

---

## Parallelisierung mit MPI (Gauß-Seidel: 500 Punkte)

Parallelisieren Sie das GS-Verfahren in dem `partdiff`-Programm gemäß dem Parallelisierungsschema.

Beachten Sie dabei folgende Anforderungen:

- Abbruch
  - Es gibt hier zwei Fälle, die auf Korrektheit der Parallelisierung zu prüfen sind:
    1. Abbruch nach fester Iterationszahl (beide Störfunktionen)
    2. Abbruch nach Genauigkeit (beide Störfunktionen) - von der Wertigkeit her entspricht dieser Aspekt **mehr als 50%** der Punkte
  - Dabei soll nach gleicher Iterationszahl das Ergebnis (Matrix und Fehlerwert) identisch bleiben.
  - Beim Abbruchkriterium „erreichte Genauigkeit“ muss die parallele Variante nicht unbedingt bei derselben Iteration wie die sequentielle abrechnen. Es kann bis zu `nprocs` Iterationen mehr gerechnet werden. Die Matrix und der Fehlerwert für `i+nprocs` Iterationen verglichen mit dem Abbruch nach `i` Iterationen muss aber gleich sein! Dabei gilt aber weiterhin, dass der Lauf deterministisch sein muss, d.h. ein wiederholter Aufruf endet immer in der gleichen Iteration.
  - Überprüfen Sie, dass die Ergebnisse mit 24 Prozessen auf zwei Knoten richtige Ergebnisse liefern.
  - Nutzen Sie keine Kenntnisse bzgl. der Initialverteilung, um Abkürzungen beim Test auf Abbruch zu nehmen.
- Code
  - Zu keinem Zeitpunkt darf ein Prozess die gesamte Matrix im Speicher halten. Die Matrix soll auf alle Prozesse gleichmäßig verteilt werden.
  - Das Programm muss weiterhin mit einem Prozess funktionieren (kontrolliert Abbrechen zählt nicht als funktionieren).
  - Das Programm muss mit beliebigen Prozesszahlen funktionieren, auch wenn es mehr Prozesse als Zeilen gibt.
  - Erstellen Sie eine eigene Funktion für die MPI-Parallelisierung des GS-Verfahrens.
  - Jacobi muss dabei weiterhin **parallel** funktionieren.
  - **Hinweis:** Sie können die in den Materialien bereitgestellte `DisplayMatrix`-Funktion als Grundlage für die parallele Ausgabe der Matrix benutzen.
- Laufzeit
  - Das Programm darf nicht langsamer als die sequentielle Variante sein.

- Kommunikation
  - Sie dürfen die Funktionen `MPI_Send` und `MPI_Isend` **nicht** verwenden. Nutzen Sie stattdessen ggf. die Funktionen `MPI_Ssend` und `MPI_Issend`.
  - Jeder nichtblockierende Kommunikationsaufruf (meist beginnend mit `MPI_I...`) muss mit einem passenden `MPI_Wait` oder einem erfolgreichen `MPI_Test` abgeschlossen werden. Anderenfalls ist die Kommunikation undefiniert bzw. falsch.

## Abgabe

Abzugeben ist ein gemäß den bekannten Richtlinien erstelltes und benanntes Archiv. Das enthaltene und gewohnt benannte Verzeichnis soll folgenden Inhalt haben:

- Alle Quellen, aus denen Ihr Programm besteht; gut dokumentiert (Kommentare bei geänderten Code-Teilen!)
  - Erwartet werden die Dateien `Makefile`, `askparams.c`, `partdiff.c` und `partdiff.h`.
- Ein `Makefile`
- **Keine** Binärdateien!

Senden Sie das Archiv an `hr-abgabe@wr.informatik.uni-hamburg.de`.