Primer Clasificatorio OIE 2021

Soluciones a los problemas

XXV OIE 2021

19 de febrero 2020

Autores de problemas:

- Félix Moreno Peñarrubia
- Izan Beltrán Ferreiro

Equipo de preparación:

- Alberto Maurel Serrano
- Blanca Huergo Muñoz
- Javier Nistal Salas
- Jordi Guillem Rodríguez Manso
- Max Balsells i Pamies
- Miquel Ortega Sánchez-Colomer
- Pablo Hidalgo Palencia

Estadísticas

- Número de participantes (con al menos un envío): 64
- Número de envíos: 613
- Puntuación máxima: 383
- Puntuación séptima posición: 358
- Puntuación mediana: 157

¡Hola!

Enunciado

Dada una palabra, decir si es anagrama de *hola*.

Autor: Félix Moreno Peñarrubia

- Número de soluciones con 100 puntos: 61
- Primera solución con 100 puntos: Clement Genninasca, 00:02:10.

¡Hola!

Solución

Para determinar en general si dos palabras s, t formadas por carácteres de un alfabeto $\mathcal{A}$ son anagramas, hay que contar las frecuencias de cada carácter en las dos palabras y decir si son iguales. Se puede hacer en tiempo $\mathcal{O}(|s| + |t| + |\mathcal{A}|)$ iterando una vez por cada palabra almacenando las frecuencias de los caracteres en una tabla que se actualiza por cada carácter incrementando en uno el valor correspondiente.

Alternativa

Una forma con poco código de resolver el problema era ordenar la palabra dada y compararla con `ahlo`.

Cerrado por la resta

Enunciado

Decimos que un conjunto de enteros S es *cerrado por la resta* si para todo $a, b \in S, a \neq b$ se cumple que $(a - b) \in S$ o $(b - a) \in S$.
Dado un conjunto S, decir si es cerrado por la resta.

Autor: Félix Moreno Peñarrubia

- Número de soluciones con 100 puntos: 7
- Número de soluciones con 65 puntos: 17
- Número de soluciones con 40 puntos: 17
- Primera solución con 100 puntos: Darío Martínez Ramírez, 00:42:48

Cerrado por la resta

Primeras observaciones

- Si hay enteros positivos y negativos en el conjunto, no puede ser cerrado por la resta.
  - **Demostración** Sean $a > 0$ y $b < 0$ los valores máximo y mínimo del conjunto, respectivamente. Entonces $a - b > a$ y por tanto no puede pertenecer al conjunto por ser mayor al máximo. Análogamente $b - a < b$ tampoco puede pertenecer al conjunto.

Cerrado por la resta

Primeras observaciones

- Si hay enteros positivos y negativos en el conjunto, no puede ser cerrado por la resta.
- Si $0 \in S$, la respuesta es la misma que para el conjunto $S \setminus \{0\}$

Cerrado por la resta

Primeras observaciones

- Si hay enteros positivos y negativos en el conjunto, no puede ser cerrado por la resta.
- Si $0 \in S$, la respuesta es la misma que para el conjunto $S \setminus \{0\}$

A partir de ahora podemos asumir que nuestro conjunto es de enteros positivos (si es de negativos cambiamos el signo a todos y si hay 0 lo quitamos).

Cerrado por la resta

Primeras observaciones

- Si hay enteros positivos y negativos en el conjunto, no puede ser cerrado por la resta.
- Si $0 \in S$, la respuesta es la misma que para el conjunto $S \setminus \{0\}$

Observación clave

Un conjunto de n enteros positivos cerrado por la resta es de la forma $S = \{i \cdot m | i = 1, \ldots, n\}$ para algún entero positivo m.

