

Hochschule Deggendorf Dr. Peter Jüttner	
Vorlesung: Einführung in die Programmierung	WS 2012
Übung 2	Termin 9.10.12

Variable und Konstante, Datentypen

Zur Erinnerung ...

Ein C-Programm hat folgende Struktur

```
#include <stdio.h>
```

```
void main(void)
{
    ...
}
```

Variablen werden durch *Typ Variablenname* vereinbart, Zuweisungen an Variable werden durch den = - Operator durchgeführt, z.B.
int i = 100;

1. Eigenschaften von Datentypen

Lassen Sie folgendes Programm laufen, vergleichen Sie die verschiedenen ausgegebenen Werte. Warum wird bei der letzten Zeile etwas „Unerwartetes“ ausgegeben? :

```
#include <stdio.h>
void main(void)
{ char buchstabe;
  buchstabe = 'x';
  printf("Ausgabe des Buchstaben 'x' in verschiedenen Formaten:\n");
  printf("%c\n", buchstabe);
  printf("%d\n", buchstabe);
```

```
  unsigned short kurze_zahl = 100;
  printf("Ausgabe der Zahl 100 in verschiedenen Formaten:\n");
  printf("dezimal:\n");
  printf("%d\n", kurze_zahl);
```

```

printf("hexadezimal:\n");
printf("%x\n", kurze_zahl);
printf("oktal:\n");
printf("%o\n", kurze_zahl);

kurze_zahl = -100;
printf("Ausgabe von -100:\n");
printf("%d\n", kurze_zahl)
}

```

2. Die Größe verschiedener Datentypen

Schreiben Sie ein C-Programm, das für die C-Standarddatentypen, die in der Vorlesung besprochen wurden, die Speichergrößen ausgibt.

Verwenden Sie dazu die Bibliotheksfunktion `sizeof(...)`, an die sie einen Datentyp als Parameter übergeben können. `sizeof` ermittelt, die für den Datentyp benötigten Speicherplatz in **Bytes** (Einheit von 8 Bits).

Beispiel: `sizeof(char)` für den Datentyp `char`

Legen Sie für jeden Datentyp eine Variable vom Typ `short` an

Beispiel: `short char_size = sizeof(char);`

Geben Sie die angelegten Variablen mittels der `printf`-Funktion inkl. eines kurzen Textes aus.

Beispiel:

```

printf("Die Größe von char ist:")
printf("%d\n", char_size);

```

Probieren Sie Ihr Programm aus.

3. Benzinverbrauchsrechner

Schreiben Sie ein Programm, das auf Basis der gefahrenen Kilometer und der Menge des verbrauchten Benzins in Litern der Verbrauch eines Fahrzeugs in l/100km ausgibt.

Vereinbaren Sie dazu zunächst drei Variable des Typs `float` für die gefahrenen Kilometer, die Menge des verbrauchten Benzins und den Verbrauch pro 100km.

Lesen Sie die Werte für Kilometer und Benzin mittels der Funktion `scanf()` ein. Das Einlesen einer Float-Variablen geschieht durch den Aufruf

```
scanf("%f",&variablenname);
```

Berechnen Sie dann den Verbrauch, weisen Sie ihn an die Variable für den Verbrauch zu und geben diese mittels

```
printf("%.2f", variablenname)
```

auf Bildschirm aus. Zur Berechnung des Verbrauchs müssen Sie dividieren. Dies geschieht durch den / Operator, z.B. `variable1 = variable2 / variable3`. *variablenname* steht für einen konkreten Variablennamen

4. Addierer

Schreiben Sie ein C-Programm, das zwei ganze Zahlen einliest und die Summe der Zahlen wieder ausgibt. Einlesen geschieht mittels `scanf("%d",&variablenname)`, Ausgeben mittels `printf("%d", variablenname)`;

Tipp: Schauen Sie die entsprechende Aufgabe vom letzten Mal an.