 1 a $i-1$), haurem d'executar el bucle mentre $i-1$ vegades i el total d'operacions serà $6(i-1)$ (3 de la condició 7, 1 de la línia 8 i 2 de la línia 9).

El nombre d'iteracions del per és $n-1$, amb 4 operacions fixes (línies 5, 6 i 11), més el total d'operacions del mentre que és $6(i-1)$, més 3 operacions fixes d'avaluar la condició de la línia 7 quan sortim del bucle. En total tenim $4+6(i-1)+3 = 6i+1$ operacions per $i = 2, \ldots, n$, o sigui 13 (comencem per $i = 2$) $+19+\cdots+(6n+1)$ (acabem per $i = n$). Per tant, el nombre total d'operacions elementals del bucle extern (per) és $3n^2+4n-7$.

El nombre total d'operacions elementals serà, en el pitjor dels casos, $2+2n+3n^2+4n-7 = 3n^2+6n-5$.

En el millor dels casos possibles, quan la llista T ja està inicialment ordenada, no es farà mai la iteració mentre interna. En aquest cas, el nombre total d'operacions elementals serà $2+2n+7(n-1)$. Recordem: 2 de la línia 3, $2n$ de la línia 4 i $7(n-1)$ de les línies 5(1), 6(2), 7(3) i 11(1) per les $n-1$ iteracions del per.

(c) D'acord amb les propietats de la complexitat, aquest algorisme tindrà una complexitat $O(n^2)$, i en el millor dels casos $O(n)$.

(d) Es pot millorar l'algorisme fent una cerca dicotòmica en lloc d'una cerca lineal per trobar el lloc que pertoca a l'element i-èssim:

```
funció Insercio(T)
  inici
    n ← Longitud(T)
    per i = 2 fins n
        element ← T[i]
        esquerra ← 1
        dreta ← i − 1
        mentre esquerra ≤ dreta
                j ← (esquerra + dreta) div 2
                si element < T[j] aleshores
                                dreta ← j − 1
                          si_no
                                esquerra ← j + 1
                fisi
        fimentre
        per j = i − 1 fins esquerra pas −1
            T[j + 1] ← T[j]
        fiper
        T[esquerra] ← element
    fiper
    retorn T
  fi
```

3. (Valoració d'un 20%)

   (a) Sigui $G(V, A)$ el graf bipartit complet $K_{2,3}$. Justifica quines de les següents afirmacions són certes: el graf complementari de G és cíclic, G és cíclic i el diàmetre de G és 2.

   (b) Quin és el nombre mínim d'arestes que s'han d'eliminar d'un graf complet de n vèrtexs ($n \geq 4$) per tal que passi a tenir cuatre components connexes?

   (c) Proposeu un graf que tingui la mateixa seqüència de graus que el graf $G = C_{11} + N_1$ però que no sigui isomorf a G.

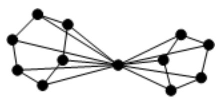
   (d) Sigui G un graf connex amb n vèrtexs. Demostreu que si el grau de cada vèrtex a G^c és el triple que el seu grau a G, aleshores G és regular i $n-1$ és un múltiple de 4.

Solució:

(a) Les tres afirmacions són certes. El graf complmentari G^c és el graf $C_3 \cup T_2$, per tant és cíclic. El graf G és cíclic, ja que també conté circuits. Finalment, el diàmetre de G és 2, ja que tots els seus vèrtexs estan entre ells, o bé a distància 1 o a distància 2.

(b) El graf complet K_n conté $\binom{n}{2} = \frac{n^2-n}{2}$ arestes. Un graf amb cuatre components connexes té, com a màxim, $\binom{n-3}{2} = \frac{n^2-7n+12}{2}$ arestes. Per tant, s'han d'eliminar, com a mínim, $\binom{n}{2} - \binom{n-3}{2} = 3n-6$ arestes.

(c) El graf següent



té la mateixa seqüència de graus que G però no és isomorf. Observeu que G té un cicle de longitud 11 i aquest graf no.

(d) Sigui n_i el grau del vèrtex $v_i \in G$. Aleshores $v_i \in G^c$ tindrà grau $n - n_i - 1 = 3n_i$. Per tant, per a tots els i, $n_i = \frac{n-1}{4}$, i tenim que G és regular. A més, per tal que n_i sigui un enter cal que $n-1$ sigui un múltiple de 4.

