Prof. Dr. Carsten Meyer
Fachbereich Informatik & Elektrotechnik
Fachhochschule Kiel

Programmieren in C – Wintersemester 2019 / 20

Aufgabenblatt 4

Abgabe: Montag, 09.12.2019, 08:30 Uhr (schriftlich und elektronisch)

Aufgabe 1 (Ausdrücke und Operatoren):

a) Ein C-Programm enthält folgende Deklarationen:

```
int i = 8, j = 5;
float x = 0.005, y = -0.01;
char c = 'c', f = 'f';
```

Bestimmen Sie die Werte folgender Ausdrücke, wenn sie hintereinander ausgeführt werden (d.h. eine Zeile nach der anderen; Änderungen in einer vorherigen Zeile haben Auswirkungen auf die nachfolgenden Zeilen):

```
(1) ( 3*i - 2*j ) % ( f - c )
(2) 2 * ( (i/5) + ( 4 * (j-3) ) % (i+j-2) )
(3) ++i
(4) i++
(5) --j
(6) ++x
(7) y--
(8) i <= j
(9) c > f
(10) x >= 0
(11) x < y
(12) j != 5
(13) c == 99
(14) 5 * (i+j) > 'c'
(15) ( 2*x + y ) == 1
```

(Quelle: Schwenk, Uni Bochum)

b) Bestimmen Sie die Werte folgender Ausdrücke (als Ganzzahl):

```
(1)     2 & 3
(2)     2 & 4
(3)     2 && 4
(4)     2 | 4
(5)     2 ^ 3
(6)     3 << 1
(7)     3 << 2
(8)     9 >> 2
```

(Quelle: Wulff, FH Münster)

c) Beschreiben Sie die Wirkung folgender Ausdrücke (mit ganzzahligem **x** und **y**):

```
x = x & 0177;
y = y & ~077;
```

Hinweis: 0177 und 077 sind Oktalzahlen!

(Quelle: Wulff, FH Münster)

Aufgabe 2 (Bitoperationen, Kontrollstrukturen):

Schreiben Sie ein Programm, das eine 32-Bit Integerzahl einliest und bitweise ausgibt. In der Ausgabe soll nach jeweils 4 Bits ein zusätzliches Leerzeichen ausgegeben werden (so daß die Ausgabe-Bitfolge in jeweils 4 Bits gruppiert und damit leichter lesbar ist).

Testen Sie Ihr Programm mit den Zahlen –1 und 23 sowie 123456789.

Die Lösung soll das Programm und die Programmausgabe für die genannten Zahlen umfassen.

Lösungsvorschlag:

Um zu ermitteln, ob in der Zahl das Bit an Position ′n′ gesetzt ist (Wert 1) oder nicht (Wert 0), können Sie den bitweisen UND-Operator auf die Zahl und eine Maske anwenden, die so definiert ist, daß sie genau an Position ′n′ ein gesetztes Bit hat (Wert 1) und die Bits an allen anderen Stellen ungesetzt sind. Diese Maske – die vom Typ **`unsigned int`** sein sollte – können Sie erzeugen, indem Sie den Shift-Operator ″**<<**″ (n–1) Mal auf die Zahl 1 (Initialisierung) anwenden (Beispiel: Die Anweisung

```
maske = 1 << 2;
```

erzeugt eine Maske, die nur ein gesetztes Bit hat, und zwar an der drittkleinsten Stelle, entsprechend dem Koeffizienten von 2^2).

Erzeugen Sie sich eine solche Maske, die zunächst dem größten Bit entspricht (entsprechend dem Koeffizienten von 2^{31}; da diese Zahl mit **`int`** nicht darstellbar ist, benötigen Sie für die Maske den Datentyp **`unsigned int`**). Prüfen Sie – wie oben angegeben -, ob dieses höchste Bit für die Zahl gesetzt ist; wenn ja, geben Sie eine „1“ aus, wenn nein, geben Sie eine „0“ aus. Schreiben Sie anschließend eine Schleife, in der das „Prüfbit“ in der Maske durch Anwendung des Shift-Operators **`>>`** sukzessive verschoben wird, bis das kleinste Bit in der Zahl geprüft wird; nach jeder Verschiebung des „Prüfbits“ in der Maske wird getestet, ob das entsprechende Bit in der Zahl gesetzt ist, und das Ergebnis auf dem Bildschirm ausgegeben.

