

DB II B: DBMS-Implementierung

Übung 6: Einführung in C++

Prof. Dr. Stefan Brass

Martin-Luther-Universität Halle-Wittenberg

Wintersemester 2021/22

<http://www.informatik.uni-halle.de/~brass/dbi21/>

Inhalt

- 1 Organisatorisches
- 2 Übungsblatt 4
- 3 Zertifizierung

Interesse an C++ (1)

- Möchten Sie mehr über C++ hören?
 - A. Ja, einige wenige weitere Videoaufzeichnungen wären willkommen, selbst wenn die Vorlesung dafür nicht ausfällt.
 - B. Ca. 2 weitere Vorlesungen über C++ fänge ich gut, aber das sollte während der normalen Vorlesungszeit sein.
 - C. Bitte nicht mehr C++!
 - D. Egal.

Interesse an C++ (2)

- Wie stehen Sie zu einem kleinen Projekt mit DB-Bezug als aufeinander aufbauende Hausaufgaben über ca. 4 Wochen?
 - A. Wäre eine Katastrophe für mich. Das ginge nur, wenn es alternative Hausaufgaben gäbe.
 - B. Ich fände das spannend.
 - C. Egal.
- Sollte so ein Projekt in Zweier-Teams durchgeführt werden?
 - A. Ja, das wäre eine Bereicherung.
 - B. Nach meinen Erfahrungen klappt es nicht. Am Ende macht einer die ganze Arbeit.
 - C. Egal.

C++ Compiler/Entwicklungsumgebung (1)

- Welchen C++ Compiler/Entwicklungsumgebung haben Sie benutzt?
 - A. Visual Studio Community Edition (Windows)
[\[https://www.visualstudio.com/vs/community/\]](https://www.visualstudio.com/vs/community/)
 - B. GNU g++ Compiler auf Linux
[\[https://gcc.gnu.org/\]](https://gcc.gnu.org/)
 - C. GNU Compiler auf Windows (MinGW)
[\[https://jmeubank.github.io/tdm-gcc/\]](https://jmeubank.github.io/tdm-gcc/)
 - D. Sonstiges: Clang, Eclipse, Netbeans, XCode, ...
 - E. Ich habe bisher keine Möglichkeit, C++ auszuprobieren.

Inhalt

1 Organisatorisches

2 **Übungsblatt 4**

3 Zertifizierung

Blatt 4 (1)

- Ihr Programm soll Folgendes tun:
 - Zunächst liest es eine ganze Zahl von der Standard Eingabe `cin` ein.
 - Bei einer ungültigen Eingabe, die gar keine Zahl ist (z.B. `abc`), drucken Sie „`Unguelteige Eingabe`“ und beenden den Programmablauf.
Falls ≤ 0 , entsprechend: „`Zahl nicht positiv`“.
 - Dann ist zu prüfen, ob die Zahl eine Primzahl ist.
D.h. nur durch sich selbst und 1 teilbar. Ausnahme: 1 ist keine Primzahl.
 - Falls die Zahl eine Primzahl ist, geben Sie „`Primzahl`“ aus.
 - Ist die Zahl dagegen keine Primzahl, so geben Sie aus: „`Keine Primzahl, z.B. durch  $p$  teilbar.`“
Dabei soll p der kleinste Teiler ≥ 2 sein.

Blatt 4 (2)

```
(1)  #include <iostream>
(2)  #include <cstdlib>
(3)
(4)  int main() {
(5)      std::cout << "Natuerliche Zahl: ";
(6)      int n;
(7)      std::cin >> n;
(8)      if(!std::cin) {
(9)          std::cout << "Unguelte Eingabe.\n";
(10)         std::exit(1);
(11)     }
(12)     if(n <= 0) {
(13)         std::cout << "Zahl nicht positiv.\n";
(14)         std::exit(2);
(15)     }
(16)
```

Blatt 4 (3)

```
(17) // Primzahl-Test:
(18) if(n == 1) {
(19)     std::cout << "1 ist keine Primzahl.\n";
(20)     return 0;
(21) }
(22) for(int i = 2; i < n; i++) {
(23)     if(n % i == 0) {
(24)         std::cout << "Keine Primzahl, " <<
(25)             "z.B. durch " << i <<
(26)             " teilbar.\n";
(27)         return 0;
(28)     }
(29) }
(30) std::cout << "Primzahl!\n";
(31) return 0;
(32) }
```

Hinweise zur C++Programmierung (1)

- Bitte stellen Sie sicher, dass Ihr Programm unter Linux und unter Windows kompiliert werden kann.