Demostración

Sean $x_1, x_2, \ldots x_n$ los elementos del conjunto ordenados de menor a mayor. Tenemos que $x_2 - x_1 \in S$, y $x_2 - x_1 < x_2$, por tanto $x_2 - x_1 = x_1$ y $x_2 = 2 \cdot x_1$. Después tenemos que $x_3 - x_1 \in S$. $x_1 = x_2 - x_1 < x_3 - x_1 < x_3$, por tanto $x_3 - x_1 = x_2$. Usando varias veces más la desigualdad $x_{i-2} = x_{i-1} - x_1 < x_i - x_1 < x_i$ para concluir que $x_i - x_1 = x_{i-1}$, tenemos el resultado.

Cerrado por la resta

Primeras observaciones

- Si hay enteros positivos y negativos en el conjunto, no puede ser cerrado por la resta.
- Si $0 \in S$, la respuesta es la misma que para el conjunto $S \setminus \{0\}$

A partir de ahora podemos asumir que nuestro conjunto es de enteros positivos (si es de negativos cambiamos el signo a todos y si hay 0 lo quitamos).

Observación clave

Un conjunto de n enteros positivos cerrado por la resta es de la forma $S = \{i \cdot m | i = 1, \ldots, n\}$ para algún entero positivo m.

Solución

Para comprobar si el conjunto de enteros positivos es cerrado se calcula el mínimo m y se comprueba que todos los elementos sean divisibles por m y $\leq n \cdot m$ (que sean diferentes ya viene garantizado). Complejidad $\mathcal{O}(n)$.

Cuadrados en la libreta

Enunciado

Hay m rectas horizontales con coordenadas y $1, 2, \ldots m$ y n rectas verticales con coordenadas x $a_1, a_2, \ldots, a_n$. Determinar cuántos cuadrados hay con vértices en intersecciones entre rectas.

Autor: Félix Moreno Peñarrubia

- Número de soluciones con 100 puntos: 1
- Número de soluciones con 71 puntos: 1
- Número de soluciones con 53 puntos: 8
- Número de soluciones con 42 puntos: 8
- Número de soluciones con 37 puntos: 3
- Número de soluciones con 26 puntos: 7
- Número de soluciones con 11 puntos: 4
- Primera solución con 100 puntos: Leonardo Costa Lesage, 01:02:41

Cuadrados en la libreta

Observación

El número de cuadrados cuyos lados verticales están en la i-ésima y j-ésima recta $(i < j)$ es:

$$f(i,j) = \begin{cases} m - (a_j - a_i) & \text{si } (a_j - a_i) \leq m \\ 0 & \text{si } (a_j - a_i) > m \end{cases}$$

Por tanto, el número total de cuadrados es:

$$\sum_{i=1}^{n} \sum_{j=i+1}^{n} f(i,j) = \sum_{i=1}^{n} \sum_{\substack{j>i \\ a_j \leq a_i + m}} (m - (a_j - a_i)) =$$

$$= \sum_{i=1}^{n} \sum_{j=i+1}^{k_i} (m - (a_j - a_i)) = \sum_{i=1}^{n} \left((k_i - i)(m + a_i) - \sum_{j=i+1}^{k_i} a_j \right)$$

Donde k_i es el mayor índice que satisface $a_{k_i} \leq a_i + m$.

Cuadrados en la libreta

Solución

Como k_i va aumentando conforme i aumenta, si vamos incrementando i desde 1 hasta n podemos ir calculando k_i al mismo tiempo en coste lineal total usando la técnica de la ventana deslizante (es decir, cada vez que incrementamos i probamos a incrementar k_i hasta que llegamos al valor adecuado, haciendo en total dos recorridos por la secuencia).

Para el término $\sum_{j=i+1}^{k_i} a_j$, si tenemos calculadas la sumas parciales $s_j = \sum_{i=1}^{j} a_i$, entonces $\sum_{j=i+1}^{k_i} a_j = s_{k_i} - s_i$, por tanto podemos calcularlo en tiempo constante para cada i.

En total, calculamos la suma con complejidad $\mathcal{O}(n)$.

Conectando pueblos

Enunciado

Dado un grafo con pesos en las aristas, encontrar el mínimo H tal que el subgrafo con aristas con pesos $\leq H$ es conexo.

