

Übungen zu Computergrundlagen WS 2010/2011

Übungsblatt 7

1. Dezember 2010

Allgemeine Hinweise

Die Lösungen solltest Du in eine Kopie der Datei `/share/Courses/CG2010/blatt7/blatt7.txt` einfügen.

Abgabetermin für die Lösungen ist

- **Montag, 6.12., 13:00** für die Übungsgruppen am Mittwoch und Donnerstag
- **Donnerstag, 9.12., 13:00** für die Übungsgruppen am Montag und Dienstag

Zur Abgabe kannst Du **entweder** den Befehl `/share/Courses/CG2010/bin/abgabe <datei>` ausführen (dabei sollte `<datei>` die Lösungsdatei bezeichnen), **oder** Du schickst die Datei per Email an den jeweiligen Tutor.

Aufgabe 7.1: Fließkommazahlen (3 Punkte)

- 7.1.1 (2 Punkte) Berechne die 32-bit Fließkommadarstellung („einfache Genauigkeit“) nach IEEE-754-Standard (http://de.wikipedia.org/wiki/IEEE_754) der folgenden Zahlen und schreibe den Lösungsweg und das Ergebnis im Hexadezimalsystem in die Lösungsdatei.
 - $\pi = 3.14159265358979323846$
 - $\frac{22}{7} = 3.1428571428571428$

Wie viele Dezimalstellen Genauigkeit hat diese Darstellung?

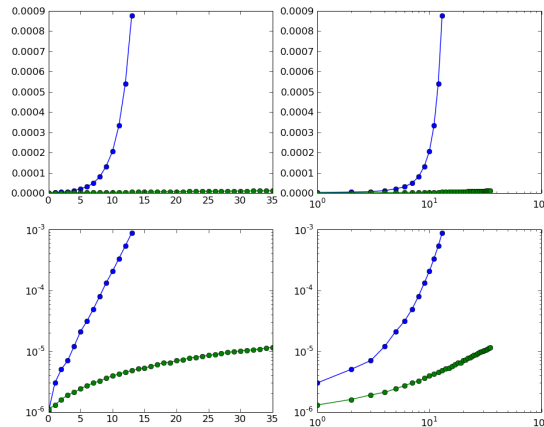
Hinweis: Du darfst gerne Python verwenden, um z.B. mit Hilfe der Funktion `hex` Dezimalzahlen in Hexadezimalzahlen umzuwandeln.

- 7.1.2 (1 Punkt) Berechne $\frac{22}{7} - \pi$ mit Hilfe der IEEE-Fließkommaarithmetik und trage das Ergebnis im Hexadezimalsystem in die Lösungsdatei ein. Wie viele Dezimalstellen Genauigkeit hat das Ergebnis?

Aufgabe 7.2: Effizienz der Berechnung von Fibonacci-Zahlen un Matplotlib (4 Punkte)

Das Skript `/share/Courses/CG2010/blatt7/timefib.py` enthält die Funktionen `fib1` und `fib2` vom letzten Übungsblatt. Erweitere das Skript wie folgt:

- Es soll die Laufzeiten $t(n)$ messen, die benötigt werden, um die Funktion `fib1(n)` für Werte von $4 \leq n \leq 17$ bzw. die Funktion `fib2(n)` für Werte von $4 \leq n \leq 40$ zu berechnen.
- Es soll mit Hilfe der `matplotlib` einen Plot erstellen, in dem die Laufzeiten $t(n)$ der verschiedenen Funktionen gegen n aufgetragen sind. Dieser soll aus vier Subplots bestehen, in denen jeweils die Plotfunktionen `plot`, `semilogx`, `semilogy` und `loglog` verwendet werden, um beide Kurven darzustellen. Der Plot sollte ungefähr so aussehen wie in umseitiger Abbildung.



Füge das Skript in die Lösungsdatei ein.

Aus den Plots kann man Aussagen über den asymptotischen Aufwand der Funktionen `fib1` und `fib2` treffen. Welchen asymptotischen Aufwand zeigen die Funktionen (exponentiell, polynomial, linear)?

Hinweis: Verwende die Funktion `time.clock()` zum Messen der Zeiten. Die Aufrufe der Funktion sind zu schnell und `time.clock()` zu ungenau, um die Zeiten einzelner Aufrufe der Funktion zu messen. Messe deswegen die Zeit, die Python benötigt, um `fib1(n)` jeweils 10000 mal und `fib2(n)` jeweils 100000 mal auszuführen.

Aufgabe 7.3: Effizienz und Konvergenz der numerischen Integration (3 Punkte)

Das Skript `/share/Courses/CG2010/blatt7/timepi.py` enthält die Funktionen `compute_pi1`, `compute_pi2` und `compute_pi3`. Diese schätzen die Kreiszahl π mit Hilfe der Algorithmen von Übungsblatt 4 ab. Erweitere das Skript wie folgt:

- Es soll die Laufzeiten t der verschiedenen Funktionen für unterschiedliche Werte von N messen. Anders als in Aufgabe 7.2 langt es dabei, die Funktion jeweils einmal aufzurufen.
- Es soll den Fehler der Abschätzung von π (also den Betrag der Differenz von π und der Abschätzung) für unterschiedliche Werte von N messen.
- N soll dabei die Werte der Zweierpotenzen von 1 bis 21 annehmen, also $N = 2^i$ für $1 \leq i \leq 21$.
- Es soll einen Plot erstellen, der aus drei Subplots besteht, in dem die folgenden Dinge gegeneinander aufgetragen werden:
 - Die Laufzeiten t der drei Funktionen über der Anzahl der Schritte N .
 - Den Fehler der drei Funktionen über der Anzahl Schritte N in einem doppeltlogarithmischen Plot (`loglog`).
 - Den Fehler der drei Funktionen über der Laufzeit t in einem doppeltlogarithmischen Plot.

Füge das Skript in die Lösungsdatei ein.

Aus den Plots kann man Aussagen über den asymptotischen Genauigkeit der Funktionen treffen. Welche asymptotische Genauigkeit zeigen die Funktionen?