Zu Übung02, Besprechung: 21.05.2015 14^{00} - Neue Chemie

Aufgabe 2.01: Verständnisfragen

a) Die Quelldatei wird zur Übersetzung an den _____ übergeben.

b) Der _____ bindet eine Objektdatei mit anderen Modulen zu einer ausführbaren Datei.

c) Beginnt die Zeile mit einem Doppelkreuz # am Anfang, so ist die Zeile für den _____ bestimmt.

d) Ein Datentyp bestimmt
   i. die Art der Darstellung der Daten auf dem Bildschirm
   ii. die Art der internen Darstellung der Daten
   iii. die Anzahl der benötigten Speicherplätze in Bytes

e) Wird eine interne Variable ohne Initialisierung definiert, so wird
   i. der Typ und Name der Variablen festgelegt
   ii. der Variablen automatisch ein Anfangswert zugewiesen
   iii. der entsprechende Speicherplatz für die Variable reserviert

f) Zur Definition einer Variablen, die einmal initialisiert wird und später nicht mehr verändert werden soll, wird das Schlüsselwort _____ verwendet.

g) In C++ ist der Index des **ersten** Arrayelements _____.

h) In C++ ist der Index des **letzten** Arrayelements _____.

i) Benennen Sie alle Fehler im folgenden Codeausschnitt:

_____

```
float index = 0;
float[5] myArray = {0, 1, 2, 3, 4, 5};
myArray[index] = 10;
```

j) Nach Ausführung der folgenden Anweisungen speichert die Variable **x** den Wert:

_____

```
int x = 1, y = 10;
if( y > 0 ) {
  if( y < 5 ) {
    x = 5;
  } else {
    x = 10;
  }
}
```

k) Nach Ausführung der folgenden Schleife speichert die Variable **x** den folgenden Wert:

i. 1 ii. 2 iii. 3

```
int x = 0, y = 1;
while( ++y < 3 ) {
  x += y;
}
```

Aufgabe 2.02: Definition und Initialisierung von Variablen

Welche der folgenden Aussagen ist zulässig; das heißt sie wird vom C++ Compiler ohne Fehler übersetzt? Welche der Aussagen ist sinnvoll, also das Ergebnis erscheint so, wie man es vom Programmaufruf her erwartet.

Aussage	Zulässig	Sinnvoll	Bemerkung
`int a = 23;`			
`int b = 5.7;`			
`char c = 300;`			
`double f = 1.2E4;`			
`float d = 9 / 4;`			
`char g = '\'';`			
`double äquatum = 4;`			
`const float epi;`			
`float e = 9.0 / 4;`			
`short h = 32769;`			

Aufgabe 2.03: Gültigkeitsbereich

Gegeben sei der folgende Programmausschnitt:

```
double fq = 0.6180339887;   //Fibonacci-Quotient
int b = 10000;

int hash( unsinged int x ) {
  double temp = x * fq - (int)( x * fq );
  return b * temp;
}
```

a) In welchem Bereich des Programms kann auf die Variablen `fq`, `b`, `x` und `temp` zugegriffen werden?

b) Welchen Wertebereich haben die Return-Werte der Funktion `hash()`?

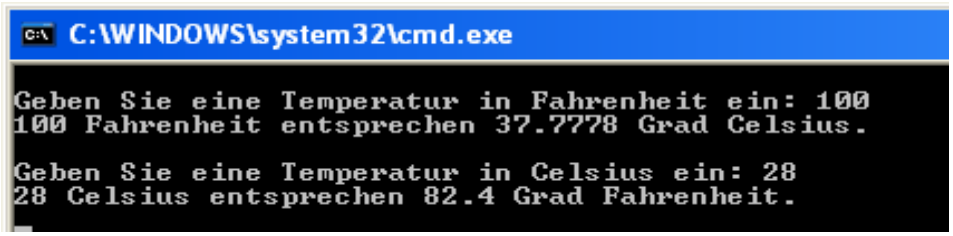
Aufgabe 2.04: Ein- und Ausgabe

Schreiben Sie ein C++-Programm, das einen Temperaturwert in der Einheit Fahrenheit im Dialog einliest und in die Einheit Celsius umrechnet und ausgibt.

Hinweis: Verwenden Sie zum Umrechnen die folgende Formel:

$$5 \cdot (Fahrenheit - 32) = 9 \cdot Celsius$$

Beispielausgabe:



Aufgabe 2.05: Mehrfachentscheidungen

Transformieren Sie den folgenden Codeausschnitt unter Anwendung einer **`switch`**-Anweisung und vermeiden Sie dabei den redundanten Code zur Ausgabe:

```
unsigned int number;
cout << "Geben Sie eine Zahl zwischen 1 und 5 ein: ";
cin >> number;

if( number == 0 ) {
  cout << "Eingabe zu klein!\n";
} else if( number == 1 ) {
  cout << "Eingabe ok und ungerade\n";
} else if( number == 2 ) {
  cout << "Eingabe ok und gerade\n";
} else if( number == 3 ) {
  cout << "Eingabe ok und ungerade\n";
} else if( number == 4 ) {
  cout << "Eingabe ok und gerade\n";
} else if( number == 5 ) {
  cout << "Eingabe ok und ungerade\n";
} else {
  cout << "Eingabe zu gross!\n";
}
```

Aufgabe 2.06: Anwendung von Kontrollstrukturen

Schreiben Sie die erforderlichen Anweisungen, um

a) zwei Gleitpunktzahlen im Dialog einzulesen und die größere Zahl von der Kleineren zu subtrahieren und das Ergebnis auszugeben.

b) Gleitpunktzahlen einzulesen und aufzusummieren, bis ihre Summe 100 überschreitet.

c) positive Ganzahlen solange einzulesen, bis eine negative Zahl eingegeben wird.

d) den Anwender aufzufordern, drei verschiedene Ganzzahlen einzugeben. Die Aufforderung wird wiederholt, solange zwei von drei Zahlen übereinstimmen.

Hinweis: Bitte beachten Sie, dass Sie auch Schleifen nur mit Bedingungen und ohne Anweisungen schreiben können.

Aufgabe 2.07: Mehrdimensionale Arrays und Schleifen

Erstellen Sie ein C++ Programm, das im Dialog eine 3x3 Matrix einliest, ihre Determinante berechnet und das Ergebnis auf dem Bildschirm ausgibt. Verwenden Sie zur internen Speicherung der Matrix ein mehrdimensionales Array.

Hinweis:

$$\det A = \det \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$
$$= a_{11}a_{22}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} - a_{13}a_{22}a_{31} - a_{12}a_{21}a_{33} - a_{11}a_{23}a_{32}.$$

Beispielausgabe: