

STIVE, COZI SI APLICATII

Ilie Dumitru

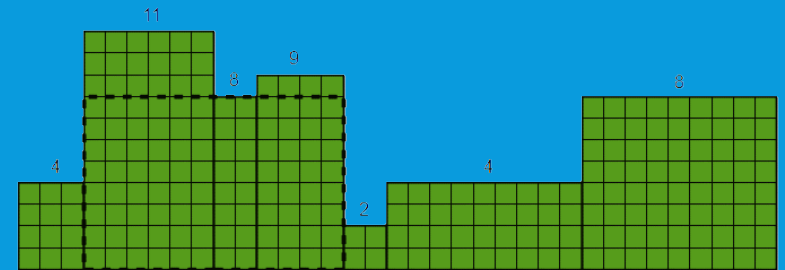
Neacsu Mihai-Costin

TEMA

- Implementati algoritmul de flood fill si rezolvati problemele:
 - [Fill](#)
 - [Lac](#)
 - [Castel](#)
 - [Labirint](#)
 - [Ferma \(extra\)](#)
- Ganditi-va la problema [Skyline](#) (nu este obligatoriu sa o rezolvati, vrem mai mult sa stiti problema pentru ca o vom rezolva data viitoare).
- Tema nu este obligatorie, dar este recomandata.

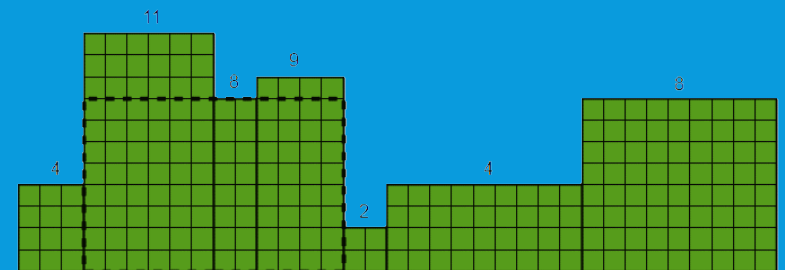
APLICATIA 2 - SKYLINE

- Problema poate fi gasita [aici](#) (exista mai multe versiuni ale problemei).
- Rezumat: Se cere aria maxima a unui dreptunghi ce poate fi suprapus cu conturul unui oras vazut de la distanta (vezi desenul).



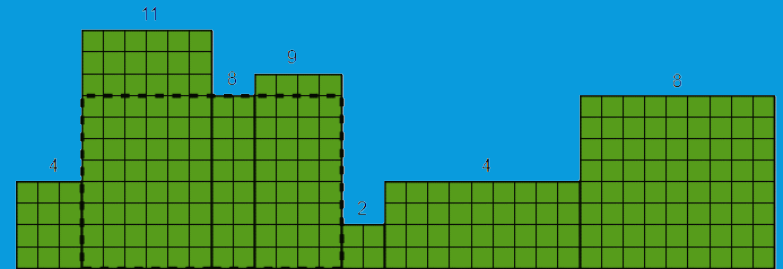
APLICATIA 2 - SKYLINE

- Problema poate fi gasita [aici](#) (exista mai multe versiuni ale problemei).
- Rezumat: Se cere aria maxima a unui dreptunghi ce poate fi suprapus cu conturul unui oras vazut de la distanta (vezi desenul).
- Pentru fiecare cladire X gasim cele mai apropiate cladiri din vecinatatea lui X (stanga si dreapta) mai scunde decat X.



APLICATIA 2 - SKYLINE

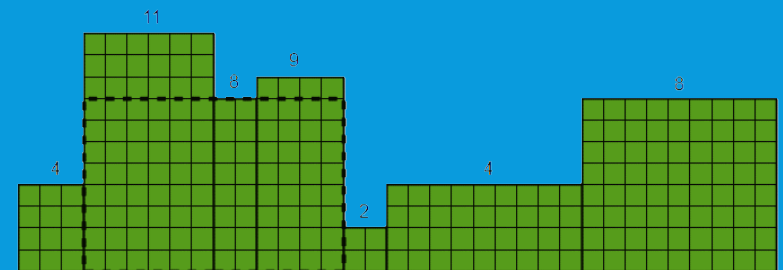
- Problema poate fi gasita [aici](#) (exista mai multe versiuni ale problemei).
- Rezumat: Se cere aria maxima a unui dreptunghi ce poate fi suprapus cu conturul unui oras vazut de la distanta (vezi desenul).
- Pentru fiecare cladire X gasim cele mai apropiate cladiri din vecinatatea lui X (stanga si dreapta) mai scunde decat X.
- Solutia se bazeaza pe urmatoarea observatie: daca cladirea i este mai joasa decat toate cladirile pana la j ($j < i$), atunci ea este mai joasa si decat cladirile mai inalte decat j.



