**Анализ на задача Лабиринт**

**Първа подзадача.** Лесно може да съобразим, че участникът с най-голямо ще изведе всички останали. Нека неговата смелост е , а гъстотата на областите без хора в тях е . Тогава търсеният отговор ще е .

Сложност– **O( N × M ).**

**Втора подзадача.** След като всички участници имат еднаква смелост, можем да забравим за , че тези с по-малко се събират с тези с по-голямо, и да ги разглеждаме независимо. Тъй като търсеното **D** е малко, ще циклим по него, докато не открием някое, при което всички успяват да излезнат. За да проверим дали дадено **D** може да е отговор правим следното: обхождаме областите по края и пускаме **DFS**, който се движи по тези с гъстота до . След това проверяваме дали областта на някой от участниците е останала непосетена. Ако няма такава, сме намерили търсеното **D**.

Сложност – **O( D × N × M ).**

**Трета подзадача.** Ще приложим една стандартна техника, с която да намираме минималното **D**, без да циклим по всички възможности, а именно – двоично търсене по отговора. Очевидно, ако участниците не могат да излезнат при компромис **D**, същото ще е вярно и за **D-1**. Аналогично, ако могат да излезнат при компромис **D**, ще могат и при **D+1**. Това важи не само за тази подзадача, но и за всички останали.

Сложност – **O( log(D) × N × M ).**

**Четвърта и пета подзадача.** Тук, както в първа подзадача, участникът с най-голямо ще изведе всички останали. Ще подходим както във втора подзадача, като този път ще пускаме **DFS** от въпросния участник и ако посетим всички останали **K-1** области и една крайна, ще сме намерили отговора.

Сложност – **O( D × N × M ) / O( log(D) × N × M ).**

**Шеста и седма подзадача.** Нека участници се събират в даден момент и излизат от лабиринта заедно. Тъй като този измежду тях с най-голяма смелост не променя своето , то всички области през които са преминали, са били проходими и за него в началото. Използвайки това наблюдение, можем да направим следното, за да проверим дали дадено **D** е отговор: сортираме участниците по в низходящ ред и започваме да ги обхождаме. Ако областта на даден участник не е посетена, пускаме **DFS** от нея, който се движи по всички области достижими от текущия участник. Ако той не достигне до някоя крайна област, **D** не е достатъчно голямо. Това е така, защото нито един участник с по-голямо не е достигнал до текущия и напрактика той трябва сам (с потенциалните други участници, до които достига) да се придвижи до изходa.

Сложност – **O( D × N × M ) / O( log(D) × N × M ).**

Автор: Александър Гатев