Autor: Izan Beltrán Ferreiro

- Número de soluciones con 100 puntos: 9
- Primera solución con 100 puntos: Victoria Durán, 00:44:57

Conectando pueblos

Solución

Si ordenamos las aristas por pesos, es fácil ver que existe un número k tal que usando las primeras k aristas el grafo es conexo. La forma de ir añadiendo las aristas y comprobando de forma eficiente si el grafo es conexto es usando un Disjoint-Set (también conocido como DSU, MF-Set o Union-Find).
El Disjoint-Set nos permite tener inicialmente n nodos separados y ir haciendo operaciones "merge", en las que dos conjuntos de nodos se unen para formar un nuevo conjunto mayor. Lo que haremos será que cuando pasamos por una arista, haremos la operación "merge" entre los conjuntos correspondientes a los nodos de sus extremos.

Conectando pueblos

Solución

Si además guardamos el tamaño de los conjuntos (que es muy sencillo de mantener), simplemente podemos mirar tras cada operación si el conjunto correspondiente a cualquier nodo tiene tamaño n. Si esto pasa, es que todos los nodos pertenecen al mismo conjunto.
Esta solución tiene coste $\mathcal{O}(m \log m)$

Conectando pueblos

Solución alternativa

Una observación útil es que si para un valor H, si el grafo con las aristas de peso $w \leq H$ es conexo, entonces también lo será para los valores $H+1, H+2, \ldots$, es decir, para cualquier valor mayor.
Esto nos permite hacer una búsqueda binaria sobre H.
Para verificar si una H nos da un grafo conexo, podemos simplemente hacer un DFS añadiendo un `if` que verifica que solo pasa por aristas de peso $w \leq H$.

Conectando pueblos

Solución alternativa

Ahora que sabemos verificar una H particular, vamos a decir que ponemos un 0 al valor H si el grafo correspondiente no es conexo, y 1 si lo es.
Si miramos $H = 1, 2, 3, ...$, nos saldrán unos valores de esta forma:
`000...00001111...111`
Es decir, será 0 hasta que en un cierto punto pasará a dar 1, es por esto que podemos hacer una búsqueda binaria.
Esta solución tiene coste $\mathcal{O}(n \log W)$

Tablero hermoso

Enunciado

Decir si es posible pintar un tablero $n \times m$ de colores blanco y negro tal que se satisfagan las dos condiciones:

- El número de casillas de color blanco es igual al número de casillas de color negro.
- El número de pares de casillas adyacentes del mismo color es igual al número de pares de casillas adyacentes de color distinto.

y en caso afirmativo construir una coloración válida.

Autor: Félix Moreno Peñarrubia

- Número de soluciones con 100 puntos: 0
- Número de soluciones con 15 puntos: 7
- Primera solución con 15 puntos: Maria Elena Sánchez Grümmer, 02:07:07

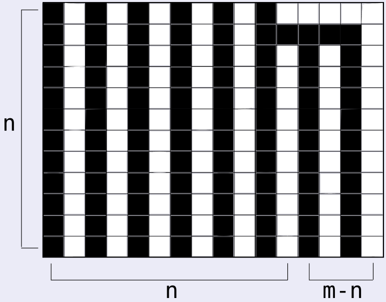
Tablero hermoso

Contando casillas y pares adyacentes

- Número de casillas: $n \cdot m$.
- Número de pares adyacentes de casillas: $(n-1) \cdot m + n \cdot (m-1)$.

Ambos números deben ser pares $\implies$ n, m deben ser pares.

Una posible solución



Coloreando la rejilla

Enunciado

Hay n segmentos horizontales $s_1, s_2, \ldots s_n$. s_i tiene uno de sus extremos en el punto $(0, i)$ y el otro en (a_i, i). Hay m segmentos verticales $t_1, t_2, \ldots t_m$. t_j tiene uno de sus extremos en el punto $(j, 0)$ y el otro en (j, b_j). Calcular el número de componentes conexas de la unión de todos estos segmentos.

