

17. En medio de una pandemia, la Escuela de Aulas Grandes y Ventiladas quiere implementar un protocolo especial de distanciamiento social que tenga en cuenta que la escuela no tienen restricciones de espacio. El objetivo es separar a cada curso en dos subcursos a fin de reducir la cantidad de pares de estudiantes que sean muy cercanos, dado que se estima que estos estudiantes tienen dificultades para respetar tan buscado distanciamiento. Para este fin, en el protocolo se estableció que cada curso que tenga  $c$  parejas de estudiantes cercanos tiene que dividirse en dos subcursos, cada uno de los cuales puede tener a lo sumo  $c/2$  parejas de estudiantes cercanos. Notar que no importa si un subcurso queda con más estudiantes que otro.

Formalmente, para cada curso contamos con un conjunto de estudiantes  $E$  y su conjunto  $C$  de pares de estudiantes cercanos. Luego, una partición  $(A, B)$  de  $E$  es una *solución factible* para  $(E, C)$  cuando  $|(A \times A) \cap C| \leq |C|/2$  y  $|(B \times B) \cap C| \leq |C|/2$ . Por ejemplo, si  $E = \{1, 2, 3, 4\}$  y  $C = \{1-2, 2-3, 3-4\}$ , entonces  $(\{1, 3, 4\}, \{2\})$  y  $(\{2, 4\}, \{1, 3\})$  son soluciones factibles.

- a) Especificar el problema descrito definiendo cuál es la instancia (i.e. cuáles son los datos de entrada y qué condiciones satisfacen) y cuál es el resultado esperado (i.e., cuáles son los datos de salida y qué condiciones satisfacen).

Data de entrada:  $\text{Conj } E/E \neq \emptyset, \text{ Conj } C/\forall (s,t) \in C \rightarrow s \in E \text{ y } t \in E \text{ con } s \neq t \text{ y } (s,t) \in C \rightarrow (t,s) \notin C \text{ y } \#C \leq \binom{|E|}{2}$

Data de salida:  $\text{Conj } A/|A \times A \cap C| \leq |C|/2 \text{ y } \text{Conj } B/|B \times B \cap C| \leq |C|/2$

- b) Demostrar que para toda instancia existe un resultado esperado que satisface las condiciones definidas por el protocolo. **Ayuda:** hacer inducción en la cantidad de estudiantes. Para el paso inductivo, considerar que si **les estudiantes se asignan iterativamente a los subcursos, entonces conviene enviar a cada estudiante al subcurso que tenga la menor cantidad de estudiantes cercanos a él.**

- c) A partir de la demostración del inciso anterior, diseñar un algoritmo que encuentre una solución factible en tiempo lineal en función del tamaño de la entrada definido en el inciso a).

Estrategia greedy: dado  $E$  voy a tomar un estudiante y voy a agregarlo al qto donde tenga la menor # de parejas cercanas a él

Ej-  $E = \{1, 2, 3, 4\}$   $C = \{1-2, 1-3, 1-4, 2-3, 3-4\} \rightarrow \#C = \binom{4}{2}$

$\begin{matrix} 1 & 2 \\ \boxed{1} & \boxed{2} \\ \boxed{3} & \boxed{4} \end{matrix}$

Tamaño de la entrada es  $|E| + |C| \rightarrow$  necesito un algo que lo haga en  $O(|E| + |C|)$

Veremos otro ej  $E = \{1, 2, 3, 4, 5, 6, 7\}$   $C = \{1-2, 1-3, 2-3, 4-5, 5-6, 6-7\}$

2	1	2	1
2	3	5	1
4	4	7	1
2	6		
	A	B	

Def  $P(E, C)$

$A, B$ ; (vacíos)

$D$  = desamos  $C$  en un dice / donde  $\in E$  y significado = elemento de  $E$  son pareja ordenada  $O(|C|)$

for  $i = 0; i < |E|; i++$   $\Theta(|E|)$

    countA = 0; countB = 0;

    for  $j = 0; j < |A|; j++$

        if  $E[i] \in A[j]$  then

            countA ++

        endif

    for  $j = 0; j < |B|; j++$

        if  $E[i] \in B[j]$  then

            countB ++

        endif

    if countA > countB

        Agregar( $E[i]$ );

    else

        B.agregar( $E[i]$ )

return A, B;

$\text{DirP}(\text{vector } E, \text{vector } \langle \text{type} \rangle C) \{$   
 $\text{vector } A, \text{vector } B,$