

**UNIVERSITATEA DE STAT DIN MOLDOVA**  
**FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ**  
**SPECIALITATEA INFORMATICA**

**LUCRARE DE LABORATOR**

**la disciplina „Programarea paralelă și distribuită”**

**Lucrările de laborator nr. 2 și 3: Transmiterea de date între procese din mediu virtual cu topologie carteziană. Divizarea și distribuirea matricelor pe grile de procese utilizând algoritmul 2D-cilic**

Verificat: Hâncu Boris, dr., conf. univ.

Efectuat: Cemîrtan Cristian, studentul grupei

I2101

## **Cuprins**

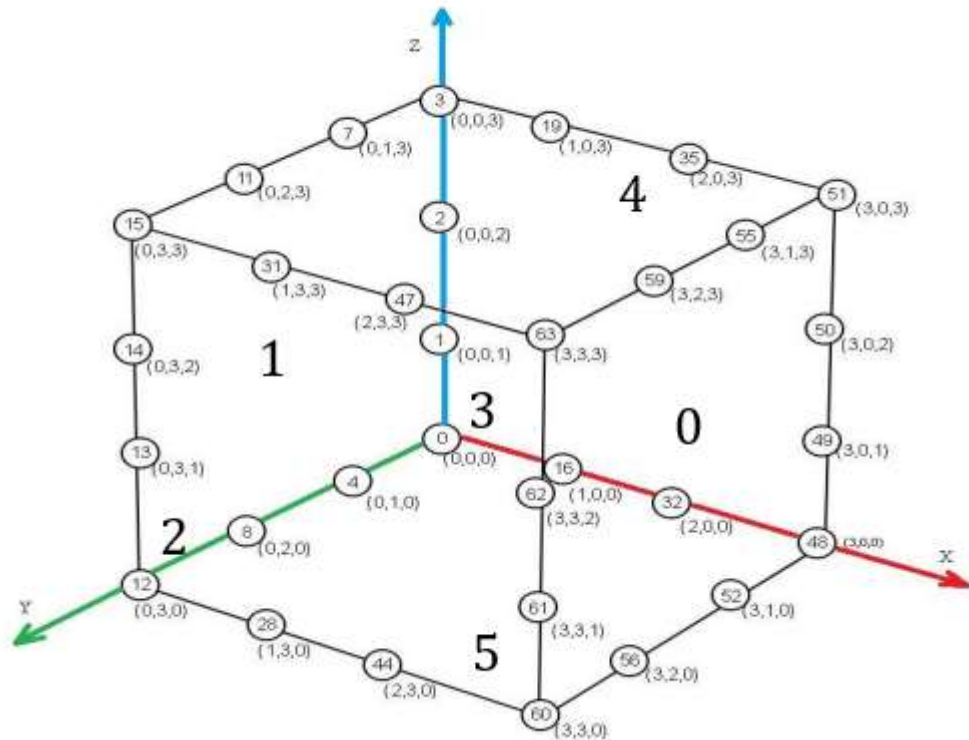
Formularea problemei I.....	3
Realizarea algoritmului I.....	4
Rezultatele r��rii programului I.....	5
Formularea problemei II.....	6
Realizarea algoritmului II.....	7
Rezultatele r��rii programului II.....	8

## **Formularea problemei I**

Să se elaboreze un program în limbajul C++ cu utilizarea funcțiilor MPI pentru gestionarea grupelor de procese și a comunicatoarelor, care ar asigura transmiterea datelor de la un proces la altul (pe cerc) în direcția acelor de ceas. Procesele aparțin muchiilor unei fațete arbitrare a cubului.

## Realizarea algoritmului I

1. Se primește de la linia de comandă, indicele fațetei a cubului;



2. Se creează un comunicator bazat pe un grup ce va fi compus din N procese, unde N este rădăcina cubică truncată a numărului de procese, la puterea 3;
3. Procesele ce nu fac parte din grupul stopează execuția programului;
4. Creăm un comunicator nou cu topologia 3D cu trei dimensiuni neperiodice cu lungimea N, pe baza comunicatorului precedent;
5. Pentru fiecare proces se află coordonatele (MPI\_Cart\_coords). Procesele stopează execuția programului dacă se află în afara fațetei specificate, sau nu au coordonatele „extreme” (se află în interiorul fațetei);
6. Fiecare proces rămas își află vecinii cu MPI\_Cart\_shift;
7. Procesele comunică între ele în direcția acelor de ceas.

## Rezultatele rulării programului I

```
/home/I01/CemirtanCristian/6$ mpiexec -host compute-1-0:64 a.out 0
Rankul 52 (3, 1, 0) expediaza rankului 56 (3, 2, 0)
Rankul 52 (3, 1, 0) a receptionat de la rankul 48 (3, 0, 0)
Rankul 48 (3, 0, 0) expediaza rankului 52 (3, 1, 0)
Rankul 48 (3, 0, 0) a receptionat de la rankul 49 (3, 0, 1)
Rankul 60 (3, 3, 0) expediaza rankului 61 (3, 3, 1)
Rankul 60 (3, 3, 0) a receptionat de la rankul 56 (3, 2, 0)
Rankul 56 (3, 2, 0) expediaza rankului 60 (3, 3, 0)
Rankul 56 (3, 2, 0) a receptionat de la rankul 52 (3, 1, 0)
Rankul 49 (3, 0, 1) expediaza rankului 48 (3, 0, 0)
Rankul 49 (3, 0, 1) a receptionat de la rankul 50 (3, 0, 2)
Rankul 50 (3, 0, 2) expediaza rankului 49 (3, 0, 1)
Rankul 50 (3, 0, 2) a receptionat de la rankul 51 (3, 0, 3)
Rankul 61 (3, 3, 1) expediaza rankului 62 (3, 3, 2)
Rankul 61 (3, 3, 1) a receptionat de la rankul 60 (3, 3, 0)
Rankul 55 (3, 1, 3) expediaza rankului 51 (3, 0, 3)
Rankul 55 (3, 1, 3) a receptionat de la rankul 59 (3, 2, 3)
Rankul 51 (3, 0, 3) expediaza rankului 50 (3, 0, 2)
Rankul 51 (3, 0, 3) a receptionat de la rankul 55 (3, 1, 3)
Rankul 62 (3, 3, 2) expediaza rankului 63 (3, 3, 3)
Rankul 62 (3, 3, 2) a receptionat de la rankul 61 (3, 3, 1)
Rankul 63 (3, 3, 3) expediaza rankului 59 (3, 2, 3)
Rankul 63 (3, 3, 3) a receptionat de la rankul 62 (3, 3, 2)
Rankul 59 (3, 2, 3) expediaza rankului 55 (3, 1, 3)
Rankul 59 (3, 2, 3) a receptionat de la rankul 63 (3, 3, 3)
/home/I01/CemirtanCristian/6$ mpiexec -host compute-1-0:8 a.out 3
Rankul 4 (1, 0, 0) expediaza rankului 5 (1, 0, 1)
Rankul 4 (1, 0, 0) a receptionat de la rankul 0 (0, 0, 0)
Rankul 0 (0, 0, 0) expediaza rankului 4 (1, 0, 0)
Rankul 0 (0, 0, 0) a receptionat de la rankul 1 (0, 0, 1)
Rankul 5 (1, 0, 1) expediaza rankului 1 (0, 0, 1)
Rankul 5 (1, 0, 1) a receptionat de la rankul 4 (1, 0, 0)
Rankul 1 (0, 0, 1) expediaza rankului 0 (0, 0, 0)
Rankul 1 (0, 0, 1) a receptionat de la rankul 5 (1, 0, 1)
```

## **Formularea problemei II**

Să se elaboreze un program în limbajul C++ cu utilizarea funcțiilor MPI pentru gestionarea grupelor de procese și a comunicatoarelor în care se realizează următoarele:

- Se realizeze un mediu de comunicare care corespunde unei grile doi dimensionale de procese.
- Procesul root inițializează cu valori aleatoare o matrice și distribuie pe procesele din grila de procese aceasta matrice în baza algoritmului 2D-ciclic.

## Realizarea algoritmului II

1. Se primește de la linia de comandă, dimensiunile pentru grila de procese, matricei și a submatricei pentru fiecare proces;
2. Creăm un comunicator bazat pe un grup nou ce va fi format din  $G_n * G_m$  de procese, unde  $G_n$  și  $G_m$  reprezintă dimensiunile grilei de intrare de procese. Procesele ce nu se încadrează în acest grup termină execuția programului;
3. Creăm un nou comunicator de tip cartezian 2D bazat pe comunicatorul precedent, unde fiecare dimensiune este periodică;
4. Procesul root inițializează cu valori aleatoare o matrice și distribuie pe procesele din grila de procese aceasta matrice în baza algoritmului 2D-ciclic;
5. Fiecare proces își afișează la ecran submatricea primită.

## Rezultatele rulării programului II

```
/home/I01/CemirtanCristian/6$ mpiexec -host compute-1-0:6 a.out 2 3 9 9 2 2
```

Matricea initiala:

```
139 149 226 237 105 202 28 243 118
131 139 25 237 167 202 157 227 127
77 48 21 107 71 76 228 212 202
199 158 97 48 169 255 110 187 59
248 186 120 96 174 95 105 40 198
8 226 163 224 94 52 243 97 195
137 165 118 67 15 66 86 199 89
32 199 140 2 126 227 212 147 208
28 220 166 88 126 13 76 246 52
```

Rankul 0:

```
139 149 28 243
131 139 157 227
248 186 105 40
8 226 243 97
28 220 76 246
```

Rankul 1:

```
226 237 118
25 237 127
120 96 198
163 224 195
166 88 52
```

Rankul 2:

```
105 202
167 202
174 95
94 52
126 13
```

Rankul 3:

```
77 48 228 212
199 158 110 187
137 165 86 199
32 199 212 147
```

Rankul 4:

```
21 107 202
97 48 59
118 67 89
140 2 208
```

Rankul 5:

```
71 76
169 255
15 66
126 227
```