Autor: Félix Moreno Peñarrubia

- Número de soluciones con 100 puntos: 0
- Número de soluciones con 67 puntos: 4
- Número de soluciones con 46 puntos: 8
- Primera solución con 67 puntos: Pablo Sáez Reyes, 2:57:34

Coloreando la rejilla

Observación

Mientras vamos recorriendo la secuencia a, podemos ir manteniendo cuál es el b_j con menor j tal que $b_j \geq i$ (donde i es el índice actual de la secuencia a que visitamos). Este es el segmento vertical más a la izquierda que es candidato a intersecar con el segmento i.

Envolvente

Para la secuencia $a_1, \ldots, a_n$ calculamos la *subsecuencia envolvente* $a_{s_1}, \ldots, a_{s_k}$ definida de esta forma:

- $s_1 = 1$.
- s_{i+1} es el menor índice $> s_i$ tal que $a_{s_{i+1}} > a_{s_i}$.

Esta subsecuencia se puede calcular en $\mathcal{O}(n)$ y nos es útil porque satisface la siguiente propiedad (para la secuencia a de segmentos horizontales, para la b es igual): Si un segmento vertical interseca algún segmento horizontal, entonces interseca un segmento horizontal de la envolvente. Tener calculadas las envolventes de las dos secuencias facilita la implementación de algunas soluciones.

Coloreando la rejilla

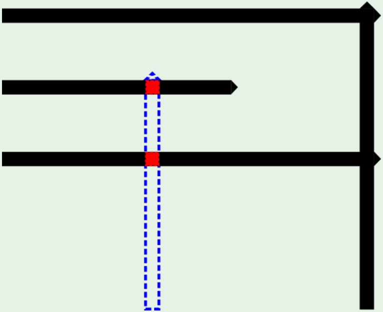
Solución, primera parte

Lo primero que hay que hacer es detectar los segmentos que no intersecan con ningún otro segmento y por tanto constituyen una componente conexa ellos mismos.
Para ello, vamos manteniendo el candidato a intersecar de la envolvente de la otra secuencia mientras iteramos por cada una de las secuencias y sabemos que si un segmento no interseca con ese candidato, entonces no intersecará con ningún otro segmento. Por tanto, encontramos todos los segmentos en $\mathcal{O}(n + m)$.
Una vez hemos encontrado estos segmentos, los eliminamos, es decir, a partir de ahora no consideramos que estén (Esto en cuanto a la explicación de la solución; según la implementación de la siguiente parte, puede no ser necesario eliminar explícitamente los segmentos).

Coloreando la rejilla

Observación

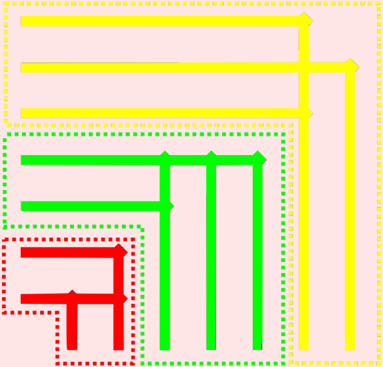
Si todos los segmentos intersecan con algún otro segmento, se satisface la siguiente propiedad:



Si dos segmentos intersecan un segmento, entonces todos los segmentos entre los dos pertenecen a la misma componente conexa que los tres segmentos originales.

Observación clave

Esto hace que las componentes conexas tengan forma de "L invertida"



Coloreando la rejilla

Solución, segunda parte

Para calcular las componentes conexas de esta forma, hacemos lo siguiente.

Iteramos por la secuencia a, manteniendo el candidato b_j de la envolvente y manteniendo también cuál es el valor máximo de un a_i que hemos visto hasta ahora. En el momento que el valor de j del candidato sea mayor al valor máximo visto, estamos en una nueva componente conexa.