
Vorlesung 1 (27.10.2017)

1 Formales

Aufbau:

heute (+ Folgetermin 24.11., 16-18 Uhr): Einführung in C sowie: zweimal (MÖGLICHST (?): Fr 3.11., 16-18 und (z.B.) 8.12. 16-18) praktische Übungen im Computerraum W1 0-008

Ende der Vorlesungszeit (12.1. 16-18): Zulassungstest.

ACHTUNG: Muss nicht bei Nachweis von C Kenntnissen (anderer Kurs etc) durchgeführt werden.

19. Februar- 2. März: Kompakturs für 2 Wochen, 4 Tage pro Woche (freier Tage Mittwochs): jeweils: 10-12:00 Vorlesung, 13-17++: Vorlesung/praktische Übungen am Computer

Kreditpunkte: regelmäßige erfolgreiche Teilnahme an Übungen (1-2 Personen pro Gruppe)

Stud.IP: Bitte anmelden (maximale TeilnehmerInnen Zahl: 30)

Tafelanschriften+Programme+weiterführende Texte dort.

2 Einführung in C

Erste Programmiersprache (mit Unterprogrammen etc für das *Analytical Engine* von Charles Babbage)

Augusta Ada Byron King, Countess of Lovelace (*Ada Lovelace*), 1815-1852 (England).

(Zu ihren Ehren später: Programmiersprache ADA).



Erster Compiler (1952 für UNIVAC Computer) von

Grace Brewster Murray Hopper (1906-1992, USA)
(→ später die Programmiersprache COBOL)



2.1 Mein erstes Programm

Inhalt dieses Abschnitts sollen Sie selber am Computer nachvollziehen

Benutze Text Editor für `first.c`

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

```
int main()
{
    printf("my first program");
    return(0);
}
```

Compiliere

```
cc -o first first.c -Wall
```

Starte:

```
first
```

Um mehr über Funktion von `printf` zu erfahren: Schreibe man 3 `printf` in Shell.

2.2 Variablen, Befehle, Ausdrücke

Variablen müssen deklariert werden. Namen (z.B. für Variablen) müssen mit Buchstaben (a...z,A...Z) anfangen, enthalten Buchstaben, zahlen und Unterstrich (_).

```
#include <stdio.h>

int main()
{
    int counter;

    for(counter=0; counter<10; counter=counter+1)
        printf("%d\n", counter);

    return(0);
}
```

Zählen Sie Typen von Variablen auf

Wichtige Typen:

- int
- double
- char
- Arrays=Zeiger
- Funktionen
- Strukturen
- Selbst-definierte Datentypen
- beliebige Kombinationen

Mehrere Variablen auf einmal deklarieren: `double number1, average, sum.`
Wichtige Befehle

- Zuweisung, arithmetische Ausdrücke, z.B.

```
a = b + c;
value = 3.0*sin(angle);
hours = minutes/60;
mod = x % y;
value += delta;
counter++;
number--;
code = input & mask      /* bitwise AND */
code2 = input | mask2    /* bitwise OR */
rest = sequence << n     /* bit shift left by n bits */
```

- for-Schleife

```
for( <initial> ; <condition> ; <command> )
    <command>;
```

- while Schleife

```
while( <condition> )
    <command>;
```

Bsp.:

```
while( counter < 10 )
    printf("counter=%d\n", counter++);
```

- if-Befehl

```
if ( <condition > )
    <command1>;
else /* optional */
    <command2>;
```

Mögliche <conditions>

```

a==b    !!!!!
a!=b
a<b
a<=b
a>b
a>=b

```

a und b können nicht nur Variablen sondern letztlich beliebige Ausdrücke sein.

Bedingungen können per AND (&&) oder/und OR (||) verknüpft werden, z.B.

```

if( (money>100) || ((price < 10) && (money>20)))
    printf("Buy it!!\n");

```

- Blöcke können anstatt eines einzelnen Befehls stehen

```

{
    <command1>;
    <command2>;
    ...
}

```

Achtung

```

double x;
x = 2/3;
printf("%f\n", x)

```

ergibt 0.

Casts übersetzen Datentypen, falls möglich:

```

int z;

z = (int) 3.2;
printf("%d\n", z);

```

ergibt 3;

2.3 Arrays

Arrays = Vektoren oder “Listen” von Variablen. Deklaration: `<type> <name>[<size>]`. Arrays starten mit Index 0, laufen bis `<size>-1`. Bsp.:

```
#include <stdio.h>

int main()
{
    int counter;
    double value[10];
    for(counter=0; counter<10; counter++)
        value[counter]=counter*counter + 0.1;
    return(0);
}
```

Matrix = Array von Arrays: `double matrix[10][10]`
Summe zweier 10x10 Matrizen

```
for(row=0; row<10; row++)
    for(column=0; column<10; column++)
        result[row][column] = matrix1[row][column] + matrix2[row][column];
```

TAKE HOME WORK

Besorgen Sie sich das des C Tutorials vom StudIP, lesen sie die Seiten 1-30.
Vollziehen sie die Programmteile praktisch nach!
