Paul Diac, Facultatea de Informatica Iasi

**Soluție - problema 1– piscina 100 puncte**

În explicație vom considera doar soluțiile care au în comun punctul **(0, 0)**. Celelalte 3 posibilitați se pot obține simiar iar cel mai simplu în implementare este să oglindim toate punctele.De exemplu **xi** devine (**N – xi**) pentru toate punctele, similar și pentru **yi** (**M-yi**), iar la final atât pentru **xi**cât și pentru **yi**. Răspunsul va fi valoarea maximă dintre cele 4 soluții obținute prin oglindire.

O primă observație este faptul că piscina de arie maximă va avea câte un copac pe fiecare din acele două laturi care nu sunt adiacente punctului **(0, 0)**. Altfel, ar putea fi extinsă iar aria ar crește. O singură excepție de la această regulă ar fi atunci când și una din aceste laturi este comună cu una din laturile dreptunghiului.

Putem considera oricare două puncte pentru a limita dimensiunea piscineiîn partea de dreapta și sus. Dimensiunea piscinei limitate de aceste doua puncte**i** și **j** este (**max(xi, xj) \* max(yi, yj)**). O astfel de piscină este validă doar daca nu există alte puncte în acest dreptunghi.

**O(P3) – 30 puncte** se pot obține dacă iteram toate perechile **(i, j)**de puncte care pot limita piscina și apoi iar toate celelate puncte pentru a testa dacă ele sunt sau nu în interiorul piscinei.

**O(P2) – 60 puncte** se pot obține optimizând astfel:limităm dimensiunea piscinei doar pe coordonata **y**prin orice punct **(xi, yi)**. Latura maximă pe dimensiunea **x** va fi acum egală cu **x**-ul minim al oricărui punct care are valoare **y** mai mică decat **yi**. (punctele cu **y** mai mare oricum sunt în afara piscinei datorita limitei stabilite). Astfel nu trebuie iterate decât două for-uri imbricate, de unde rezultă complexitatea.

**O(PlogP) – 100 puncte** se pot obtine dacăîmbunătățim ideea cu încă un pas. Sortăm punctele dupa **y** crescător, și considerăm în această ordine că fiecare punct limitează dimensiunea **y** a piscinei. Valoarea maximă**x** pană la care putem extinde piscina este posibil doar sa scadă pe masura ce iteram punctele, atunci cand un nou punct are valoare **x** mai mica decat punctul anterior (el va avea oricum valoare **y** mai mare pentru ca punctele sunt sortate). Dacănoul punct are valoare **x** mai mare, el va fi ignorat, deoarece are valoare atat **x** cat și **y** mai mare și deci nu poate fi un punct limită optim (piscina conține coltul **(0, 0)**).

Pentru a trata simplu cazul în care piscina e limitată de dimensiunile **N**și respectiv **M** se pot adăuga punctele **(0, M)**și **(N, 0)**, apoi algorimul nu trebuie să trateze cazul diferit.

Este necesară folosirea tipului **long long** pentru a obține punctajul maxim. Folosind doar numere pe 32 biți, se pierd cel puțin cate 10, 20 și respectiv 30 puncte pentru fiecare solutie în ordine.

Solutie 2-Divide et impera

**DI** - solutiainitialaDivide&Imperafaranici o modificare

**DI\_o1** - solutia cu optimizarea o1 (un if)

**DI\_o2** - solutia cu optimizarea o2 (alt if)

**DI\_o21** - solutia cu ambeleoptimizari / if-uri:

**DI**- 40 puncte

**DI\_o1**- 50 puncte

**DI\_o2**- 60 puncte

**DI\_o21**- 70 puncte