Algorísmia QP 2018–2019

Una solució per al Examen final **11 de Juny de 2019**

Durada: 3h

Exercici 1 (4 punts)

(a) (0.5 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 $n^{\lg n}$?

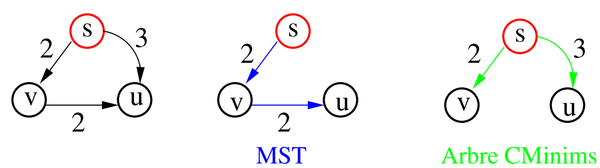
Solució: Cert, si utilitzem RADIX amb base n, aleshores cada enter nombre te $d = \log_n n^{\lg n} = \lg n$ dígits, per tant el temps de computació es $\Theta(nd) = \Theta(n \lg n)$ i $n \lg n = O(n \lg n \lg n)$

(b) (0.5 punts) Si ens donen un vector A no ordenat, amb n elements diferents i volem trobar $n/2$ elements d'A que tinguin la mateixa mediana que A, qualsevol algorisme que utilitzem tindra $\omega(n)$ passos (i.e. no serà lineal).

Solució: FALS, es possible fer-ho en temps $O(n)$ utilitzant la seleccio linear determinista: trobar la mediana en temps linear, agafeu $n/4$ elementd de la dreta i $n/4$ elements de l'esquerra. Aquests $n/2$ elements tenen la mateixa mediana que A.

(c) (0.5 punts) Donat un digraf $\vec{G} = (V, \vec{E}, w)$ amb pesos, i un vèrtex $s \in V$, és cert que l'arbre de camins mínims arrelat a s ha de ser necessàriament un MST de $\vec{G}$?

Solució: FALS veure figura



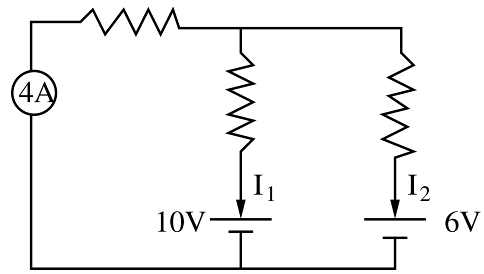
(d) (0.5 punts) Si una iteració de Ford-Fulkerson envia 1 unitat de flux a través de l'aresta (u, v) aleshores, en el flux màxim en finalitzar l'execució, el flux a través de (u, v) serà ≥ 1?

Solució: FALS Pot passar que a una iteració posterior el camí d'augmentació baixe a 0 el flux a traves (u, v).

(e) (0.5 punts) Considereu una xarxa de comunicació formada per n servidors que modelitzem com un graf $G = (V, E)$ no dirigit amb $|V| = n$ vèrtexs i $|E| = m$ arestes. Per a cada $(i, j) \in E$, sigui $b_{ij} = b_{ji}$ l'amplada de banda (bandwith) entre i i j. Donat un camí P de $v \to u$ definim l'amplada de banda del camí P com el valor de l'amplada de banda de l'aresta a P amb mínima amplada de banda. Donats dos vèrtexs fixats s i t doneu un algorisme per a trobar el camí amb màxima amplada de banda entre s i t. Doneu la complexitat del vostre algorisme.

Solució: Podem adaptar qualsevol algorisme per calcular distancies, canviant suma per mínim. En aquest cas els pesos son positius i podem adaptar Dijkstra amb cost $O(m + n \lg n)$.

(f) (0.5 punts) Volem carregar dues bateries de 10 i 6 volts, respectivament. Per fer això dissenyem un circuit com el de la següent figura:



Volem determinar les intensitats de corrent I_1 i I_2 que maximitzen la potència total del circuit. Cal tenir en compte que si donem més de 3 ampers d'intensitat a cada bateria, les cremarem, i que el sistema pot mantenir una intensitat total de 4 ampers com a màxim. Finalment, per tal d'assegurar que cap bateria es descarrega, necessitem assegurar que les intensitats I_1 i I_2 són sempre més grans o iguals a 0.

Enuncieu el problema com un sistema de programació lineal, on maximitzeu una funció objectiu amb unes restriccions. Dibuixeu el polítop resultant i la funció objectiu. Quina serà la una solució optima? (Recordeu la potencia $P = V \cdot I$ i que, a un circuit muntat en paral·lel, la intensitat resultant és la suma de les intensitats a cada línea).

