# Algoritmos y Estructuras de Datos III - 2$^{\underline{\text{do}}}$ Recuperatorio
**Fecha de examen: 7-DIC-2022**

| | Nº Orden | Apellido y nombre | | L.U. | # hojas [1] |
|---|---|---|---|---|---|
| | | | | | |
| Notas: | Ej1 | Ej2 | Ej3 | Ej4 | Final |
| | | | | | |

**Aclaraciones**: El parcial se aprueba con 2 (dos) ejercicios aprobados. Cada hoja debe estar numerada y debe tener el número de orden y L.U. El parcial dura 4 horas y es a libro abierto.

1) A Juli le gusta andar en bicicleta por la ciudad y es su medio preferido de transporte. Sin embargo, a veces las distancias que debe recorrer son demasiado largas y se le complica llegar a tiempo. Es por eso que a veces utiliza el tren para desplazarse entre distintos puntos de la ciudad, que le permite viajar 4 veces más rápido: en vez de promediar 15km/h se mueve a 60km/h. Sin embargo, entre que llega a la estación de tren y espera a que llegue, siempre le suma unos 10 minutos para subirse al tren[2]. Una vez que está en este tren, ese tiempo no se suma en cada estación[3]. Vamos a modelar la ciudad con un grafo. Sean $P$ distintos puntos de la ciudad y $C$ los caminos entre los puntos, el grafo $G = (P, C)$ modela la ciudad y la función $\text{dist}(u, v)$ con $u, v \in P$ representa la longitud en kilometros entre dos puntos de la ciudad. El tren se modela con un grafo $T = (P', C')$ con $P'$ incluido en $P$ y $C'$ son caminos que unen puntos de $P'$ usando vias del tren.

   a) Proponer un algoritmo basado en grafos que dada la lista de puntos $P$ de la ciudad, la lista $C$ de caminos, la lista $C'$ de caminos del tren, y dos puntos $A$ y $B$ determine un camino de $A$ a $B$ que minimice el tiempo que tarda en llegar de $A$ a $B$ considerando los tiempos de espera al subirse al tren. El algoritmo debe tener complejidad $O(|P|^2)$.
   b) Justificar la correctitud del algoritmo.

2) Sea $S$ un sistema de restricciones de diferencias formado por un conjunto de variables $x_1, \ldots, x_n$ y un conjunto de desigualdades $E$ de la forma $e_{ij} \colon x_i - x_j \leq c_{ij}$, y sea $m = \min\{c_{ij} \mid e_{ij} \in E\}$. Demostrar que $S$ admite una solución si y solamente si $S$ admite una solución donde el valor asignado a $x_i$ pertence al intervalo $[0, |m|(n-1)]$ para todo $1 \leq i \leq n$.

3) Luego de vencer en la primera edición de GH (Grupo Humano), Tuki se volvió parte del staff que organiza la competencia. Específicamente, Tuki ahora es parte del equipo que selecciona los participantes de la casa.

   Para la nueva edición ya se tienen preseleccionadas a $n$ personas, pero solo $k < n$ de estas van a ser parte del concurso. A estos $n$ candidatos se los separaron en $k$ grupos disjuntos $G_j$, y se le asignó a Tuki la tarea de elegir un miembro de cada grupo, buscando garantizar algunos requisitos del programa. Puntualmente, se quiere que exactamente la mitad de los elegidos[4] tengan más de $T$ años, y que dentro de los integrantes jóvenes (es decir, los que tienen una edad menor o igual a $T$ años), no haya ninguna localidad a la cual pertenezcan más de la mitad de ellos.

   Dada la descripción de los $k$ grupos $G_j$, con $1 \leq j \leq k$, y la edad $e_i$ y localidad de origen $l_i$ de cada uno de los candidatos $1 \leq i \leq n$, Tuki debe encontrar una asignación que cumpla los requisitos del programa. En caso de que no la hubiese, también debe informarlo.[5]

   a) Modelar el problema como un problema de flujo máximo. Justificar su correctitud.
   b) Proponer un algoritmo que resuelva el modelo propuesto.

[1] Incluyendo esta hoja.
[2] 10 minutos son 1/6 de hora.
[3] Si se baja del tren, toma un camino en bici y se toma otro tren, debe volver a sumar ese tiempo de espera.
[4] El valor de $k$ es par.
[5] Por simplicidad, las localidades $l_i$ serán representadas con números de 1 a $n$.

c) ¿Qué complejidad tiene el algoritmo propuesto? Dar la cota más ajustada posible en función de los parámetros que recibe Tuki.

4) Considerar el problema IndivisibleEnIndependientes (IEI), definido como: "Dado un grafo $G$ y un natural $k > 0$, retorna **sí** si y sólo si es imposible particionar $G$ en conjuntos independientes de vértices tales que todos tengan tamaño por lo menos $k$. Probar la pertenencia de IEI en la clase coNP-Completo, mostrando que

   a) pertenece a la clase coNP, y

   b) que el problema es completo para esa clase de complejidad

   **Ayuda:** Utilizar el hecho de que 3-coloreo es NP-Completo y elegir un $k$ que, conociendo el tamaño $n$ de un grafo arbitrario $G$, permita armar un grafo $G'$ de $3k$ vértices que contenga a $G$.