Dateien wie `stdafx.h`, `conio.h`, `pch.h` existieren nicht unter Linux.

- Wenn man ein Konsolenprogramm unter Windows nicht in einem Terminal-Fenster (Shell, Command Prompt) ausführt, wird es sofort nach Programmende geschlossen.
 - Man hat dann keine Gelegenheit mehr, die Ausgabe zu sehen/lesen.
 - Also muss man in diesem Fall auf eine erneute Eingabe warten, bevor man das Programm beendet.

- Schlechte Lösung: `system("pause");`

Führt Shell als eigenen Prozess aus. Dann wieder Prozess für pause. Den Befehl pause gibt es nur unter Windows. Mögliches Einfallstor für Hacker.

Hinweise zur C++Programmierung (2)

- Funktion zum Beenden des Programms mit bedingter Compilierung für ein Teilstück:

```
(6) void end(int exit_code) {  
(7)     #ifdef VER_PRESS_RETURN  
(8)         std::cout <<  
(9)             "\n(Press Return to finish)\n";  
(10)        for(char c = ' ';  
(11)            !std::cin.fail() &&  
(12)            !std::cin.eof() &&  
(13)            c!='\n'; )  
(14)            std::cin.get(c);  
(15)     #endif  
(16)        std::exit(exit_code);  
(17) }
```

Hinweise zur C++Programmierung (3)

- Bei Bedarf oben im Programm:

(1) `#define VER_PRESS_RETURN`

Auch auf Kommandozeile möglich: Option `-DVER_PRESS_RETURN` für g++. Oder:
`#if defined(_MSC_VER) || defined(_WINDOWS_) || defined(WIN32)`

- Statt `exit` und `return` muss man jetzt `end` aufrufen.
- Im Fehlerfall (Eingabe war keine Zahl) muss man
`std::cin.clear();`
aufrufen, um den Fehlerstatus aufzuheben.
- Außerdem ist nach der Zahl noch ein `'\n'` im Eingabepuffer.
Das muss man weglesen:

`std::cin.ignore(80, '\n');`

Liest 80 Zeichen — stoppt aber, nachdem `'\n'` gelesen wurde.

Hinweise zur C++Programmierung (4)

- Bitten zur Code-Formatierung:

- Nicht mehr als 80 Zeichen pro Zeile.
- Auch mit Einrechnung der Tabulatorgröße 8.

Das ist der klassische Wert unter UNIX/Linux.

- Die geschweifte Klammer zu `}` sollte auf einer eigenen Zeile stehen und von dort sollte man hochpeilen können zum `if`, `else`, `for` u.s.w., das mit dieser Klammer geschlossen wird.

D.h. die `}` sollte in der Spalte stehen, in der das `if` beginnt.

- Beispiel für Leerzeichen: `if(n <= 0) {`

Google würde nach dem `if` noch ein Leerzeichen setzen.

- Google C++ Style Guide:

[\[https://google.github.io/styleguide/cppguide.html\]](https://google.github.io/styleguide/cppguide.html)

Hinweise zur C++Programmierung (5)

- **signed** ist eine komische Typangabe. Nie gesehen.

Es gibt **signed int**, und ganz antik kann man das **int** wohl weglassen.

Aber **int** ist immer mit Vorzeichen, wenn man nicht explizit **unsigned int** schreibt. Nur bei **char** ist **signed/unsigned** implementierungsabhängig.

- Es ist korrekt, dass man nach dem Test auf negative Eingaben theoretisch mit **unsigned int** arbeiten könnte (z.B. Laufvariable in der **for**-Schleife).

Siehe Abgabe 178775.

- Sie brauchen das zusätzliche Bit nicht, was Sie dadurch gewinnen, dass Sie nur nicht-negative Zahlen haben.
- Sie machen sich nur das Leben schwer.

Zuweisungen zwischen **signed** und **unsigned** sind problematisch (Warnungen!). Wenn Sie das Bit brauchen, müssen Sie die Mühe auf sich nehmen.