Solució:

$$\begin{aligned} \max \ \ & 10I_1 + 6I_2 \\ \text{subject to:} \ & I_1 \leq 3 \\ & I_2 \leq 3 \\ & I_1 + I_2 \leq 4 \\ & I_1 \geq 0 \\ & I - 2 \geq 0 \end{aligned}$$

(g) (1 punt) Tenim 20 elements i fem un MakeSet. Suposem que els conjunts resultants estan enumerats de 0 fins a 19. Executem el següent codi:

for $i = 0$ **to** 14 **do**
  union(**find**(i), **find**($i + 5$))
end for
for $i = 15$ **to** 17 **do**
  union(**find**(i), **find**($i + 5$))
end for

Dibuixeu la seqüencia d'arbres fins a arribar al resultat final (dos arbres) quan la implementació utilitza unió per rang amb compressió de camins. Doneu el rang dels nusos al dos arbres resultants.

Exercici 2 (2 punts) – Traductors –

El centre de documentació de la UE gestiona el procés de traducció de documents pels membres del parlament europeu. En total han de treballar amb un conjunt de n idiomes. El centre ha de gestionar la traducció de documents escrits en un idioma a tota la resta d'idiomes.

Per fer les traduccions poden contractar traductors. Cada traductor està especialitzat en dos idiomes diferents; és a dir, cada traductor pot traduir un text en un dels dos idiomes que domina a l'altre, i viceversa. Cada traductor té un cost de contractació no negatiu (alguns poden treballar gratis).

Malauradament, el pressupost per a traduccions és massa petit per contractar un traductor per a cada parell d'idiomes. Per tal d'optimitzar la despesa, n'hi hauria prou en establir cadenes de traductors; per exemple: un traductor anglès $\leftrightarrow$ català i un català $\leftrightarrow$ francès, permetria traduir un text de l'anglès al francès, i del francès a l'anglès. Així, l'objectiu és contractar un conjunt de traductors que permetessin la traducció entre tots els parells dels n idiomes de la UE, amb cost total de contractació mínim.

El matemàtic del centre els hi ha suggerit que ho poden modelitzar com un problema en un graf amb pesos $G = (V, E, w)$. G té un node $v \in V$ per a cada idioma i una aresta $(u, v) \in E$ per a cada traductor (entre els idiomes u i v de la seva especialització); el pes de cada aresta seria el cost de contractació del traductor en qüestió. En aquest model, un subconjunt de traductors $S \subseteq E$ permet portar a terme la feina si al subgraf $G_s = (V, S)$ hi ha un camí entre tot parell de vèrtexs $u, v \in V$; en aquest cas direm que S és una *selecció vàlida*. Aleshores, d'entre totes les seleccions vàlides han de triar una amb cost mínim.

(a) Demostreu que quan S és una selecció vàlida de cost mínim, $G_s = (V, S)$ no té cicles.

(b) Proporcioneu un algorisme eficient per a resoldre el problema. Justifiqueu la seva correctesa i el seu cost.

Solució:

(a) Supngamos que $G_s = (V, S)$ es una selección válida de coste mínimo que tiene ciclos. Si eliminamos una arista (u, v) de un ciclo en $G_s = (V, S)$ seguimos teniendo caminos entre todos los vértices, ya que podemos ir de u a v a través de lo que queda del ciclo.

Como la selección tiene coste mínimo, y eliminando una arista de un ciclo también es solución. Tenemos que todas las aristas de un ciclo tienen coste 0. Así, mientras tengamos ciclos vamos eliminando una arista de peso 0 del ciclo. Hasta que tengamos una selección válida con coste mínimo sin ciclos.

(b) Por el aparatdo a) nos basta con buscar un árbol con peso mínimo que cubra todos los idiomas. Es decir tenemos que obtener un MST del grafo. Utilizando el algoritmo de Kruskal, podemos encontrarlo en tiempo $O(m \log m)$

Exercici 3 (2 punts) – Copistes –

Abans de la invenció de la impremta, era molt difícil fer una còpia d'un llibre. Tots els continguts havien de ser redactats a mà pels anomenats copistes. A un copista se li donava un llibre i, després de diversos mesos de treball, tornava una còpia del mateix. El temps que trigava era, molt probablement, proporcional al nombre de pàgines del llibre. La feina devia ser prou avorrida i per això podríem assumir que tots els copistes d'un monestir trigaven el mateix temps a copiar una pàgina.

El monestir de Pedralbes va decidir fer una còpia dels llibres de la seva biblioteca i, donat que no tenen copistes propis, han d'enviar els llibres a un altre monestir que sí que en tingui.

Podeu assumir que hi ha un total de n llibres per copiar i que el llibre i-ésim té p_i pàgines. A més, els llibres tenen un ordre predeterminat. Una vegada triat el monestir on es faran les còpies, s'hauran de repartir els llibres entre el seus m copistes. Cada llibre només se li pot assignar a un copista, i a cada copista només se li pot assgnar una seqüència contigua de llibres (d'acord amb l'ordre inicial del llibres). Amb aquesta forma d'assignar llibres a copistes es minimitza el temps de buidar i tornar a omplir la biblioteca. El temps necessari per fer la còpia total de la biblioteca queda determinat pel temps que necessita el darrer copista que finalitza la còpia dels llibres que se li han assignat.