4. (Valoració d'un 20%)

   (a) Sigui $G(V, A)$ un graf simple i connex de mida m que té 6 vèrtexs de grau 3, 2 de grau 2 i x de grau 1; i sigui $G'(V', A')$ un graf simple i connex de mida m que té 2 vèrtexs de grau 3, 9 de grau 2 i y de grau 1.

      i. Quina relació hi ha entre els vèrtexs de grau 1 de G i de G'?

      ii. Quants vèrtexs de grau 1 de G es necessiten, com a mínim, per tal que el graf G deixi de ser connex?

   (b) Considerem les següents seqüències:
   $s_1 : 3, 3, 4, 1, 7, 2, 4$; $s_2 : 2, 4, 3, 3, 1, 3, 2$; $s_3 : 1, 4, 5, 2, 3, 3, 3$; $s_4 : 5, 4, 5, 2, 5, 1, 1, 1$

      i. Demostreu que algunes d'aquestes seqüències no són gràfiques sense usar l'algorisme de Havel-Hakimi.

      ii. Determineu quines d'aquestes seqüències són gràfiques usant l'algorisme de Havel-Hakimi.

Solució:

(a) i. Aplicant la fórmula dels graus, tenim que $2m = 3 \cdot 6 + 2 \cdot 2 + x$ per al graf G i $2m = 3 \cdot 2 + 2 \cdot 9 + y$ per al graf G'. Per tant, $2m = 3 \cdot 6 + 2 \cdot 2 + x = 3 \cdot 2 + 2 \cdot 9 + y$, d'on obtenim que la relació entre x i y és $x + 2 = y$.

ii. El graf G té $V = 6 + 2 + x$ vèrtexs i $A = m$ arestes. A més, com que $2m = 3 \cdot 6 + 2 \cdot 2 + x$, tenim que $m = 11 + x/2$. Tot graf connex ha de complir $|A| \geq |V| - 1$. Per tant, $11 + x/2 \geq 8 + x - 1$, d'on obtenim que $x \leq 8$. A més, com el nombre de vèrtexs de grau senar ha de ser parell, x ha de ser parell. Així tenim que necessitem, com a mínim, 10 vèrtexs de grau 1 per a que el graf G deixi de ser connex.

(b) i. La seqüència s_1 no pot ser gràfica, perquè el nombre de vèrtexs seria $n = 7$ i apareix un vèrtex de grau 7, quan el grau màxim hauria de ser $n - 1 = 6$. La seqüència s_3 no pot ser gràfica, ja que per la fórmula dels graus, el nombre de vèrtexs de grau senar ha de ser parell, i en aquesta següència apareixen 5 vèrtexs de grau senar

ii. Considerarem les seqüències s_2 i s_4.

s_2	s_4
$4,3,3,3,2,2,1$	$5,5,5,4,2,1,1,1$
$2,2,2,1,2,1$	$4,4,3,1,0,1,1$
$2,2,2,2,1,1$	$4,4,3,1,1,1,0$
$1,1,2,1,1$	$3,2,0,0,1,0$
$2,1,1,1,1$	$3,2,1,0,0,0$
$0,0,1,1$	$1,0,-1,0,0$
$1,1,0,0$	
$0,0,0$	
És gràfica	No és gràfica

5. (Valoració d'un 20%)

(a) Sigui $G(V,A)$ el graf de la Figura 1.

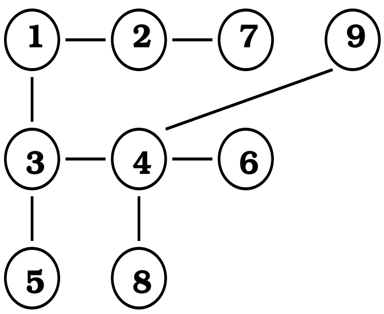


Figura 1: Graf simple

Aplicant l'algorisme *Breadth First Search (BFS)* (amplada prioritària), retorneu el recorregut que es faria prenent com a vèrtex de partida el vèrtex 3. En cas de poder escollir més d'un vèrtex en el mateix pas de l'algorisme, trieu primer aquell que té el número menor. Mostreu tots els passos de l'algorisme.

