**Laborator 2**

Se considera problema de la laboratorul 1 cu urmatoarea modificare:

Se cere un program care sa asigure urmatoarea postconditie:

**Postconditie:** Matricea initiala contine imaginea filtrata.

**Constrangere:** NU se aloca o matrice rezultat (V) temporara!

**Obiectiv**: optimizarea complexitatii-spatiu in conditiile obtinerii unei performante ridicate.

**Datele de intrare** se citesc dintr-un fisier de intrare “date.txt”.

(Fisierul trebuie creat anterior prin adaugare de numere generate aleator.)

**Implementare**

1. Java
2. C++ ( cel putin C++11 )

**Testare:** masurati timpul de executie pentru

1. N=M=10 si n=m=3; p=2;
2. N=M=1000 si n=m=5; p=1,2,4,8,16
3. N=10 M=10000 si n=m=5; p=1,2,4,8,16
4. N=10000 M=10 si n=m=5; p=1,2,4,8,16

**ObservatII (valabile pentru fiecare caz de testare 1-4):**

* Fiecare test trebuie repetat de 10 ori si evaluarea timpul de executie se considera media aritmetica a celor 10 rulari.
* Pentru fiecare varianta de testare (dintre cele 10) folositi acelasi fisier “date.txt”;
* Pentru fiecare varianta de testare (dintre cele 10) verificati corectitudinea prin comparatia rezultatului cu fisierul rezultat prin executia secventiala.

**Analiza:**

Verificati corectitudinea prin comparatie cu rezultatul de la laboratorul 1.

Comparati performanta pentru fiecare caz !

Comparati timpii obtinuti cu implementarea Java versus implementarea C++.

Evaluati complexitatea-spatiu.

**Deadline:**

* saptamana 5