



SISTEME TOLERANTE LA DEFECTE

Tema #1 High Life

Termen de predare: 05-Apr-2024 23:55

Obiective

Scopul acestei teme este de a implementa, în C, folosind biblioteca MPI, un program distribuit, scalabil cu numărul de procese, având ca scop simularea unei variante a simulatorului celular [High Life](#).

Date introductive

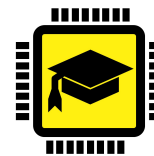
Simulatorul presupune existența unei hărți sub forma unei matrice de dimensiune $H \times W$, fiecare element având una din următoarele două stări:

- **OFF**
- **ON**

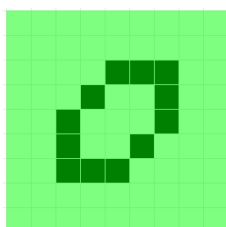
Simularea va fi executată pentru un număr dat de etape (pași). Astfel, fiecare dintre elementele hărții își schimbă starea după următoarele reguli:

- O celulă trece în starea **ON**, dacă are 3 sau 6 vecini în starea **ON**;
- O celulă își păstrează starea **ON** dacă are 2 sau 3 celule vecine în starea **ON**.

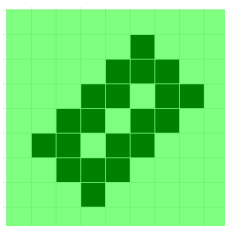
Fiind vorba de o matrice bidimensională, orice celulă are 8 vecini. O excepție apare în privința vecinilor celulelor marginale, care vor fi considerați permanent în starea **OFF**.



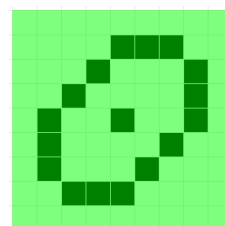
Exemplu simulare pe o hartă de dimensiune 9 x 9



