

## Aufgabe 8: einfache OpenMP Übungsaufgaben

Dieses Übungsblatt umfasst zwei Aufgaben zur Einarbeitung in die Programmierung mit OpenMP. Sollten Probleme auftauchen schreiben Sie bitte an die Mailingliste:

`PPG-14@wr.informatik.uni-hamburg.de`

### Aufgabe 8A: Iterative Berechnung von $\pi$ (120 Punkte)

In der ersten Aufgabe soll die Zahl  $\pi$  mittels der Integration der Funktion

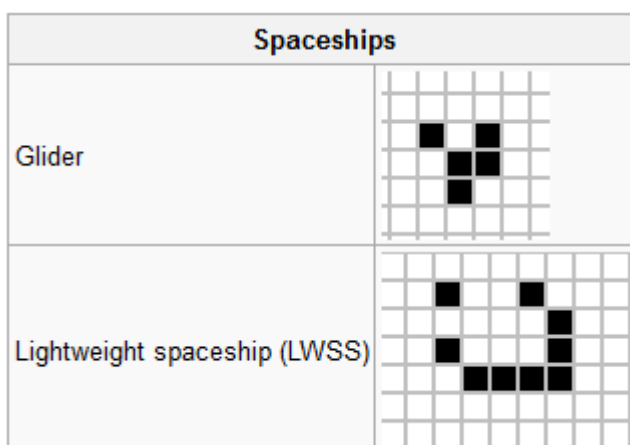
$$f(x) = 4/(1 + x^2)$$

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 OpenMP Umsetzung sollen die Teile der Kurve auf 4 Threads 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 OpenMP Implementierung für jeweils 5 Durchläufe. Die dabei erzielten Näherungen für  $\pi$  sind auszugeben.

### Aufgabe 8B: Game of Life (Bonus 120 Punkte)

In der zweiten Aufgabe soll wieder das Konzept von Conway's "Game of Life" in einem Programm, diesmal unter Verwendung von OpenMP, umgesetzt werden. Als Muster sollen der "Glider" und das Lightweight spaceship (LWSS) verwendet werden, siehe Abbildung "Spaceships".



Hierbei sollen die "Lebenszyklen" der verschiedene Muster mit 160 Iterationen durchlaufen werden. Die Berechnung soll auf einem Feld der Größe 40 X 40 erfolgen.

Da die Figuren keine stationäre Eigenschaft haben, werden zyklische Randbedingung angenommen, d.h. wenn die Figur die Matrix z.B. am rechten Rand verlassen hat soll sie am linken Rand wieder in die Matrix eingefügt werden.

Die OpenMP Implementierung soll mit 4 Threads arbeiten. Die geforderte Ausgabe ist die Darstellung der berechneten Matrix zu jeder Iteration auf dem Bildschirm, Die einzelnen Teilmatrizen sollen auf den einzelnen Threads berechnet werden und gemeinsam ausgegeben werden.

Für die Ausgabe ist wieder das Programm Glider Movie zu verwenden. Ihr dürft beliebige Teile von Glider Movie wiederverwenden, die **Subroutine printTwoDLogical** müsst Ihr verwenden.

## **Abgabe**

Die auf dem Cluster lauffähigen FORTRAN Programme sollen bis zum Dienstag den 24.6.2014 geschickt werden an:

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

Bitte dabei folgende Form wählen

1. bitte **NUR den Quellcode und das Makefile** 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-14 Blatt8 und die Liste der Familiennamen der Personen in der Übungsgruppe.