

# OpenMP

## Parallele Architekturen und Algorithmen

*Seminar SoSe 08*

Alan Blotz  
Michael Haft

# Übersicht

- Was ist OpenMP?
- Fraktale
- Demoprogramm
- OpenMP Programmierung
  - Direktiven
  - Library

# Was ist OpenMP? (1)

- *Open specification for Multi Processing*
- OpenMP Architecture Review Board (ARB)
- Soft- & Hardwarehersteller:
  - Sun, NASA, IBM, Fujitsu, Intel et al.
- 11 permanente, 5 zusätzliche Mitglieder
- 3 Vorstandsmitglieder
- 5 Mitglieder in Board Of Directors

# Was ist OpenMP? (2)

- Modell für shared memory Architekturen
- Erweitern (threadsichere) Basissprache
  - Single Program Multiple Data (SPMD)
  - Work-Sharing
  - Synchronisation
  - Gemeinsame & private Datennutzung
- => Explizite Parallelisierung

# Was ist OpenMP? (3)

- C: `#pragma omp directive clause`
  - `_OPENMP` Macro für bedingte Kompilierung
  - Direktiven sind case-sensitive
- Fortran: *sentinel directive clause*
  - Fixed Source Form:
    - `!$OMP *$OMP C$OMP; !$ *$ C$ c$ (conditional)`
  - Free Source Form:
    - `!$OMP; !$ (conditonal)`
  - Direktiven sind case-insensitive

# Was ist OpenMP? (4)

- Abhängigkeiten, Konflikte, Deadlocks, Race Conditions vom Benutzer zu überprüfen
- Parallelisierung durch fork-join Modell
- Direktive `parallel` erzeugt Threadteam
  - Statements werden parallel ausgeführt
  - Work-Sharing Konstrukt muss gleichartig von Threads abgearbeitet werden
- Synchronisation an impliziter Barrier

# Was ist OpenMP? (5)

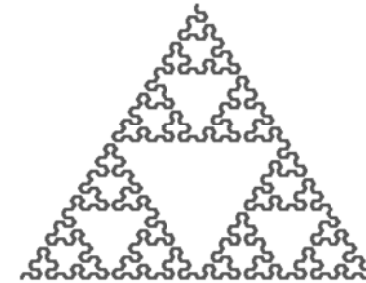
- Unsynchronisiertes I/O unbestimmt
- Direktiven in Routinen aus `parallel` Sektionen lassen einfach Parallelität auf hoher Ebene realisieren (orphaned directives)
- C: Modifikation an Shared Object vollständig:
  - Am nächsten sequence point (wenn Objekt volatile)
  - Nach `flush` Direktive bzgl. des Objekt (erst Modifikator, dann andere Threads)

# Was sind Fraktale?

- Begriff geprägt von Benoît Mandelbrot
- natürliches oder künstliches Gebilde oder geometrisches Muster
- weisen hohen Grad von Skaleninvarianz bzw. Selbstähnlichkeit auf
- weisen gebrochene Dimension auf
- Objekt das aus mehreren verkleinerten Kopien seiner selbst besteht. z.B.: Serpinski Dreieck

# Fraktale Dimension

- Motivation: Welche Dimension sollte man einer Kurve zuordnen, die eine gesamte Fläche ausfüllt?
- Es existieren mehrere Definitionen
- Beispiel: Ähnlichkeitsdimension  $D$ 
  - Fraktal aus verkleinerten Kopien seiner selbst
  - Verkleinerungsfaktor ist für alle Kopien derselbe



$$\text{Ähnlichkeitsdimension } D = \frac{\log(\text{Anzahl selbstähnlicher Teile})}{\log(\text{Verkleinerungsfaktor})}$$

# Fraktale Dimension

- Beispielrechnung für Serpinksidreieck
  - Anzahl selbstähnlicher Teile = 3
    - 3 kleinere Dreiecke verbleiben nach dem Herausnehmen des Mitteldreiecks
  - Verkleinerungsfaktor = 2
    - Länge der neuen Dreiecksseiten ist  $\frac{1}{2}$  \* Länge der Seiten des Ursprungsdreiecks