(b) Sigui $G(V,A)$ el graf ponderat, on $V = \{1,2,3,4,5,6,7\}$ i el pes de l'aresta $\{i,j\} \in A$ correspon a l'entrada $w_{i,j}$ de la matriu següent:

$$W = (w_{i,j}) = \begin{pmatrix} & 6 & 12 & 3 & & & \\ 6 & & & 9 & 20 & & \\ 12 & & & 6 & & 15 & \\ 3 & 9 & 6 & & 6 & 24 & 12 \\ & 20 & & 6 & & & 18 \\ & & 15 & 24 & & & 3 \\ & & & 12 & 18 & 3 & \end{pmatrix}$$

Aplicant l'algorisme de Dijkstra, retorneu el camí més curt i la distància entre els vèrtexs 1 i 6. En cas de poder escollir més d'un vèrtex en el mateix pas de l'algorisme, trieu primer aquell que té el número menor. Mostreu tots els passos de l'algorisme.

(c) Sigui d^2 la matriu correspon a l'instant d'execució $k = 2$ de l'algorisme de Floyd aplicat a un graf ponderat d'ordre 4. Calculeu la matriu d^4 corresponent a l'últim pas de l'algorisme.

$$d^2 = \begin{pmatrix} 0 & 3 & 5 & 7 \\ \infty & 0 & 2 & 3 \\ 1 & 2 & 0 & 1 \\ \infty & \infty & 5 & 0 \end{pmatrix}$$

(d) D'una xarxa d'autobusos de línia que uneixen diferents ciutats d'un país, en tenim la següent informació: la distància entre tots els parells de ciutats connectades directament per un autobús de línia, el temps que triga cada autobús per anar del seu origen al seu destí i el preu del bitllet per cada autobús.

Per a cadascún dels problemes següents indica quin graf consideraries (vèrtexs, arestes i pesos) i quin algorisme utilitzaries per resoldre'l:

i. Trobar el trajecte més econòmic entre dues ciutats concretes.
ii. Trobar el trajecte més curt entre tots els parells de ciutats qualssevol.
iii. Comprovar si podem desplaçar-nos entre dues ciutats qualssevol amb els autobusos de la xarxa.

Solució:

(a) La taula següent resumeix els passos de l'algorisme:

Q	Aresta afegida	S
3	-	$\emptyset$
3,1	$\{3,1\}$	$[\{3,1\}]$
3,1,4	$\{3,4\}$	$[\{3,1\},\{3,4\}]$
3,1,4,5	$\{3,5\}$	$[\{3,1\},\{3,4\},\{3,5\}]$
1,4,5	-	$[\{3,1\},\{3,4\},\{3,5\}]$
1,4,5,2	$\{1,2\}$	$[\{3,1\},\{3,4\},\{3,5\},\{1,2\}]$
4,5,2	-	$[\{3,1\},\{3,4\},\{3,5\},\{1,2\}]$
4,5,2,6	$\{4,6\}$	$[\{3,1\},\{3,4\},\{3,5\},\{1,2\},\{4,6\}]$
4,5,2,6,8	$\{4,8\}$	$[\{3,1\},\{3,4\},\{3,5\},\{1,2\},\{4,6\},\{4,8\}]$
4,5,2,6,8,9	$\{4,9\}$	$[\{3,1\},\{3,4\},\{3,5\},\{1,2\},\{4,6\},\{4,8\},\{4,9\}]$
5,2,6,8,9	-	$[\{3,1\},\{3,4\},\{3,5\},\{1,2\},\{4,6\},\{4,8\},\{4,9\}]$
2,6,8,9	-	$[\{3,1\},\{3,4\},\{3,5\},\{1,2\},\{4,6\},\{4,8\},\{4,9\}]$
2,6,8,9,7	$\{2,7\}$	$[\{3,1\},\{3,4\},\{3,5\},\{1,2\},\{4,6\},\{4,8\},\{4,9\},\{2,7\}]$
6,8,9,7	-	$[\{3,1\},\{3,4\},\{3,5\},\{1,2\},\{4,6\},\{4,8\},\{4,9\},\{2,7\}]$
8,9,7	-	$[\{3,1\},\{3,4\},\{3,5\},\{1,2\},\{4,6\},\{4,8\},\{4,9\},\{2,7\}]$
9,7	-	$[\{3,1\},\{3,4\},\{3,5\},\{1,2\},\{4,6\},\{4,8\},\{4,9\},\{2,7\}]$
7	-	$[\{3,1\},\{3,4\},\{3,5\},\{1,2\},\{4,6\},\{4,8\},\{4,9\},\{2,7\}]$
$\emptyset$	-	$[\{3,1\},\{3,4\},\{3,5\},\{1,2\},\{4,6\},\{4,8\},\{4,9\},\{2,7\}]$

Per tant, el recorregut és $3, 1, 4, 5, 2, 6, 8, 9, 7$.

