

## Problema 1 – Alee

### Descrierea soluției

Este o problemă „clasică” a cărei rezolvare se bazează pe algoritmul lui Lee.

#### Reprezentarea informațiilor:

Vom utiliza o matrice  $A$  în care inițial vom reține valoarea  $-2$  pentru zonele libere, respectiv valoarea  $-1$  pentru zonele în care se află un copac.

Ulterior, pe măsură ce explorăm matricea în scopul determinării unei alee de lungime minimă vom reține în matrice pentru pozițiile explorate un număr natural care reprezintă distanța minimă de la poziția primei porți la poziția respectivă (exprimată în numărul de dale necesare pentru construirea aleei).

Vom parcurge matricea începând cu poziția primei porți.

Pozițiile din matrice care au fost explorate le vom reține într-o coadă, în ordinea explorării lor.

La fiecare pas vom extrage din coadă o poziție  $(x, y)$ , vom explora toți cei 4 vecini liberi neexplorați ai poziției extrase și îi vom introduce în coadă. Când explorăm un vecin, reținem în matricea  $A$  pe poziția sa distanța minimă  $(1 + A[x][y])$ .

Procedeul se repetă cât timp în coada nu este vidă și poziția în care se află cea de a doua poartă nu a fost explorată.

Rezultatul se va obține în matricea  $A$ , pe poziția celei de a doua porți.

Alocarea memoriei pentru coadă ar putea reprezenta o problemă (indicele de linie și indicele de coloană trebuie să fie memorați pe 1 byte, deoarece coada poate avea maxim  $175 \times 175$  elemente).

O soluție de a "economisi" spațiul de memorie este de a utiliza coada în mod circular (vezi soluția Pascal).

Problema poate fi abordată și recursiv, dar datorită adâncimii recursiei o astfel de abordare obține 80 de puncte.