Hinweise zur C++Programmierung (6)

- Aus der Übungsplattform kann man ein zip-Archiv exportieren, das jede Abgabe in einem eigenen Unterverzeichnis hat.
- Ich habe alle Abgaben mit `g++` compiliert:

```
for i in *  
do  
    echo  
    echo $i  
    echo  
    cd $i  
    g++ *.cpp  
    cd ..  
    echo  
done
```

Hinweise zur C++Programmierung (7)

- Zu besprechende Abgaben:
 - 178772: Muss es unbedingt C++ 2011 sein?
 - 178779, 178783: `sqrt` nicht deklariert
 - 178781: `exit` nicht deklariert
 - 178796, 178797: Syntaxfehler
 - 178802: `#include` fehlt, umständlich
- Mit `-Wall` zusätzlich:
 - 178775: Comparison between `signed` and `unsigned`
 - 178804: right operand of comma operator has no effect
- Mein übliche Auswahl an Warnungen:
 - 178783: ISO C++ 1998 does not support `long long`

Inhalt

- 1 Organisatorisches
- 2 Übungsblatt 4
- 3 Zertifizierung

Zertifizierung: Was ist das?

- Prüfung der (systemspezifischen) Kenntnisse durch den Datenbank-Hersteller.
- Im Erfolgsfall bekommt man
 - eine Urkunde,
 - das Recht, ein Logo auf seinen Visitenkarten und anderen Schriftstücken zu benutzen,
 - ggf. Zugriff auf eine spezielle Webseite.
- Die Zertifizierung gilt für eine bestimmte Version.

Nach einiger Zeit sinkt der Wert der Zertifizierung, oder sie kann sogar durch den Anbieter als ausgelaufen (nicht mehr gültig) erklärt werden (z.B. bei Microsoft). Man braucht dann eine Upgrade-Prüfung.

Vorteile für Sie

- Die Zertifizierung belegt Detailkenntnisse von dem jeweiligen System, die in Vorlesungen normalerweise nicht vermittelt werden.
- Bei manchen potentiellen Arbeitgebern bringt die Zertifizierung entscheidene Pluspunkte für eine Bewerbung, bei anderen eher nicht.

Falls der Arbeitgeber annimmt, dass ein Universitäts-Studium nur ganz theoretisches Wissen vermittelt, wird eine Zertifizierung einen wichtigen Unterschied machen, und eventuell etwas Berufspraxis aufwiegen, die Ihnen fehlt. Wenn der Arbeitgeber aber nur Persönlichkeit und Grundlagenwissen voraussetzt, und annimmt, dass Sie sich die Details bei ihm aneignen, bringt die Zertifizierung weniger: Größere Firmen können es sich leisten, ihre Mitarbeiter ggf. länger zu schulen.

- IT-Dienstleister und Unternehmensberatungen werben mit der Zertifizierung ihrer Mitarbeiter.
- Die Datenbank-Hersteller suggerieren, dass man die Zertifizierung nur mit viel praktischer Erfahrung am System bekommen kann, und dass dies dann auch zu sicherer und effektiverer Arbeit führt.

6-21 / 27

Vorteile für DBMS-Anbieter

- Schulung und Zertifizierung ist ein Geschäft.

DBMS-Anbieter verdienen keineswegs nur mit dem Verkauf ihrer Software. Die Zertifizierungsprüfungen kosten Geld, und häufig brauchen Kandidaten vorher Schulungen, Bücher, Beispielfragen, etc.

- Wer eine Zertifizierungsprüfung geschafft hat, wird seinem Arbeitgeber auch eher das entsprechende DBMS empfehlen.

Aufgrund des Aufwands sind Doppel-Zertifizierungen selten.

- Eine Technologie kann sich nur durchsetzen, wenn es ausreichend viele geschulte Experten gibt.

Anerkennung durch Zertifizierung eventuell Anreiz.

Nachteile für Sie

- Kostet viel Zeit (und eventuell auch Geld).
- Das erworbene Wissen ist einem wesentlichen Teil sehr systemspezifisch.

Natürlich sind viele Konzepte z.B. in DB2 und in Oracle recht ähnlich. Genaue Kenntnis eines Systems könnte Ihnen dann helfen, sich leichter in das andere System einzuarbeiten (wenn die Unterschiede in den Details Sie nicht durcheinander bringen). Für die Prüfung werden Sie aber auch Dinge lernen müssen, die nur für das eine System gelten.

