Día 1: Light cycles

Resumen del enunciado

Dados dos coches que se mueven por una cuadrícula, decir si uno de los coches visita la misma casilla que otro menos de k turnos después.

- *Autores: Iván Martín de San Lázaro y Cristian Pérez Corral*
- Primera solución con 100 puntos: Adriana Aguiló Martínez

Solución:

- En cada casilla, guardas el número de turno en la que la ha visitado el coche amarillo por última vez.

Día 1: Parejas Segedinenses

Resumen del enunciado

Dados (a,b), determina si existen enteros positivos u,v,x,y tales que:

- $a = u + v$, u y v son divisores de b.
- $b = x + y$, x e y son divisores de a.

- *Autor: Félix Moreno Peñarrubia*
- Primera solución con 100 puntos: Javier Badesa Pérez

Solución:

- Si $a = b$, tienen que ser pares los dos. Suponemos SPDG $a < b, x \leq y, u \leq v$.
- $y = a \implies b - a$ debe ser divisor de a. Esto además implica $b \leq 2a$.
- $b \leq 2a \implies v \in \left\{\frac{b}{2}, \frac{b}{3}, \frac{b}{4}\right\}$. Hay que comprobar si los valores correspondientes de u son divisores de b.

Día 1: Distancia a Múltiplo o Divisor

Resumen del enunciado

Debes determinar n haciendo estos tipos de preguntas:

1. Si m es múltiplo o divisor de n. (Coste 1)
2. La distancia mínima de m a un múltiplo o divisor de n. (Coste i^2 la i-ésima vez)

- *Autor: Félix Moreno Peñarrubia*
- Primera solución con 100 puntos: Ninguna hasta la congelación del marcador.

Solución:

- Puedes obtener un N que sabes que es múltiplo de n haciendo una pregunta de tipo 2.
- Puedes saber si un divisor d de N es divisor de n preguntando (tipo 1) por $(d+1)\frac{N}{d}$.
- Para divisores grandes, haz dos preguntas aleatorias de tipo 2 al principio, haz el MCD y será improbable que N tenga divisores grandes “extra”.

Día 1: Sueño

Resumen del enunciado

Dados $n \leq 1000$, $x \leq 10^9$, calcula para todo $1 \leq k \leq n$ el número de secuencias de n enteros en $[1, x]$ con k máximos absolutos.

- *Autor: Manuel Torres Cid*
- Primera solución con 100 puntos: Innokentiy Kaurov

Solución:

- Resolvamos el problema para potencias de 2. Consideremos el rango [0, x-1], la representación binaria de los componentes del vector y $DP[i][j]$ = Número de formas de tener i máximos absolutos procesados los j primeros bits. Observamos que para pasar de k maximos a s máximos tenemos $\binom{k}{s}$ 2^{n-k} formas de hacer más el caso degenerado de todo 0's (de k máximos a k máximos hay 2 x 2^{n-k}.
- Ya que x no es una potencia de dos, consideremos $x - 1 = 11001..._2$. Para ello retocaremos el algoritmo para cada vez que llegamos a un 0, restaremos los casos en los que se añaden valores mayores que x que no habían sido descartados por los bits anteriores.

Día 1: Sueño

Solución alternativa:

- También posible ver que para un número de máximos i, tenemos $\binom{n}{i} \sum_{j=2}^{x}(j-1)^{n-k}$ + if $(n == i)$ sumamos 1 (cuando todos son máximos absolutos pueden ser 1).
- Podemos precomputar los coeficientes binomiales pero para calcular el sumatorio tendremos que hacer uso de diferentes técnicas, ya sea usar formulas recursivas, trucos con potencias de 2,...

Día 1: Flechas

Resumen del enunciado

Dado un tablero cuadrado donde cada casilla tiene una ficha con una flecha, hay que deducir si se puede dejar el tablero vacío. Se puede sacar una ficha si, moviéndola en la dirección de su flecha, podemos deslizarla fuera del tablero sin que se choque con ninguna otra ficha.

- *Autora: Blanca Huergo Muñoz*
- Primera solución con 100 puntos: Oscar Carballo Puebla

Solución:

- Complejidad esperada: $O(n^2)$, ya que hay n^2 fichas
- El prodecidimiento voraz de quitar una ficha si es posible quitarla será óptimo. Si en algún momento nos encontramos con que no es posible quitar fichas y el tablero no está vacío, no será posible vaciarlo.
- Podemos implementar esto de forma eficiente con una BFS más $4n$ punteros que nos guarden la primera ficha por arriba y abajo de cada columna y la primera ficha por la izquierda y la derecha de cada fila, actualizando los valores a medida que sacamos las fichas.