Soluciones a los problemas

OIE 2020

17 de abril 2020

Día 1: Ataques DDOS

Enunciado

Dada una serie de consultas consultas de la forma (IP, tiempo, tamaño) debes ignorarlas si el conjunto de paquetes con esta misma IP en las últimas δ unidades de tiempo tiene al menos α elementos o una suma de tamaños de al menos β.
Los parámetros α, β y δ pueden cambiar tras cualquier consulta.

Autor: Izan Beltrán Ferreiro

- Número de soluciones con 100 puntos: 6
- Número de soluciones con 39 puntos: 1
- Número de soluciones con 18 puntos: 5
- Primera solución con 100 puntos: 0:52, Ignasi Segura

Día 1: Ataques DDOS

Solución $\mathcal{O}(Q \log(Q))$

Digamos que los paquetes son de la forma (IP_i, t_i, b_i), donde t_i es el tiempo de i-ésimo paquete y b_i su tamaño en bytes.
Para cada IP mantenemos un vector con pares (t_j, p_j) según van llegando nuevos paquetes de esta IP, donde t_j es el tiempo del j-ésimo paquete con esta IP y p_j es la suma de tamaños de todos los paquetes hasta ese momento *(prefix sum)*, es decir, si el primer paquete para una IP es (IP_k, t_k, b_k), en el vector insertamos el par (t_k, b_k) y para los siguientes de paquetes, si en el vector hay n pares anteriores, entonces insertamos el par $(t_k, p_{n-1} + b_k)$
...

Día 1: Ataques DDOS

Solución $\mathcal{O}(Q \log(Q))$

Para consultar si este paquete debe ser ignorado podemos hacer una búsqueda binaria sobre este vector respecto al tiempo, es decir, entre los n pares en el vector buscamos la máxima j tal que $t_j \leq t_k - \delta$. En caso de que no exista tal índice diremos que $j = -1$. Definimos $p_{-1} = 0$ por comodidad.

Entonces sólo deberíamos considerar los paquetes $j+1, j+2, \ldots, n-1$.

El número de paquetes considerados es $A = n - j - 1$.

La suma de tamaños de paquetes considerados es $B = p_{n-1} - p_j$

Finalmente ignoramos el paquete si $A \geq \alpha \vee B \geq \beta$.

Día 1: Cubriendo agujeros

Enunciado

Dados n segmentos en una recta, Hay que cubrir estos segmentos con otro conjunto S de segmentos que la suma de las distancias sea mínima.
Se tenía que dar la respuesta para cuando S tenía $1, 2, ..., n$ segmentos.

Autor: Cesc Folch Aldehuelo

- Número de soluciones con 100 puntos: 14
- Número de soluciones con 53 puntos: 1
- Primera solución con 100 puntos: 0:14, Javier Nistal Salas

Día 1: Cubriendo agujeros

Solución $\mathcal{O}(n \log(n))$

- Si usamos k segmentos, serán segmentos de la forma $[a_0, b_{i_1}], [a_{i_1+1}, b_{i_2}], [a_{i_2+1}, b_{i_3}], \ldots, [a_{i_{k-1}}, b_n]$
- La suma de las longitudes de estos segmentos es $suma = (b_n - a_0) - \sum_{j=1}^{k-1} (a_{i_j+1} - b_{i_j})$
- Para minimizar *suma* tenemos que maximizar $\sum_{j=1}^{k-1} (a_{i_j+1} - b_{i_j})$.
- Si consideramos todos los $n-1$ enteros $(b_{i+1} - a_i)$, para minimizar *suma* para k elementos tenemos que quedarnos con los $k-1$ elementos más grandes de estos.
- Esto se puede conseguir ordenando un vector.

Día 1: Secuencias VonitAs

Enunciado

Dada una secuencia de enteros, cual es el mínimo de enteros que tenemos que modificar para que la secuencia sea primero decreciente y luego creciente o al revés, primero creciente y luego decreciente.

Autor: Maria Prat Colomer y Cesc Folch Aldehuelo

- Número de soluciones con 100 puntos: 1
- Número de soluciones con 61 puntos: 5
- Primera solución con 100 puntos: 3:00, Oscar Balcells Obeso

Día 1: Secuencias VonitAs

Solución $\mathcal{O}(n \log(n))$

- Supondremos que primero es decreciente y luego crecient, el otro caso es simétrico
- Se puede ver que el elemento más pequeño de la secuencia final no se ha modificado
- Para cada elemento podemos calcular cual es el coste de hacer la secuencia VonitA donde este elemento no se modifique y se el más pequeño
- Esto se puede ver como calcular la secuencia creciente más larga (LIS) desde este punto hasta los dos extremos, estos serán los elementos que no hay que modificar
- Haciendo una sola iteración del algoritmo para calcualar LIS desde cada uno de los extremos podemos encontrar el resultado en cada punto.

Día 1: Tazos de Pokémon

Enunciado

Dado un grafo dirigido donde cada vértice tiene exactamente una arista de salida, tenemos que ir pasando los tazos de cada vértice al vértice que apunta dejando exactamente uno a cada vértice del camino (tantas veces como aparezca).

Autor: Cesc Folch Aldehuelo

- Número de soluciones con 100 puntos: 0
- Número de soluciones con 17 puntos: 1
- Número de soluciones con 11 puntos: 3
- Primera solución con 17 puntos: 3:57, Javier Nistal Salas

Día 1: Tazos de Pokémon

Solución $\mathcal{O}(n \log(n)^2)$

- Vamos a tratar las componentes conexas por separado
- La primera observación es que cada componenete conexa tiene exactamente un ciclo.
- El primer paso era detectar el ciclo de la componente y todos sus vértices.
- De cada uno de esos vértices cuelga un arbol, tenemos que propagar la información de todos los elementos del arbol hasta su vértice correspondiente en el ciclo. Dejando los tazos que tocan por el camino.

Día 1: Tazos de Pokémon

Solución $\mathcal{O}(n \log(n)^2)$

- Una vez en el ciclo, podemos actualizar los resultados de los elementos del ciclo usando un arbol de segmentos para hacer las actualizaciones eficientemente.
- Para propagar la información de los arboles se tenía que ir propagando de abajo a arriba y en cada vértice, al juntar la información de los que lo señalaban, se tenía que hacer de forma eficiente.

Día 1: Pintando piedras 2

Solución $\mathcal{O}(n)$

- Llamaremos S_i la solución para i piedras.
- S_i cuando $0 \leq i < k$ es $S_i = c^i$, porque no se pueden conseguir k piedras consecutivas.
- Cuando $i = k$, tenemos que $S_k = c^k - c$.
- Ahora, si $i > k$, Definimos $S'_i = S_{i-1} * c$.
- S'_i son todas las formas de pintar sin k consecutivas, y con exactamente k consecutivas iguales al final.
- Para pintar $i > k$ piedras sin k piedras consecutivas iguales, excepto las k últimas que son iguales, se puede hacer de $S_{i-k} * (c - 1)$ maneras.
- Entonces tenemos la recurrencia para $i > k$ de $S_i = (S_{i-1} * c) - (S_{i-k} * (c - 1))$.

Día 1: Pintano piedras 2

Enunciado

Dadas n piedras en fila, ¿de cuántas formas podemos pintarlas de c colores sin que k piedras consecutivas sean del mismo color?

Autor: Cesc Folch Aldehuelo

- Número de soluciones con 100 puntos: 0
- Número de soluciones con 24 puntos: 2
- Número de soluciones con 11 puntos: 3
- Primera solución con 24 puntos: 3:54, Leonardo Costa Lesage