(b) La taula següent resumeix els passos de l'algorisme:

1	2	3	4	5	6	7
$(0,1)$	$(\infty,1)$	$(\infty,1)$	$(\infty,1)$	$(\infty,1)$	$(\infty,1)$	$(\infty,1)$
$(0,1)^*$	$(6,1)$	$(12,1)$	$(3,1)$	$(\infty,1)$	$(\infty,1)$	$(\infty,1)$
$(0,1)$	$(6,1)$	$(9,4)$	$(3,1)^*$	$(9,4)$	$(27,4)$	$(15,4)$
$(0,1)$	$(6,1)^*$	$(9,4)$	$(3,1)$	$(9,4)$	$(27,4)$	$(15,4)$
$(0,1)$	$(6,1)$	$(9,4)^*$	$(3,1)$	$(9,4)$	$(24,3)$	$(15,4)$
$(0,1)$	$(6,1)$	$(9,4)$	$(3,1)$	$(9,4)^*$	$(24,3)$	$(15,4)$
$(0,1)$	$(6,1)$	$(9,4)$	$(3,1)$	$(9,4)$	$(18,7)$	$(15,4)^*$
$(0,1)$	$(6,1)$	$(9,4)$	$(3,1)$	$(9,4)$	$(18,7)^*$	$(15,4)$

Per tant, el camí més curt del vèrtex 1 al 6 és $(1, 4, 7, 6)$ amb una distància de 18.

(c) Aplicant l'algorisme de Floyd, obtenim les matrius següents:

$$d^3 = d^4 = \begin{pmatrix} 0 & 3 & 5 & 6 \\ 3 & 0 & 2 & 3 \\ 1 & 2 & 0 & 1 \\ 6 & 7 & 5 & 0 \end{pmatrix}.$$

(d) En tots els casos, els vèrtexs serien les ciutats i hi hauria una aresta entre dos vèrtexs si les ciutats corresponents estan connectades directament per un autobús de línia.

i. El pes a assignar a cada aresta seria el preu del bitllet per a cada autobús, i l'algorisme a aplicar seria el de Dijkstra.

ii. El pes a assignar a cada aresta seria el temp que triga cada autobús, i l'algorisme a aplicar seria el de Floyd.

iii. No cal assignar pesos a les arestes, i l'algorisme a aplicar seria el BFS o DFS.

Recursos

Recursos Bàsics

- Mòdul didàctic 1. Conceptes previs: funcions i algorismes
- Mòdul didàctic 2. Fonaments de grafs
- Mòdul didàctic 3. Recorreguts i connectivitat
- Col·lecció de problemes

Recursos Complementaris

- PACs i exàmens de semestres anteriors
- Programari per a l'estudi d'algorismes sobre grafs
- Enllaç: Applets interactius sobre algorismes de grafs

Criteris d'avaluació

- La PAC s'ha de resoldre **de forma individual**.
- Cada exercici té un pes del 20% de la nota final.
- És necessari justificar la resposta a cadascun dels apartats. Es valorarà tant la correctesa de la resposta com la justificació donada.
- En els apartats on calgui aplicar algun algorisme, es valorarà la tria de l'algorisme apropiat, els passos intermedis, el resultat final i les conclusions que se'n derivin.

Format i data de lliurament

Cal lliurar **un únic document** PDF amb les respostes a tots els exercicis. El nom del fitxer ha ser: **PAC1_Cognom1Cognom2Nom.pdf**.

Aquest document s'ha de lliurar a l'espai de **Lliurament i Registre d'AC** de l'aula **abans de les 23:59 del dia 23/10/2013**. **No s'acceptaran lliuraments fora de termini.**