

Programación Competitiva


Competencia #3



acm International Collegiate
Programming Contest

Problema #1: Vanya and Triangles

- Encontrar cuantos puntos son colineales, llamemoslo c .
- Respuesta es $n \text{ choose } 3 - c$.
- Para encontrar los colineales, generar la pendiente $y = mx + b$, y para cada par de segmentos meterlos a un map con un count.
- Notar que si hay x puntos con la misma pendiente, el count sera


$$x * (x-1) / 2$$

- Hacer $n \text{ choose } k$ con x y 3 y asi obtienes cuantos tienen área o para esa pendiente.

Problema #2 Password

- Encontrar el prefijo más grande que sea un prefijo y este contenido en el string.
- Usar Z-function y la respuesta es, el número más grande que esté adentro del string y también sea un sufijo,



Problema #3 Crazy Town

- Encontrar todas las pendientes que se intersecten con el segmento (a,b) .



Problema #4 Mammoth's Genome Decoding

- Contar la cantidad de A, C, G y T.
- Reemplazar los signos de interrogación para que la cantidad sea la misma.
- Si no se puede, imprimir ===



Problema #5 Han solo and lazer gun

- Generar las pendientes $y = mx + b$ y meterlas a un set.
- La respuesta es la cantidad de pendientes en el set.
- Se pueden usar vectores en vez de pendientes.



Problema #6 The Text Splitting

- Iterar sobre todos las posibles combinaciones de p con q .
- Si se puede dividirlo en p y luego en q .



Problema #7 DNA

Alignment

- Notar que el carácter que aparece más veces en el String es el que vas a poner más veces en el otro String.
- En caso de que hayan empates notar que cualquiera de los dos caracteres generan la misma cantidad de ocurrencias.
- La respuesta va a ser c^n , donde c es la cantidad de caracteres diferentes que aparecen más en el original.



Problema #8

Randomizer

- La cantidad de polígonos diferentes esta dada por la fórmula: $2^n - n \text{ choose } 2 - n \text{ choose } 1 - n \text{ choose } 0$, llamemoslo T.
- Para cada par de puntos i, j donde j es mayor que i . Si ya lo seleccionaste y no hay ningún punto seleccionado entre ellos hay $2^{(n-j+i-1)} - 1$, llamémoslo c.

- El valor esperado del área está dado por, $c / T * (\text{Area triangulo } i, j, \text{Origen})$.
- El valor esperado del área está dado por, $c / T * \gcd(\text{abs}(p_i.x - p_j.x), \text{abs}(p_i.y - p_j.y))$.
- Despejar los valores esperados usando Pick's theorem y obtener la respuesta.
- Notar que para N s muy grandes se pierde precisión y double no maneja tantos dígitos, así que solo iteramos sobre los primos 60 puntos para cada i .
- Para N mayor a 60 es $O(N*60)$, para N menor a 60 es $O(N^2)$.