$$\text{Ähnlichkeitsdimension } D = \frac{\log(3)}{\log(2)} = 1,585$$

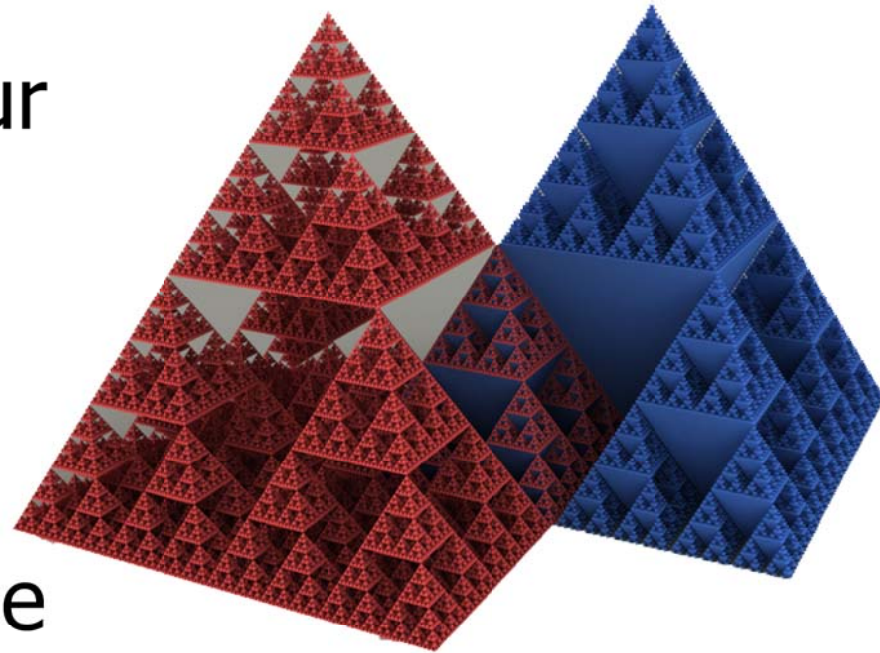
# Fraktale

- Beispiele aus der Natur

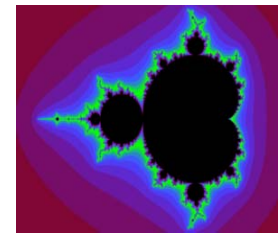
- Küstenlinien
- Schneeflocke
- Farne, Blumenkohl

- Geometrische Beispiele

- Kochkurve
- Pythagorasbaum
- Serpinksidreieck/-teppich, Menger-Schwamm
- Mandelbrotmenge



# Mandelbrotmenge



- Ist ein Fraktal, welches 1980 von Benoît Mandelbrot erstmals computergrafisch dargestellt und untersucht wurde.
- Die Mandelbrot-Menge wird als das formenreichste geometrische Gebilde bezeichnet.
- Dieser Formenreichtum zeigt sich an stark vergrößerten Ausschnitten des Randes, die überdies Beispiele für das Konzept der Selbstähnlichkeit bei Fraktalen liefern.

# Definition Mandelbrotmenge

Die Mandelbrot-Menge  $M$  ist die Menge aller komplexen Zahlen  $c$ , für welche die rekursiv definierte Folge komplexer Zahlen  $z_n$  mit dem Bildungsgesetz

$$z_{n+1} := z_n^2 + c$$

und der Anfangsbedingung

$$z_0 := 0$$

beschränkt bleibt. Die grafische Darstellung dieser Menge erfolgt in der komplexen Ebene. Die Punkte der Menge werden dabei in der Regel schwarz dargestellt.

# OpenMP / Fraktale

Demonstration des Programms

# OpenMP Direktiven (1)

- Work Sharing:
  - for [schedule, ordered, nowait, private, firstprivate, lastprivate, shared, reduction]
  - sections [nowait, private, firstprivate, lastprivate, reduction] / section
  - single [nowait, private, firstprivate, copyprivate]
  - [WORKSHARE]

# OpenMP Direktiven (2)

- Parallelisierung

- `parallel [if, num_threads, private, firstprivate, default, shared, copyin, reduction]`
- Kombinationen mit Work Sharing Direktiven möglich
- Priorität Threadanzahl:
  1. `num_threads`
  2. `omp_set_num_threads`
  3. `OMP_NUM_THREADS`
  4. Implementationsspezifisch

# OpenMP Direktiven (3)

- Synchronisation

- master

- critical [*name*]

- barrier

- atomic

- flush [*list*]

- Implizit an: barrier, parallel, critical, ordered, for, sections, single

- ordered

# OpenMP Direktiven (4)

- Datenumfeld
  - `threadprivate (list)`
    - `copyin`
  - `private (list)`
    - `firstprivate (list)`
    - `lastprivate (list)`
  - `shared (list)`
  - `default ([PRIVATE |] shared | none)`
  - `reduction (operator: list)`
  - `copyprivate (list)`

# OpenMP Library (1)

- Environment
  - `omp_(get|set)_num_threads`
  - `omp_get_max_threads`
  - `omp_get_thread_num`
  - `omp_get_num_procs`
  - `omp_in_parallel`
  - `omp_(get|set)_dynamic`
  - `omp_(get|set)_nested`

# OpenMP Library (2)

- Locks
  - Typen: `omp_[nest_]lock_t`
  - `omp_init_[nest_]lock`
  - `omp_destroy_[nest_]lock`
  - `omp_set_[nest_]lock`
  - `omp_unset_[nest_]lock`
  - `omp_test_[nest_]lock`

# OpenMP Library (3)

- Zeit:
  - `omp_get_wtime`
  - `omp_get_wtick`
- Umgebungsvariablen
  - `OMP_SCHEDULE`
  - `OMP_NUM_THREADS`
  - `OMP_DYNAMIC`
  - `OMP_NESTED`

# OpenMP / Fraktale

Danke für Ihre Aufmerksamkeit!

Haben Sie Fragen?

# Quellen

- <http://www.openmp.org/mp-documents/cspec20.pdf>
- <http://www.openmp.org/mp-documents/fspec20.pdf>
- <https://computing.llnl.gov/tutorials/openMP>
- Bausteine des Chaos – Fraktale (Springer)
- Wikipedia