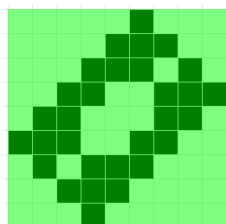
(a) Harta inițială



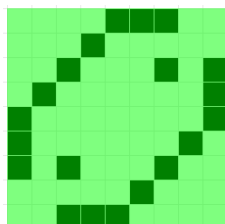
(b) După pasul 1



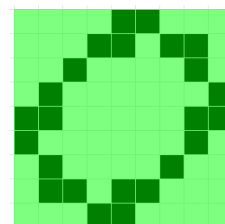
(c) După pasul 2



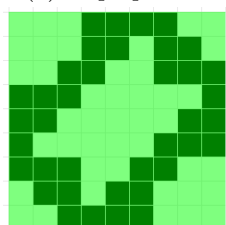
(d) După pasul 3



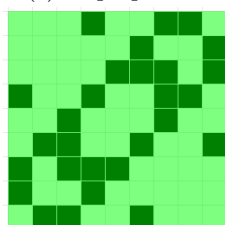
(e) După pasul 4



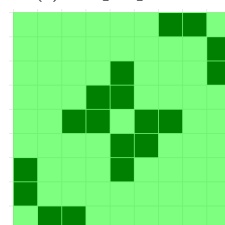
(f) După pasul 5



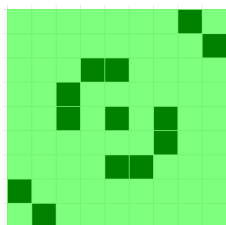
(g) După pasul 6



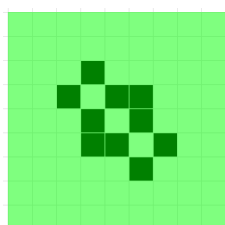
(h) După pasul 7



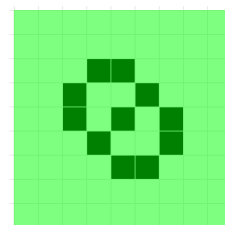
(i) După pasul 8



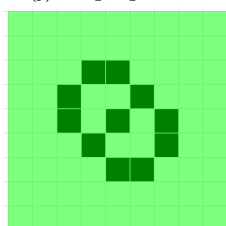
(j) După pasul 9



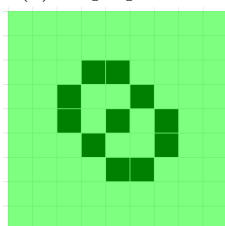
(k) După pasul 10



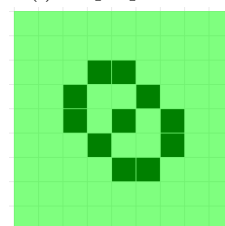
(l) După pasul 11



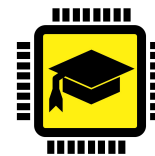
(m) După pasul 12



(n) După pasul 13



(o) După pasul 14



Procesul de simulare

Simularea va fi constituită dintr-un număr de etape, starea celulelor schimbându-se 'simultan' în funcție de celulele din etapa anterioară.

Programul va primi ca date de intrare un fișier și un număr de pași, valoare ce reprezintă numărul de etape ale simulării, iar rezultatul final va fi scris într-un fișier de ieșire.

Pe prima linie vom avea:

- Un număr întreg **H** ce indică înălțimea hărții (numărul de linii);
- Un număr întreg **W** ce indică lungimea hărții (numărul de coloane).

Pe urmă vor urma **H** linii cu **W** coloane, fiecare valoare reprezentând starea celulei la momentul inițial:

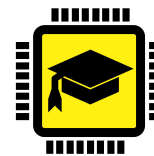
- Valoarea **0** reprezintă starea **OFF**;
- Valoarea **1** reprezintă starea **ON**.

Exemplu fișier de intrare

9	9							
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	1	1	1	0	0
0	0	0	1	0	0	1	0	0
0	0	1	0	0	0	1	0	0
0	0	1	0	0	1	0	0	0
0	0	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

Exemplu fișier de ieșire

9	9							
0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0
0	0	0	0	1	1	1	0	0
0	0	0	1	1	0	1	1	0
0	0	1	1	0	1	1	0	0
0	1	1	0	1	1	0	0	0
0	0	1	1	1	0	0	0	0
0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0



Rularea programului

Rularea programului se va realiza astfel:

```
1 mpirun --oversubscribe -np NUM_PROCS ./homework IN_FILENAME OUT_FILENAME NUM_STEPS
```

Unde:

- **NUM_PROCS** - numărul de procese;
- **IN_FILENAME** - numele (path-ul) fișierului de intrare;
- **OUT_FILENAME** - numele (path-ul) fișierului de ieșire;
- **NUM_STEPS** - numărul total de pași ce vor fi efectuați în cadrul simulării.

Exemplu:

```
1 mpirun --oversubscribe -np 2 ./homework in.txt out.txt 3
```

Trimitere și punctare

- Este necesară crearea unei arhive **.zip** care trebuie să conțină fișierul “homework.c” și uploadarea acesteia în cadrul checker-ului.
- O temă care nu compilează va primi 0 puncte. Tema va fi rulată și testată pentru corectitudine și scalabilitate pe un checker al ATM.
- Link-ul către acest checker va fi transmis ulterior.

Orice încercare de a abuza checker-ul va duce la un punctaj de 0 pe toate temele.

Distribuția punctajului este următoarea:

- 60 puncte - Output corect program distribuit și scalabil pentru hărți de dimensiuni divizibile cu 2, 4 și 8 procese.
- 40 puncte - Output corect program distribuit și scalabil pentru hărți de dimensiuni nedivizibile cu 2, 4 și 8 procese.

Arhiva .zip trebuie să conțină fișierul “homework.c”. O temă care nu compilează va primi 0 puncte. Tema va fi rulată și testată pentru corectitudine și scalabilitate pe un checker al ATM. Link-ul către acest checker va fi transmis ulterior.

Orice încercare de a abuza checker-ul va duce la un punctaj de 0 pe toate temele.