

Vorkurs Programmierung

– Übungsblatt 1 (Tag 4) –

Fachgruppe für Computergraphik und Multimediasysteme

Andreas Kolb, Nicolas Cuntz, Felix Heide

1. Zu diesem Übungsblatt:

Ziel der ersten Programmierübung ist das Erlernen der grundlegenden Benutzung des C++-Compilers `g++`.

2. Grundlegende Informationen zum Compilieren:

Es soll im Folgenden ein erstes C++-Programm geschrieben, compiliert und schließlich ausgeführt werden.

a) Zum erstellen und editieren einer C++ Quellcode-Datei (Endung `.cpp`) kann der Editor `nedit` verwendet werden, welcher über den Befehl `nedit [Dateiname]` aufgerufen werden kann.

b) Zum Compilieren des Quellcodes wird der GNU-Compiler verwendet. Infos zur Installation unter Windows finden sich auf der Übungsseite. Zum Compilieren eines einfachen Programms starten Sie das Terminal-Programm und geben Sie den folgenden Shell-Befehl ein:

Ziel: Übersetzen von C++ Quellcode in ein ausführbares Programm Syntax:

```
> g++ cpp-Datei
```

Beschreibung: `>` ist ein Shell-Prompt, also eine Eingabezeile des Kommandozeileninterpreters. Nachfolgender Text ist ein Shell-Befehl. `cpp-Datei` ist die Datei, die den Quellcode enthält.

c) Nach erfolgreichem Compilieren befindet sich im gleichen Verzeichnis das lauffähige Programm `a.out`. Das Programm wird ausgeführt mit: `> ./a.out`

Aufgabe 1 ("Hello Triples")

Es soll ein Programm geschrieben werden, das alle durch drei teilbaren Zahlen von 1-100 in die Kommandozeile ausgibt. Nehmen Sie als Ausgangsbasis das Fakultät-Programm aus der Vorlesung. Verwenden Sie in Ihrer Lösung eine Schleife sowie grundlegende arithmetische Operatoren. Als Hilfestellung sollten Sie einen Blick auf die im Glossar angegebene Online-Dokumentation `cplusplus.com` und den folgenden Abschnitt werfen.

3. Kurze Einführung zur Benutzung von `std::cin` / `cout`:

`cout` und `cin` sind Objekte der Klasse `ostream` bzw. `istream`. Bei Programmstart werden diese Objekte automatisch für das Lesen von der Standardeingabe bzw. für das Schreiben auf die Standardausgabe angelegt. Die Standardeingabe ist standardmäßig die Tastatur, die Standardausgabe der Bildschirm. Standardein- und -ausgabe können aber auch z.B. in Dateien umgelenkt werden.

Die Shift-Operatoren `<<` und `>>`:

Die Operatoren `<<` und `>>` dienen zur formatierten Ein- und Ausgabe. Der Code:

```
int x;
cout << "Bitte eine Zahl eingeben: ";
cin >> x;
```

schiebt zunächst die Zeichenkette auf den Ausgabestrom für die Standardausgabe ins Terminal. Anschließend wird eine Zahl von der Tastatur eingelesen und in der Variable `x` abgelegt.

Manipulatoren:

Es können Manipulatoren in den Ein- bzw. Ausgabestrom eingefügt werden, die z.B. auf einfache Weise neue Formatierungen für nachfolgende Ein-/Ausgaben festlegen. Ein Beispiel ist der Manipulator `endl`, welcher für einen Zeilenwechsel sorgt. Der Code:

```
cout << "Erste Zeile" << endl << "Zweite Zeile";
```

fügt zwischen beiden Ausgaben einen Zeilenvorschub ein. Eine kurze Übersicht einiger IO-Manipulatoren für die formatierte Ein-/Ausgabe findet sich unter

<http://www.ccs.neu.edu/home/fell/COM1101/CPPManipulators.html> .

Aufgabe 2 (Quadratzahlen)

(optional)

Wenn Sie nach Bearbeitung von Aufgabe 1 noch einige Zeit haben, erweitern Sie das Programm so, dass alle Quadratzahlen von 1-1000 ausgegeben werden.

Glossar:

<http://www.ee.surrey.ac.uk/Teaching/Unix/>
Unix-Tutorial

<http://www.cplusplus.com/>
C++-Dokumentation

<http://www.ccs.neu.edu/home/fell/COM1101/CPPManipulators.html>
Einige IO-Manipulatoren