

L2 Utils

Varianta1b

1. Fiecare thread copiaza intai vecinii (frontiera) pe care trebuie sa ii foloseasca folosind vectori auxiliari.
2. Bariera de sincronizare
3. Actualizeaza elementele folosind vectorii auxiliari acolo unde este cazul.

Varianta1f

1. Fiecare thread copiaza intai vecinii (frontiere) pe care trebuie sa ii foloseasca folosind vectori auxiliari.
2. Seteaza (flag = true)
3. Verifica (vecin.flag=true pentru toti vecinii) si daca da atunci aplica 4.
4. Actualizeaza elementele folosind vectorii auxiliari acolo unde este cazul.

Varianta 2b

1. Fiecare thread calculeaza valorile noi dar pentru celulele aflate pe frontiere (vecinatate cu alte threaduri) valorile noi se salveaza in vectori separati, iar pentru celulele care nu sunt pe frontiera iar valorile noi se salveaza direct in matrice.
2. Bariera de sincronizare
3. Actualizeaza elementele de pe frontiere in matrice prin copierea vectorilor calculati la 1.

Varianta 2f

1. Fiecare thread calculeaza valorile noi pentru celulele aflate pe frontiere (vecinatate cu alte threaduri) si le salveaza in vectori separati
2. Seteaza (flag = true)–
3. Fiecare thread calculeaza valorile noi dar pentru celulele care nu sunt pe frontiere iar valorile noi se salveaza direct in matrice.
4. Verifica (vecin.flag=true, pentru toti vecinii) si daca da atunci aplica 5.
5. Actualizeaza elementele de pe frontiera prin copierea elementelor din vectorii calculati la 1.

... alte variante

Cum se compara variantele, din punct de vedere al complexitatii spatiu?

Cum se compara variantele, din punct de vedere al performantei?

Obs.

- a) flag - variabila atomica
- b) flag=false – initial
- c) pentru o granularitate mai fina a sincronizarii, in locul unui flag se pot folosi mai multe flaguri (nr_flag=nr_threaduri_vecine).

Bariera de sincronizare =primitivă de sincronizare folosită care are rolul de a se asigura că niciun thread nu poate trece mai departe de punctul în care este plasată decât atunci când toate thread-urile gestionate de barieră ajung în acel punct.

Implementare C++ (vezi directorul Utils) sau folosirea implementarii existenta in C++20.

Implementare Java (CyclicBarrier)