v2d – descrierea soluţiei

Soluţia 1 – 100 puncte
Autor: Asist. univ. dr. ing. Mugurel Ionuţ Andreica
(în colaborare cu **Asist. univ. drd. ing. Eliana-Dina Tîrşa**)

Pentru fiecare zi `z` din cele `T` zile se vor calcula valorile `Pmax(z,i,j)` în felul următor. Se iniţializează `Pmax(z,i,j)` la `0`. Apoi se vor efectua câte `4` parcurgeri ale matricii, începând din fiecare din cele `4` colţuri şi vom calcula o altă matrice `cmax`. Mai exact, vom considera toate cele `4` perechi `(pi,pj)`, unde `pi=0,1` şi `pj=0,1`. Dacă `pi=0`, vom parcurge matricea în sens crescător al liniilor, iar dacă `pi=1`, o vom parcurge în sens descrescător. Vom proceda în mod similar pentru coloane.

Pentru a trata toate cele `4` cazuri în mod unitar, putem defini: `idir=2·(1-pi)-1`, `istart=(N-1)·pi` şi `ifin=(N-1)·(1-pi)+idir` (şi, în mod similar, `jdir`, `jstart` şi `jfin`). Apoi, cele `2` for-uri ale parcurgerii vor arăta în felul următor (folosind un pseudocod similar limbajului C):

```
for (i=istart; i<ifinl i += idir)
      for (j=jstart; j<jfin; j += jdir) {
            ...
      }
```

Fiecare parcurgere corespunde unui caz de „explicitare" a modulelor ce apar în formula distanţei. Astfel, în cadrul parcurgerii, vom seta:

`v = max{cmax[i-1][j], cmax[i][j-1]}`

(considerăm `cmax[0][*]=cmax[*][0]=-∞`).

Vom seta apoi `cmax[i][j] = max{v, P(z,i,j) + i·idir + j·jdir}` şi `Pmax(z,i,j) = max{Pmax(z,i,j), v - i·idir - j·jdir}`.

O dată ce sunt calculate valorile `Pmax(z,i,j)`, putem calcula imediat valorile `P(z+1,i,j)`.

Complexitatea de timp a acestei soluţii este $O(N^2 \cdot T)$. Observăm că matricile P şi Pmax corespunzătoare unor zile diferite pot fi memorate în aceeaşi zonă de memorie, astfel că soluţia foloseşte doar $O(N^2)$ memorie.

Soluţia poate fi generalizată pentru „matrici" d-dimensionale, ajungându-se la o complexitate de $O(2^d \cdot N^d \cdot T)$ (deoarece o matrice d-dimensională are 2^d „colţuri").

Soluţia 2 – 70-100 puncte
Autor: Paul-Dan Băltescu

În continuare va fi descrisă o alternativă pentru calculul matricei `Pmax` pentru fiecare din cele `T` zile. Putem presupune iniţial că un vrăjitor va ataca toate celulele din matrice, inclusiv celula corespunzătoare lui. În fiecare zi, putem sorta descrescător vrăjitorii după puterea iniţială `P` folosind countsort. Aplicăm o parcurgere în lăţime începând din celula vrăjitorului cu putere iniţială maximă şi extinzând zona afectată de el în cele 4 direcţii. Când puterea iniţială a următorilor vrăjitori este egală cu "puterea de afectare" a poziţiei curente din coadă, adăugăm în coadă şi celulele corespunzătoare acestor vrăjitori. Mai rămane de calculat `Pmax` pentru celulele vrăjitorilor care au afectat o zonă în jurul lor. Pentru o astfel de celulă, `Pmax` poate fi determinat doar de celulele din zona afectată de celula respectivă sau de celulele de start ale zonelor afectate adiacente. Verificările necesare pentru determinarea acestor valorile celulelor de start se pot face simultan cu parcurgerea în lăţime. Complexitatea acestei abordări este $O((N^2+Q)\cdot T)$, însă constanta este semnificativ mai mare, iar punctajul obţinut în urma implementării acestei idei depinde de modul de implementare.