APLICATIA 2 - SKYLINE

- Problema poate fi gasita [aici](#) (exista mai multe versiuni ale problemei).
- Rezumat: Se cere aria maxima a unui dreptunghi ce poate fi suprapus cu conturul unui oras vazut de la distanta (vezi desenul).
- Pentru fiecare cladire X gasim cele mai apropiate cladiri din vecinatatea lui X (stanga si dreapta) mai scunde decat X.
- Solutia se bazeaza pe urmatoarea observatie: daca cladirea i este mai joasa decat toate cladirile pana la j ($j < i$), atunci ea este mai joasa si decat cladirile mai inalte decat j.

Exemplu: cladirea 5 (inaltime 2) este mai joasa decat cladirea 3 (inaltime 8), deci e mai joasa si decat cladirea 2 (inaltime 11).



APLICATIA 2 - SKYLINE

Pseudocodul pentru gasirea cladirilor:

fie S o stiva

pentru fiecare cladire in ordine de la stanga la dreapta:

APLICATIA 2 - SKYLINE

Pseudocodul pentru gasirea cladirilor:

fie S o stiva

pentru fiecare cladire in ordine de la stanga la dreapta:

- cat timp S nu e goala si cladirea din varful lui S este mai inalta
scoate o cladire din S

APLICATIA 2 - SKYLINE

Pseudocodul pentru gasirea cladirilor:

fie S o stiva

pentru fiecare cladire in ordine de la stanga la dreapta:

- cat timp S nu e goala si cladirea din varful lui S este mai inalta
scoate o cladire din S

- daca S e goala atunci

 - cladirea e cea mai scunda de pana acum

APLICATIA 2 - SKYLINE

Pseudocodul pentru gasirea cladirilor:

fie S o stiva

pentru fiecare cladire in ordine de la stanga la dreapta:

- cat timp S nu e goala si cladirea din varful lui S este mai inalta
scoate o cladire din S

- daca S e goala atunci

 - cladirea e cea mai scunda de pana acum

- altfel

 - cladirea din varful lui S este cea mai apropiata cladire mai scunda

APLICATIA 2 - SKYLINE

Pseudocodul pentru gasirea cladirilor:

fie S o stiva

pentru fiecare cladire in ordine de la stanga la dreapta:

- cat timp S nu e goala si cladirea din varful lui S este mai inalta
scoate o cladire din S

- daca S e goala atunci

 - cladirea e cea mai scunda de pana acum

- altfel

 - cladirea din varful lui S este cea mai apropiata cladire mai scunda
adaug cladirea i in coada

APLICATIA 2 - SKYLINE

Pseudocodul pentru gasirea cladirilor:

fie S o stiva

pentru fiecare cladire in ordine de la stanga la dreapta:

- cat timp S nu e goala si cladirea din varful lui S este mai inalta

 - scoate o cladire din S

- daca S e goala atunci

 - cladirea e cea mai scunda de pana acum

- altfel

 - cladirea din varful lui S este cea mai apropiata cladire mai scunda

 - adaug cladirea i in coada

Pentru a calcula cladirea din dreapta se foloseste aceeasi idee dar se merge din dreapta in stanga.

Alte aplicatii ale acestui algoritm: [nrpits](#), [strabunica](#), [turnuriz](#), [leftmax](#) si multe altele

ALTE APLICATII ALE STIVELOR

Stivele pot fi folosite pentru a simula anumite proceduri. Cateva exemple includ:

- Calcularea reuniunii mai multor intervale;
- Calcularea parantezarii corecte de lungime maxima (implicit si verificarea daca un sir de paranteze este corect parantezat);
- Evaluarea expresiilor (mai ales in forma poloneza prefixata/postfixata);
- Inversarea unui sir de numere/caractere/date;
- Turnurile din Hanoi;
- Recursivitatea (orice apel de functie/procedura este tradus in cateva modificari ale unei stive).

Multe alte aplicatii mai specifice.

TEMA

- Implementati algoritmul "skyline" si rezolvati urmatoarele probleme:
 - [Skyline](#)
 - [SumMaxSecv](#)
 - [UEMM₁](#)
 - Problemele mentionate acum 2 pagini
 - [Dreptunghiuri5](#)
- Tema nu este obligatorie dar este recomandata.
- O problema extrem de grea ce se rezolva folosind algoritmul: [Skyline Photo](#) (NU este recomandata rezolvarea acestei probleme de persoanele care nu stapanesc bine conceptul)