El que no tenen molt clar els encarregats de la biblioteca és com fer l'assignació de llibres a copistes per garantir que el temps de còpia total de la biblioteca sigui el més curt possible. Ajudeu a aquests monjos i dissenyeu un algorisme eficient per a trobar l'assignació òptima dels n llibres als m copistes del monestir. Podeu assumir que coneixeu el temps t_p que necessita cadascun dels copistes per a copiar una pàgina.

Solució:

Todos los copistas copian a la misma velocidad, asumo que $t_p = 1$ y multiplicaremos por t_p al final.

Los libros no se pueden desordenar, una asignación a un copista será de todos los libros entre el i y el j, para $i \leq j$ adecuados.

Una solución óptima del problema es una asignación de un segmento inicial de libros a un copista seguida de una asignación óptima de los libros restantes a $m-1$ copistes. El coste es el máximo de los dos valores.

Vamos a calcular $T[i][k]$, el tiempo mínimo para copiar los libros $i, \ldots, n$ cuando disponemos de k copistas.

Utilizaremos $P[i][j]$, $1 \leq i \leq j \leq n$ que nos indica el número de páginas de los libros i a j.

A partir de la estructura de suboptimalidad obtenemos la siguiente recurrencia:

$$T[i][k] = \begin{cases} P[i][n] & \text{k=1,} \\ P[n][n] & \text{i=n,} \\ \min_{i<j\leq n}\{\max\{P[i][j], T[j][k-1]\}\} \end{cases}$$

Podemos precalcular P en tiempo $O(n)$ o precalcular $Q[i]$ el número de páginas de los libros 1 a i en tiempo $O(n)$. Así podemos utilizar $Q[j] - Q[i]$ en vez de $P[i][j]$, si $i < j$, y ahorrar algo de espacio y tiempo.

Para obtener la solución utilizaremos PD, calculando la tabla T de acuerdo con la recurrencia, empezando por la ultima columna. Como tenemos que obtener la solución, tambien guardaremos en una tabla auxiliar, para cada valor $[i][k]$ el valor j dónde se alcanza el mínimo.

Si precalculamos P, utilizando PD podemos obtener T en tiempo $O(n)$ por elemento, lo que nos da un coste de $O(n^2m)$ para todo el algoritmo.

Exercici 4 (2 punts) – Congrés –

Suposeu que esteu organitzant un congrés on els investigadors presentin els articles que han escrit. Els investigadors que vulguin presentar un article envien un document als organitzadors de la conferència. Els organitzadors de la conferència tenen accés a un comitè de revisors que estan disposats a llegir-ne com a molt R articles cadascun. Tots els articles enviats han de ser revisat per, com a mínim, A revisors.

El congrés té declarats un conjunt de temes. Cada enviament té assignat un tema concret i cada revisor té declarada una especialització per a un conjunt de temes. Els articles sobre un tema determinat només es revisen per part dels revisors experts en aquell tema.

Els organitzadors del congrés han de decidir (sempre que es pugui) quins avaluadors revisaran cadascun dels articles o, equivalentment, quins articles seran revisats per cada revisor. Es demana:

Proporcioneu un algorisme eficient per a resoldre aquest problema d'assignació.

Solució: Lo resolveremos como un problema de circulación en una red de flujo. La red tendrá un vértice por cada artículo $\{v_1, \ldots, v_n\}$ y un vértice por cada revisor $\{r-1, \ldots, r_m\}$. Añadimos las aristas (s, v_i) con cota inferior A y cota superior m, las aristas (r_j, t) con capacidad R y aristas (v_i, r_j) con capacidad 1, siempre que el revisor j sea experto en el tema del artículo v_i.

Una unidad de flujo que se transmita de s a t, pasará a través de un camino $s \to v_i \to r_j \to t$ representando que el trabajo i se asigna al revisor j.

La capacidad 1 en las aristas (v_i, r_j) garantiza que no se asigna un artículo más de una vez a un revisor. La capacidad R en las aristas (r_j, t) evita que se asignen más de R artículos a un revisor. La cota inferior en las otras aristas garantiza que cada artículo tiene al menos A revisiones. La red admite una circulación si y solo si el problema planteado tiene solución.

Sea $t(j)$ el número de temas en el que es experto el revisor j y T la suma de los valores de $t(j)$. La red tiene $N = n + m + 2$ nodos Y $M = n + T + m$ aristas.

El flujo máximo está acotado por mR, así el coste será $O(mR(n + m + T))$ si utilizamos la cota superior al flujo.