



## Laborator 13

[Tutorial IIInI](#)  
[MPI The complete Reference](#)

### Exerciții

**Pentru fiecare exercițiu se va scrie în fișierul REPORT.txt rezultatul rulărilor și răspunsul la întrebări.**

1. Implementați programul pentru descoperirea topologiei.
  - Algoritm pentru **TOATE** nodurile (nu există inițiator):
    - Se trimite topologia parțială tuturor vecinilor
    - Repetă
      - Se primește topologia parțială de la oricare vecin
      - Se unește topologia primită cu cea locală
      - Dacă cea locală s-a modificat se trimite tuturor vecinilor (fără cel de la care am primit)
      - Dacă au trecut mai mult de 5 secunde fără să primim o topologie, se trece mai departe.
  - **ATENȚIE:** Un nod poate comunica doar cu vecinii. MPI vă va permite să trimiteți mesaje de la orice nod la oricare altul, dar noi vrem să simulăm o rețea reală în care se poate comunica doar cu vecinii. Astfel, nu aveți voie să faceți send sau recv, decât de la un nod care este în lista de vecini.
  - La sfârșit, toate nodurile trebuie să aibă aceeași topologie la sfârșitul.
2. Având topologia **TOATE** nodurile vor construi tabele de rutare.
  - Tabela de rutare este o listă de **next hop-uri** pentru fiecare nod destinație. Next hop este primul nod din cea mai scurtă cale dintre sursă și destinație.
  - Se va folosi BFS/Dijkstra (sau alt algoritm de drum minim) pentru a determina cea mai scurtă cale de la nodul curent la toate celelalte.
  - Se va parcurge invers lista de părinți pentru a descoperi nextHop-ul.
3. După construirea tabelului de rutare, aceasta se folosește pentru a trimite un pachet de la nodul cu rank 0 la cel cu rank 10 (4 pentru versiunea mai mică a rețelei).
  - Pachetul va fi un vector de 100 caractere ce va conține pe poziția 0 nodul destinație. În restul vectorului va fi scris un “mesaj oarecare”.
  - Un mesaj cu destinație **x** va fi trimis către **nextHop[x]**.
  - Toate procesele vor printa un mesaj când primesc și trimit mai departe un pachet. Ex: “**Sunt A, am primit X de la nodul B trimit la nodul C**”
  - Se pot folosi pentru acest exercițiu funcții blocante (procesele rămân blocate în execuție).

**Exercițiile de la 1 la 3 sunt obligatorii.** Conceptele explorate sunt esențiale pentru obținerea notei **minime** de promovare.

**Vă recomandăm, pentru a crește șansele de a obține o notă cât mai mare să explorați și următoarele exerciții:**



4. Extindeți implementarea de la 3 în așa fel încât un set de 4 noduri inițiatore vor transmite fiecare 10 mesaje cu destinații alese la întâmplare.
  - Trebuie să aveți oprire curată (toate procesele să termine execuția).
  - Fiecare proces va verifica timp de 5 secunde, o dată la fiecare secundă primirea unui mesaj.
  - Nu uitați că în MPI un program nu poate ieși fără să apeleze `MPI_Finalize()`.