Ampliació d'Algorísmia **QP 2023-2024**

Llista de possibles problemes per al 1er parcial **8 d'abril de 2023**

1. *Conjunt dominant* En un graf no dirigit $G = (V, E)$, diem que $D \subseteq V$ és un conjunt dominant en G si per cada vèrtex $u \in V$, $u \in D$ o u és adjacent a un vèrtex $v \in D$, $(u, v) \in E$. Definim el problemes següents:

   - ConjuntDominantDeMida: Donats un graf no dirigit $G = (V, E)$ i un natural b, decidir si existeix un un conjunt dominant D en G tal que $|D| \leq b$.
   - ConjuntDominant: Donat un graf no dirigit $G = (V, E)$, calcula un conjunt dominant D de mida o cardinalitat mínima.

   Demostreu que:

   (a) Demostreu que ConjuntDominantDeMida és NP-complet.

   (b) Demostreu que ConjuntDominantDeMida $\in$ P si i només si Conjunt Dominant $\in$ FP (computable en temps polinòmic).

2. *Millor Resposta.* Recordem els jocs de creació de xarxes (NCG) introduïts per Fabrikant et al. Un joc Γ es defineix per un parell $\Gamma = \langle V, \alpha \rangle$ on $V = \{1, \ldots, n\}$ és el conjunt de jugadors (o nodes de la xarxa) i α el cost d'establir un enllaç. Cada node $u \in V$ pot establir enllaços a qualsevol dels altres nodes. Una estratègia del jugador u és un subconjunt $s_u \subseteq V - \{u\}$ indicant els enllaços que u ha comprat. Un *vector d'estratègies* per Γ és una tupla $s = (s_1, \ldots, s_n)$ on per cada $u \in V$, s_u és l'estratègia del jugador u. A cada vector d'estratègia s li correspon un *outcome graph*, un graf no dirigit definit per $G[s] = (V, E)$ amb $E = \{(u, v) | (u \in s_v) \vee (v \in s_u)\}$.

   El cost d'un jugador u depèn de les estratègies de tots els jugadors i es defineix de la manera següent: $c_u(s) = \alpha |s_u| + \sum_{v \in V} d_{G[s]}(u, v)$.

   Considerem ara el problema MillorResposta:

   > Donats $\Gamma = \langle V, \alpha \rangle$, un vector d'estratègia $s = (s_1, \ldots, s_n)$ i un jugador u,
   >
   > calcula una estratègia s_u^* per al jugador u de manera que que *no hi ha cap estratègia s'_u tal que $c_u(s_{-u}, s'_u) < c_u(s_{-u}, s_u^*)$*.

   Demostreu que si el problema MillorResposta fos computable en temps polinòmic aleshores P = NP.

3. *Random Selection.*

   (a) Definim la funció Select de la manera següent:

   > Donats un conjunt $S = \{a_1, \ldots, a_n\}$ de n nombres i un nombre $k \in \{1, \ldots, n\}$,
   >
   > retorna el k-èsim element de S, si enumeréssim per ordre creixent els seus elements.

   Demostreu que hi ha un algorisme aleatori que computa Select(S, k) amb un temps esperat $O(|S|)$.

(b) Considerem ara la funció Median:

> Donat un conjunt S de n nombres,
>
> retorna el nombre de S que estaria en la posició del mig, si enumeréssim per ordre creixent els seus elements.

Utilitzant l'algorisme aleatori per a Select, demostreu que hi ha un algorisme aleatori que calcula $\mathsf{Median}(S)$ en un temps esperat $O(|S|)$.

4. *The Contraction Algorithm.* Un *cut-set* d'un graf no dirigit $G = (V, E)$ és un subconjunt d'arestes $C \subseteq E$ tals que si les esborrem d'E, el graf resultant $(V, E - C)$ conté 2 o més components connexes. Un *global min-cut* o *min-cut* (depèn de les fonts bibliogràfiques) és un cut-set de cardinalitat mínima. Fixeu-vos que la cardinalitat d'un min-cut d'un graf G és el mínim nombre d'arestes que cal esborrar per a desconnectar G.

   Presenteu el *Contraction Algorithm* conegut també per *Karger's Algorithm* i analitzeu-ne el temps de computació i la probabilitat d'error.

5. *El sistema criptogràfic* RSA. Diem que el sistema RSA és fàcilment vulnerable quan donada la clau pública i un missatge codificat, aquest es pot decodificar en temps polinòmic.

   (i) Demostreu que si $\mathsf{P} = \mathsf{NP}$, aleshores els sistema RSA seria fàcilment vulnerable.

   (ii) I si tinguéssim manera de vulnerar fàcilment el sistema RSA, això implicaria que $\mathsf{P} = \mathsf{NP}$?

6. *Espiant* RSA. Suposeu que en el sistema RSA l'espia Eve aconsegueix (N, d), la clau privada d'Alice. La clau publica d'Alice és (N, e) amb $e = 3$. Demostreu que per aquesta clau pública, l'espia Eve pot calcular eficientment la factorització de N.