- Sie werden auch Dinge auswendig lernen müssen, die im realen Leben auch ein erfahrener DBA eher nachschlagen würde, weil er sie so selten braucht.

Zu einem Teil kann man aber raten. Erfahrung verbessert Intuition.

Zertifizierung: Ablauf (1)

- Ankreuzaufgaben am Rechner

- Z.B. 60 Fragen in 90 Minuten

- Typischerweise 4–5 mögliche Antworten

Meist ist genau eine richtig. Es gibt aber auch Aufgaben, bei denen mehrere Antworten richtig sind, dann wird explizit gesagt, wie viele man ankreuzen soll. Die beiden Arten von Aufgaben unterscheiden sich auch durch Radiobuttons vs. Checkboxes.

- Typischerweise muss man 60–65% der Fragen richtig beantwortet haben, um zu bestehen.

Der genaue Prozentsatz, den man erreicht hat, steht (soweit mir bekannt) nicht auf dem Zeugnis.

Zertifizierung: Ablauf (2)

- Es sind keinerlei Unterlagen erlaubt (man muss alles auswendig wissen).

Normalerweise bekommt man ein leeres Blatt Papier und einen Stift für Notizen während der Prüfung. Wenn die Aufsichtsperson streng ist, darf man das Papier hinterher nicht mitnehmen.

- Die Zeit ist knapp, aber nicht sehr knapp.

Da keine Unterlagen erlaubt sind, kann man nicht mehr viel machen, wenn man die Antwort nicht weiß. Es lohnt sich dann nicht, zu lange hin und her zu überlegen. Bei meiner Oracle8-Zertifizierung bin ich nie in zeitliche Bedrängnis gekommen. Bei meiner DB2-Zertifizierung ein bißchen. Bei einem Microsoft-Probetest, den ich mal probiert habe, war es hoffnungslos. Das korrespondiert allerdings mit meinem Wissen der verschiedenen Systeme.

Oracle-Zertifizierung

- Oracle Certified Foundations Associate, Database

[https://education.oracle.com/products/trackp_371]

Eine Prüfung: 1Z0-006: Oracle Database Foundations: 75 Fragen in 150 min, mindestens 60% richtig, 81 €, enthält auch ER-Model und logischen Entwurf, Normalformen, SQL, eher keine Administration.

- Oracle 12c Administrator Certified Associate

[<https://education.oracle.com/>

[oracle-database-12c-administrator-certified-associate/trackp_248](https://education.oracle.com/oracle-database-12c-administrator-certified-associate/trackp_248)]

- 1Z0-071 Oracle Database 12c: SQL

- 1Z0-062 Oracle Database 12c: Administration

Die Prüfungen kosten je 220 €. 78/67 Fragen in 120 min. 63/64% richtig.

- Oracle 11g DBA Certified Professional

Weitere Prüfung (220 €) und Nachweis von Schulung (ab ca. 1400 €).

[https://education.oracle.com/oracle-certification-path/pFamily_32]

IBM DB Zertifizierungen

- IBM Certified Database Associate
DB2 11.1 Fundamentals for LUW (Linux, UNIX, Windows)
[\[https://www.ibm.com/certify/cert?id=08003506\]](https://www.ibm.com/certify/cert?id=08003506)
Enthält eine Prüfung: DB2 11.1 Fundamentals for LUW (C2090-616)
63 Fragen in 90 min, mindestens 43 müssen richtig beantwortet sein (68%).
Test Center in Halle: Comcave College GmbH (Große Ulrichstraße 60 D) und
future Training & Consulting GmbH (Fiete-Schulze Strasse 13). Ca. 200 €.
Demo Test: [\[http://www.pearsonvue.com/ibm/faq_itemtypedemo.asp\]](http://www.pearsonvue.com/ibm/faq_itemtypedemo.asp)
- IBM Certified Database Administrator
DB2 11.1 for Linux, UNIX, and Windows
[\[http://www-03.ibm.com/certify/certs/08002109.shtml\]](http://www-03.ibm.com/certify/certs/08002109.shtml)
- “Everything you need to know”
[\[https://www.ibm.com/developerworks/data/library/techarticle/dm-1305db210certoverview/\]](https://www.ibm.com/developerworks/data/library/techarticle/dm-1305db210certoverview/)