

Problema 2
elmer - descriere soluție

*autor stud. Mircea Popoveniuc
Universitatea Babeș-Bolyai, Cluj-Napoca*

Soluție 15 puncte - $O(N \cdot M \cdot V_{MAX})$ - stud. Popoveniuc Mircea

V_{MAX} = poziția maximă unde îi este suficient vânătorului să fie poziționat pentru a putea împușca numărul maxim de rațe

Se fixează fiecare poziție de pe axa O_x între 1 și V_{MAX} pentru vânător și, pentru fiecare poziție fixată, se verifică pentru fiecare rață dacă poate fi sau nu ochită. Pentru verificare, se ia fiecare zid la rând și se verifică dacă segmentul delimitat de poziția vânătorului și rață se intersectează cu segmentul delimitat de capetele zidului. Complexitatea totală a soluției este $O(N \cdot M \cdot V_{MAX})$.

Soluție 40 puncte - $O((N \cdot M)^2)$ - stud. Budău Adrian

Pentru fiecare rață, se caută pentru zidurile din dreapta raței cea mai din stânga poziție din care rața poate fi ochită, iar pentru zidurile din stânga raței cea mai din dreapta poziție din care rața poate fi ochită. Pentru pozițiile găsite, se află câte rațe pot fi împușcate, la fel ca la soluția de 15 puncte. Se obțin $N \cdot M$ poziții "speciale" ce trebuie verificate, iar complexitatea verificării unei anumite poziții este $N \cdot M$, așadar complexitatea totală a soluției este $O((N \cdot M)^2)$.

Soluție 100 puncte - $O((N \cdot M) \cdot \log(N \cdot M))$ - stud. Popoveniuc Mircea

Pentru fiecare rață și pereche de ziduri consecutive, se află subintervalul în care rața este vizibilă. Asemănător soluției de 40 puncte, pentru aflarea subintervalului se utilizează pozițiile "speciale", dar la aflarea lor se mai acordă atenție cazurilor în care un zid mai apropiat de rață ar bloca mai tare vizibilitatea raței față de un zid mai depărtat, ținându-se o stivă pentru zidurile din stânga raței și una pentru cele din dreapta raței.

Se obțin maxim $N \cdot M$ subintervale care ar crește numărul de rațe vizibile cu 1. Se aplică un algoritm de baleiere în care capetele subintervalurilor reprezintă evenimente, capătul stâng incrementând numărul de rațe curente, iar capătul drept decrementându-l. Evenimentele trebuie sortate și apoi se iterează prin ele, obținându-se o complexitate de $O((N \cdot M) \cdot \log(N \cdot M))$.