

Programación Competitiva

Competencia #4



acm International Collegiate
Programming Contest

Problema #1: Kefa and Park

- Iterar sobre el árbol, teniendo un counter de cuántos gatos se han encontrado. Si se encuentran m consecutivos, dejar de iterar.
- $O(N)$



Problema #2 Two Buttons

- Si te pasas del número que quieres llegar al multiplicarlo, le restas uno. A lo mucho vas a requerir $2*m$ estados.
- **$O(\max(2m, n))$**



Problema #3 Destroying Roads

- Pre calcular todas las distancias de entre todos los nodos.
- La respuesta será el mínimo entre la distancia entre s_1 y t_1 + la distancia de s_2 y t_2 , o checar si tienen un camino en común.
- **$O(N^2 \text{Log} n)$** precalcular distancias con dijkstra

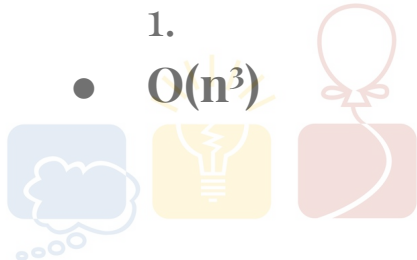


- Para cada par de nodos i, j , supone que ese es el camino común, y checar que la distancia mínima sea menor que l_1 y l_2 .
- **$O(N^2)$**

Problema #4 Ping pong

- Para query de tipo 1, verificar si se puede mover a todos los nodos pasados, y si sí puede generar una edge bidireccional en el grafo.
- Para query de tipo 2, iterar todo el grafo, y verificar si se puede visitar el nodo 2 desde el nodo 1.

- $O(n^3)$



Problema #5 Cthulhu

- Verificar que exista un solo ciclo en el grafo, y si existe un solo ciclo decir sí.
- Ese tipo de estructura se llama cactus.
- Resolver con tarjan's strongly connected components.
- $O(N)$



Problema #6 Leha and another game about graphs

- Hacer un dfs y visitar todos los hijos, sin visitar los que ya fueron visitados. El dfs regresa el estado del nodo actual. Si uno de tus hijos regresa 1 invertir el estado actual entre 0 y 1, y si se regresa -1, regresar puros -1
- El grafo es valido si el dfs en todos los nodos regresa -1 ó 0



Problema #7 Nudist Beach

- Generar un grafo en el que todos los nodos que no sean fortresses pertenezcan al subset.
- Meter todos los nodos a un set, y sortearlos en base a su valor. Y elegir el nodo con el valor más bajo, y eliminarlo del subset.
- Al eliminarlo recalcular el valor de sus edges que siguen siendo parte del subset

- En cada paso se obtiene una potencial respuesta, pero el problema pide el subset que genera el mismo valor.
- Para reconstruir la respuesta, volver a hacer los mismos pasos, y detenerse cuando se encuentre en el paso que contiene la mayor.
- $O(2^E \cdot \text{Log}N)$

Problema #8 Mr. Kitayuta's Colorful Graph

- Generar el grafo, y hacer un bfs por cada query y color, en donde solo se transversa las edges del mismo color.
- $O((N+M)*Q)$

