

Aufgabe 6: einfache MPI Übungsaufgaben

Dieses Übungsblatt umfasst zwei einfachere Aufgabe zur Einarbeitung in die Programmierung mit MPI und die Dartstellung eines Parallelisierungsschemas.

Sollten Probleme auftauchen schreiben Sie bitte an die Mailingliste:

`PPG-13@wr.informatik.uni-hamburg.de`

Aufgabe 6A: Iterative Berechnung von π (90 Punkte)

In der ersten Aufgabe soll die Zahl π mittels der Integration der Funktion

$$f(x) = 4/(1 + x^2)$$

berechnet werden. Dabei wird die Kurve für den Wertebereich zwischen 0 und 1 in n Teilbereiche aufgeteilt, für die jeweils die Berechnungen durchzuführen sind. Für die MPI Umsetzung sollen die Teile der Kurve auf 4 Prozesse aufgeteilt werden und die berechneten Ergebnisse danach zum Gesamtintegral zusammengefügt werden. Die Berechnungen sind in double precision durchzuführen. Dabei soll die Anzahl der Stützstellen, um das Intervall zwischen 0 und 1 zu berechnen, bei 10^9 liegen.

Zunächst ist eine Leistungsmessung für das sequentielle Programm vorzunehmen und danach für die Variante mit der MPI Implementierung. Die jeweils erzielten Näherungen für π sind auszugeben.

Aufgabe 6B: Game of Life mit einfacher MPI Kommunikation (120 Punkte)

In der zweiten Aufgabe soll wieder das Konzept von Conway's "Game of Life" in einem Programm, diesmal unter Verwendung von MPI, umgesetzt werden. Als Muster sollen stationären Figuren wie Blinker, Toad oder Beacon aus Aufgabe 2B verwendet werden.

Hierbei sollen die "Lebenszyklen" der verschiedene Muster mit 30 Iterationen durchlaufen werden. Dazu soll eine Logic Matrix erstellt werden, die als Initialisierung mit "False" belegt wird. Danach werden für die Muster der Figuren entsprechend der schwarzen Kästchen Werte in der Matrix mit dem "True" belegt. Die Berechnung soll auf einem Feld der Größe 40 X 40 erfolgen und auf 4 Prozessen durchgeführt werden. Dabei soll die gesamte Matrix auf die Prozesse aufgeteilt werden, d.h. jeder Prozess bearbeitet einen Teil der Daten.

Hinweis: Da bisher nur einfache MPI Konstrukte besprochen wurden, soll die Parallelisierung des 2D-Feldes über den Index erfolgen, der an zweiter Stelle steht. Dann liegen die später auszutauschenden Teilarrays zusammenhängend im Speicher. Die Matrix wird also nicht in gleiche Quadrate geteilt, sondern über die ganze Breite der Matrix in 4 untereinanderliegende Streifen.

Die geforderte Ausgabe ist die Darstellung der **zusammengeführten Matrix** zu jeder Iteration auf dem Bildschirm. Es findet also nur verteiltes Rechnen statt mit Ausgabe der Daten über den Masterprozess.

Es sollen die “Lebenszyklen” der verschiedene Muster mit 30 Iterationen durchlaufen und die Muster entsprechend dargestellt werden. Für die Ausgabe steht das Programm Glider Movie auf unserer Webseite zur Verfügung, dessen Ausgangsroutine Unicode-Zeichen verwendet um eine gute lesbare Darstellung zu erreichen. Ihr dürft beliebige Teile von Glider Movie wiederverwenden, die **Subroutine printTwoDLogical** müsst Ihr verwenden.

Aufgabe 6C: Konzept zur MPI Kommunikation für das “Game of Life“ mit zyklischen Randbedingungen (90 Punkte)

Damit auch dynamische Objekte mit mehreren Prozessen korrekt ablaufen, soll in dieser Aufgabe ein Parallelisierungsschema für das “Game of Life” mit zyklischen Randbedingungen in Prosaform dargestellt werden. Um das Speicherverhalten zu optimieren dürfen die einzelnen Prozesse nur die benötigte Teilmatrix im Speicher halten. Die Problematik bei der Umsetzung der zyklischen Randbedingungen liegt darin, dass die Randwerte der Matrix zum richtigen Zeitpunkt ausgetauscht werden müssen.

Abzugeben ist ein PDF-Dokument “Parallelisierungsschema.pdf” mit folgendem Inhalt:

- Prosabeschreibung der Datenaufteilung der Matrix auf die einzelnen Prozessen:
Welche Daten der Matrix werden von welchem Prozess verwaltet?
Beschreibt die Dimensionierung und Indizierung der Matrizen.
Wie werden die Daten auf die Prozesse verteilt?
- Parallelisierungsschema für die Berechnungen:
Beschreiben Sie aus der Sicht eines Prozesses, wann die Berechnung und wann die Kommunikation mit welchen Nachbarn erfolgt.
Welche Daten benötigt der Prozess von seinem Nachbarn und wann tauscht er die Daten aus?

Abgabe

Die auf dem Cluster lauffähigen FORTRAN Programme sollen als Quellcode mit der Angabe der Gruppe (Personen in der Gruppe) bis zum Dienstag den 28.5.2013 geschickt werden an:

ppg-abgabe@wr.informatik.uni-hamburg.de

Bitte dabei folgende Form wählen:

1. bitte **NUR den Quellcode, das Makefile bzw. das Konzept (Aufg. 6C) als PDF Datei** schicken,
2. bitte für **jede Aufgabe ein separates Verzeichnis anlegen** und
3. alles **als komprimiertes Archiv .tgz oder zip** schicken! D.h. es soll wirklich nur **ein einzelnes Archiv** geschickt werden!

Als Subject im Kopf der Mail bitte die Angabe: PPG-13 Blatt6 und die Liste der Familiennamen der Personen in der Übungsgruppe.