Aufgabe 3 (Verzweigungen, Operatoren):

a) Analysieren Sie das folgende Programm und sagen Sie die Programmausgabe aus Ihren Überlegungen heraus vorher. Überprüfen Sie Ihre Vorhersage, indem Sie das Programm kompilieren und ausführen.

```
#include <stdio.h>

int main(void)
{
    int x, y, z;
    y = 1;
    if ( y != 0 ) x = 5;
    if ( y == 0 ) x = 3;
    else z = 7;
    x = 1;
    if ( y > 0 ) if ( x < 0) z = 3;
    else y = 4;
    if ( z = y > 0 ) x = 3;
    else if ( y == 0 ) x = 5;
    else x = 7;
    if (  z = ( y == 4 ) ) x = 2;
    printf("x = %d, y = %d, z = %d\n", x, y, z);
    if ( x = z = y ); x = 3;
    printf("x = %d, y = %d, z = %d\n", x, y, z);

    return 0;
}
```

(nach RRZN)

b) In unserem Kalender sind zum Ausgleich der astronomischen und kalendarischen Jahreslänge in regelmäßigen Abständen Schaltjahre eingebaut. Zur exakten Festlegung der Schaltjahre dienen die folgenden Regeln:
   - Wenn die Jahreszahl durch 4 teilbar ist, ist das Jahr ein Schaltjahr.
     Diese Regel hat allerdings eine Ausnahme:
   - Wenn die Jahreszahl durch 100 teilbar ist, ist das Jahr *kein* Schaltjahr.
     Diese Ausnahme hat wiederum eine Ausnahme:
   - Wenn die Jahreszahl durch 400 teilbar ist, ist das Jahr doch ein Schaltjahr.

Erstellen Sie ein Programm, das berechnet, ob eine vom Benutzer eingegebene Jahreszahl ein Schaltjahr bezeichnet oder nicht.

Testen Sie Ihr Programm mit den Jahreszahlen 2011, 2008, 2000, 1900. Die Lösung soll das Programm und die Programmausgabe für die 4 genannten Jahreszahlen umfassen.

(nach Univ. Regensburg)

Aufgabe 4 (Übungsprogramm zu Bit-/Shiftoperatoren, Verzweigungen):

Schreiben Sie ein Programm zum Üben der Bit- und Shiftoperatoren. Dabei soll der Benutzer zwei Zahlen eingeben, die vom Programm mit den Bit- und Shiftoperatoren verknüpft werden. Der Benutzer wird nach dem Ergebnis gefragt und seine / ihre Vorhersage mit dem tatsächlichen Ergebnis verglichen; am Ende des Programms wird die Anzahl korrekter Benutzereingaben ermittelt. Ein Programmlauf könnte wie folgt aussehen:

```
Uebungen zu den Bit-Operatoren
==============================

Geben Sie zwei Ganzzahlen >= 0 (durch Leerzeichen getrennt) ein: 2 3

Wieviel ist 2 & 3 ? 2
            richtig !

Wieviel ist 2 | 3 ? 4
            falsch ! (richtig waere: 2 | 3 = 3)

Wieviel ist 2 ^ 3 ? 1
            richtig !

Wieviel ist ~2 ? -3
            richtig !

Wieviel ist 2 >> 3 ? 1
            falsch ! (richtig waere: 2 >> 3 = 0)

Wieviel ist 2 << 3 ? 16
            richtig !

Ergebnis der Bit-Operator-Uebungen:

Sie haben 4 Aufgaben richtig geloest!
```

Arbeiten Sie mit dem Datentyp **`unsigned int`** (Formatierung: **`%u`**), außer bei dem Ergebnis der Operation "**`~`**" (bitweises Komplement), da diese Operation, auf eine positive Ganzzahl angewendet, auch ein negatives Ergebnis erzeugen kann.

Die Lösung der Aufgabe soll das Programm und die Programmausgabe umfassen.

Tip: Sie können das Programm auch nutzen, um die Wirkungsweise der Operatoren zu üben.