

Aufgabe 07

Can Nayci, Leonhard Rattmann, Emil Sharaf

09.12.2023

1 Circle

N : Anzahl der Elemente des Arrays
 $nprocs$: Anzahl der Prozesse
 $start_p$: erstes Element für Prozess p
 end_p : letztes Element für Prozess p
 $start_0 = 1$
 $end_p = p \cdot (int) \frac{N}{nprocs} + a$
 $a = \begin{cases} \text{wenn } ((N \bmod nprocs) < p) : 0 \\ \text{sonst} : 1 \end{cases}$
 $start_p = end_{p-1}$

	Prozess	Array-Elemente
für $N = 13, nprocs = 5$:	0	0, 1, 2
	1	3, 4, 5
	2	6, 7, 8
	3	9, 10
	4	11, 12

2 Visualisierung

2.1 Richtung der Kommunikation

2.2 Communication Matrix View

2.3 Programmphasen

2.4 MPI.Init

3 Parallelisierung mit MPI

```
int n = Anzahl der Reihen;
int nprocs = Prozessanzahl;
n = (getInterlines() * 8) + 9 - 1;
int startIndex, endIndex;

nur Prozess 0:
{
    int s_startIndex, s_endIndex;
    int lpp, lpp_rest;

    lpp = (int) (n - 1) / nprocs;
    lpp_rest = (n - 1) % nprocs;

    startIndex = 1 (0er-Reihe wird ignoriert);
    endIndex = lpp + (lpp_rest < 0)? 1 : 0);
    lpp_rest--;

    s_startIndex = endIndex;
    s_endIndex = 0;
    fuer alle nprocs exkl. 0:
    {
        MPI_Ssend(*s_startIndex an Prozess i);
        s_endIndex = startIndex + lpp;
        if (lpp_rest < 0)
        {
            endIndex++;
            lpp_rest--;
        }
        MPI_Ssend(*s_endIndex an Prozess i);
        s_startIndex = s_endIndex;
        // => startIndex ist inklusive, endIndex exklusive.
    }
}

alle anderen Prozesse:
{
    MPI_Recv(*startIndex);
    MPI_Recv(*endIndex);
}
```

z.B. 64 Interlines, 4 Prozesse:

$$n = 64 * 8 + 9 - 1 = 520$$

$$lpt = \frac{519}{4} = 129$$

$$lpt_rest = 3$$

für Prozess 0:

$$startIndex = 1$$

$$endIndex = 130 + 1$$

für Prozess 3:

$$startIndex = 262$$

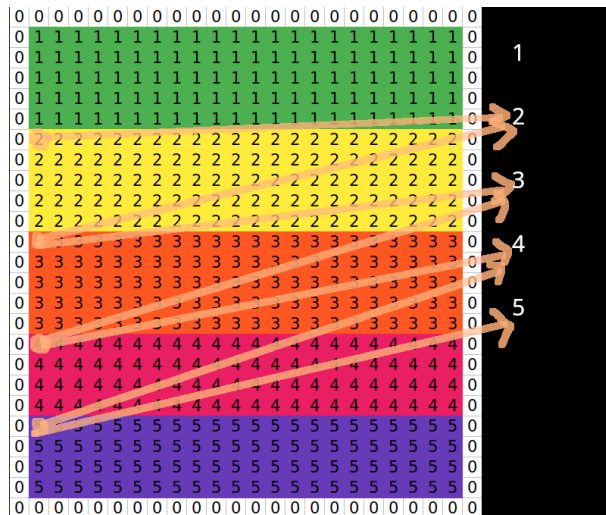
$$endIndex = 391 + 1$$

für Prozess m:

$$startIndex = 1 + m \cdot lpt + lpt_rest$$

nicht $(m - 1)$, weil Prozesse ab 0 nummeriert sind

$$endIndex = 521$$



Kommunikation

1. Alle Prozesse außer 0 (bzw. 1) brauchen ihre Start- und endIndizes. 2. Jeder Prozess braucht in jeder Iteration den oben und unten benachbarten Teil der Matrix.