

Übungen zur Vorlesung
“Objektorientiertes Programmieren im Wissenschaftlichen Rechnen”

Dr. O. Ippisch, Dr. C. Engwer

Abgabe am 01. 06. 2010 in der Vorlesung

ÜBUNG 1 STATISCHER VS. DYNAMISCHER POLYMORPHISMUS

Wir wollen Laufzeitunterschiede der Programme bei Verwendung von statischem bzw. dynamischen Polymorphismus untersuchen. Als Beispielprogramm betrachten wir wie auf dem letzten Blatt in Übung 2 die numerische Integration.

Statt der Verwendung von dynamischen Polymorphismus können wir die Quadraturregel und/oder die zu integrierende Funktion.

1. Schreiben drei zusätzliche Varianten ihres bereits entwickelten Integrationsprogrammes:
 - (a) Statische Auswahl der Quadraturregel, dynamische Auswahl der Funktion.
 - (b) Dynamische Auswahl der Quadraturregel, statische Auswahl der Funktion.
 - (c) Statische Auswahl der Quadraturregel, Statische Auswahl der Funktion.
2. Messen sie in den vier Hauptprogrammen die für die Integration benötigte Zeit. Am einfachsten geht dies, indem Sie sich den header timer.hh von der Homepage herunterladen. Die dort enthaltene Klasse `Timer` besitzt zwei Funktionen. `Timer::elapsed` gibt die benutzte CPU-Zeit in Sekunden an und `Timer::reset` setzt den Zähler auf Null.
3. Messen Sie die für die Integration von $\int_{-3}^{13} 2t^2 + 5$ mit 10002 Stützstellen benötigte Zeit für alle vier Versionen einmal ohne Optimierung (Compiler-Option “-O0”) und einmal mit Optimierung (Compiler-Option “-O3”). Messen Sie sowohl für die Simpson als auch die Trapezregel!
4. Interpretieren und begründen Sie die Laufzeitunterschiede!

20 Punkte

Wie immer gilt: Kommentieren Sie Ihr Programm. Erklären Sie was Sie tuen.