

Rechnernutzung in der Physik

Institut für Experimentelle Teilchenphysik
Institut für Theoretische Teilchenphysik
Interfakultatives Institut für Anwendungen der Informatik

Dr. Th. Kuhr, Prof. Dr. M. Steinhauser, Prof. Dr. U. Husemann
Mildenberger / Hoff / Hermann / Heck
<http://comp.physik.kit.edu>

WS2012/13 – Blatt 04

Prog: Di, 13.11.2012 / Ausarb: 16.11.2012

Aufgabe 9: π – Compilieren, Linken, Makefiles Programmtestat + Ausarbeitung

Die Übersetzung eines Programmes muss nicht notwendigerweise mit einem einfachen Compileraufruf geschehen. Im Gegenteil, es ist oft erwünscht oder gar notwendig, diese Arbeit in Teilschritte zu zerlegen, z.B. wenn man mehrere Quelldateien hat oder auf bereits compilierten Code (Bibliotheken) zurückgreift.

Programmtestat:

(a) Laden Sie von der Kurswebseite die Dateien `pimain.cc`, `mathhelp.h`, `mathhelp.cc` und `Makefile.mk` und benennen Sie die Datei `Makefile.mk` in `Makefile` um. Lesen Sie bitte die Inhalte aller vier Dateien aufmerksam durch!

Starten Sie nun eine Übersetzung, indem Sie das Kommando `make` eingeben. `make` sucht voreingestellt nach den Dateien `Makefile` und `makefile` und arbeitet diese ab. Über Optionen von `make` (siehe `man make`) kann dies auch modifiziert werden.

Sie können nun beobachten, welche Kommandos der Reihe nach ausgeführt werden: Zunächst werden die einzelnen Teildateien übersetzt und dann diese Teile zusammen mit Systembibliotheks-routinen zu einem ausführbaren Programm zusammengebunden. Was passiert, wenn Sie `make` ein zweites Mal starten? Was passiert, wenn Sie nur eine Quelltextdatei ändern und dann erneut `make` aufrufen? Was bewirken `make clean` und `make chrono`?

(b) Es soll nun π schneller und genauer berechnet werden. Da die Genauigkeit der Standardzahlentypen nicht ausreichen wird, verwenden wir nun die Programmbibliothek *GNU Multiple Precision Arithmetic Library* (<http://gmplib.org>). Diese Bibliothek stellt Datentypen mit hoher Genauigkeit und zahlreiche Rechenoperationen zur Verfügung. Die Dokumentation der Bibliothek steht unter <http://gmplib.org/manual/>. Bequemerweise sind für die neuen, genauen Datentypen in der Regel die üblichen Rechenoperatoren von C++ überladen.

Der Quelltext dieser Bibliothek ist offen und einsehbar, dennoch werden in der Regel Bibliotheken in kompilierter Form verwendet und nicht immer auf den Quelltext zurückgegriffen. Konkret geschieht das hier so: Im eigenen Programm werden per `#include <gmpxx.h>` die Funktionsaufrufe und Klassenstrukturen bekannt gemacht. Der eigentliche Code der Bibliotheks-routinen wird später beim „Linken“ dazugebunden. Die benötigten Optionen beim Linken sind `-l gmpxx` `-l gmp`, das `l` steht dabei für Library. Im Poolraum ist `gmplib` installiert.

Laden Sie von der Webseite das Programmfragment `pi-gl.cc` herunter. Dieses enthält beispielhaft einige Rechenoperationen in hoher Genauigkeit. Passen Sie als ersten Schritt Ihr `Makefile` so an, dass dieses Programm übersetzt wird und ausgeführt werden kann.

Programmieren Sie nun bitte in `pi-gl.cc` den sogenannten Gauß-Legendre-Algorithmus¹ zur Berechnung von π . Dieser besteht aus folgender Iteration.

¹Quelle und Hintergrund: http://en.wikipedia.org/wiki/Gauss-Legendre_algorithm

Startwerte:

$$a_0 = 1, \quad b_0 = \frac{1}{\sqrt{2}}, \quad t_0 = \frac{1}{4}, \quad p_0 = 1.$$

Iterationsvorschrift:

$$a_{n+1} = \frac{a_n + b_n}{2}, \quad (1)$$

$$b_{n+1} = \sqrt{a_n b_n}, \quad (2)$$

$$t_{n+1} = t_n - p_n (a_n - a_{n+1})^2, \quad (3)$$

$$p_{n+1} = 2p_n. \quad (4)$$

Approximation von π :

$$\pi \approx \frac{(a_n + b_n)^2}{4t_n}.$$

Ihr Programm soll 100 Stellen von π ermitteln. Dies ist in weniger als 10 Iterationen möglich. Tipp: Der Startwerte, insbesondere $b_0 = 1/\sqrt{2}$, müssen bereits in hoher Genauigkeit berechnet werden, damit das richtige Ergebnis erzielt wird.

Ausarbeitung:

Erklären Sie bitte in eigenen Worten in jeweils etwa zwei bis fünf Sätzen die folgenden fünf Begriffe: Compiler, Objektcode, Linker, (Programm-)Bibliothek, Makefile.

Aufgabe 10: Einführung in Mathematica

freiwillig

(a) Auf der Homepage zur Vorlesung finden Sie die Datei `Intro.nb`. „nb“ steht dabei für Notebook. Kopieren Sie diese Datei in Ihr Verzeichnis und führen Sie das Kommando

```
> mathematica Intro.nb &
```

aus. Folgen Sie im Weiteren den Anweisungen des Notebooks. Machen Sie sich außerdem mit der Notebook-Oberfläche vertraut und versuchen Sie Hilfestellungen zu den verschiedenen Befehlen zu bekommen.

(b) `Mathematica` kann auch direkt aus der Shell aufgerufen werden. Nachdem Sie das Kommando `> math`

eingegeben haben, erscheint eine Befehlszeile `In[1] :=`, die Sie zur Eingabe auffordert. Probieren Sie ein paar Befehle aus der Datei `Intro.nb` aus. Mit `Quit[]` können Sie die `Mathematica`-Session beenden.
