



Nagl, Becker, Ranger, Wörzberger

## Übungen zur Vorlesung „Grundgebiete der Informatik 2: Algorithmen und Programmieretechniken“

— Lösung zu Blatt 2 —

### 4. Aufgabe

(a) Es werden folgende Werte angenommen:

a	b	c
-3	4	0
-2	2	-10
-8	11	10
0	4	-128
42	42	42

Zu beachten war, daß  $-8/12=0$  (bei  $a=a/12$ ), da bei ganzzahliger Division abgerundet wird und, daß hier  $10+118=-128$  (bei  $c+=118$ ), da  $c$  vom Typ `signed char` ist und deswegen der Wertebereich den Wert  $+128$  nicht enthält.

(b) Es gibt z.B. folgende Möglichkeiten (hier jeweils die Kurzschreibweisen):

```
int a = 28;

a = 12;
a -= 16;
a += -16;
a %= 16;
a ^= 16;
a &= 12;

-----a;
a = (int) (a / 2.3 );
a = (int) (a * 0.43);
```

### 5. Aufgabe

(a) Das Programm initialisiert zunächst jeden Eintrag des Felds `gezaehlteZeichen` mit 0 und liest dann Zeichen von der Standard-Eingabe ein, bis das Ende der Eingabe erreicht wird. Dabei wird im Feld `gezaehlteZeichen` für jedes mögliche Zeichen mitgezählt, wie oft es gelesen wurde. Mit Hilfe der berechneten Häufigkeiten wird danach ein häufigstes Zeichen gesucht und ausgegeben.

Im Detail finden sich noch einige Fragen, deren Beantwortung hier nicht verlangt war. Dazu gehören die Abschlußbedingungen der Schleifen, die genaue Bedeutung der Feldzugriffe und das Wirken der Typumwandlung (*type casts*, beispielsweise `(char) zeichenIndex`). So wird die Leseschleife verlassen, sobald die Standardeingabe leer ist, was `cin.get` dadurch anzeigt, daß es `NULL` anstelle von `cin` zurückliefert.

(b) Die Beantwortung dieser Frage ist natürlich subjektiv, deswegen hier nur einige Stichpunkte

- + gute Formatierung des Quelltextes erleichtert das Lesen des Programms
- + aussagekräftige Bezeichner wurden verwendet
- + `ANZAHL_ZEICHEN` ist als Konstante deklariert, deswegen könnte das Programm relativ leicht angepaßt werden, wenn die Größe von `gezaehlteZeichen` geändert werden muß
- keine Kommentare
- Verwendung des `char gelesenesZeichen` als Index zum Feldzugriff ist problematisch, da der Wertebereich von `char` plattformabhängig ist
- bei ganz leerer Eingabe (kein Zeichen) wird `'0'` als häufigstes Zeichen ausgegeben
- der ausgegebene Text ist nicht ganz korrekt, falls es mehrere häufigste Zeichen gibt
- das Feld `gezaehlteZeichen` hätte statt in einer Schleife mit einer Bibliotheksfunktion (`memset()`) gelöscht werden können

(c) Das Programm würde `a` ausgeben, da beim Suchen des häufigsten Zeichens ein Zeichen nicht zum neuen häufigsten Zeichen wird, wenn es gleichhäufig wie das bisherige häufigste Zeichen vorkommt.

**(Zusatzaufgabe)** Das Programm verläßt sich darauf, daß der Datentyp `char` den Wertebereich `0–255` hat, was nicht auf allen Plattformen der Fall sein muß. Es wäre besser gewesen, hier ein `unsigned char` zu verwenden, dem das gelesene Zeichen mit expliziter Typumwandlung zugewiesen wird.

Überläufe der Zähler werden nicht getestet. Das führt dazu, daß an plattformabhängigen Stellen falsche Summen berechnet werden. Das Problem kann mit größeren Zählern vermieden werden. Überlauf sollte auch dann noch getestet oder die Begrenzung dokumentiert werden.

In C++ kann man Variablen nicht nur am Funktionsanfang, sondern auch innerhalb von Funktionen deklarieren (hier z.B. `haeufigsteszeichen`). Meist ist es übersichtlicher, alle in einer Funktion verwendeten Variablen am Anfang zu deklarieren.