

Fachhochschule Bingen

Programmieren

Break, Switch, Preprozessor

Prof. Dr. Maximilian Mengel,
Professur Programmiermethodik,
Grundlagen der Informatik und Multimedia
Gebäude 1, Raum 212
Tel.: 06721-409 152
E-Mail: mengel@fh-bingen.de

break, continue

- Mittels der **break** und der **continue** Anweisung kann die Abarbeitung einer Kontrollstruktur (z.B. Schleife) abgebrochen werden
- **break**: eine Kontrollstruktur wird verlassen:
 - Es wird mit der nächsten Anweisung nach der Kontrollstruktur fortgefahren
- **continue**: der Anweisungsblock einer Schleife wird verlassen:
 - Es wird mit der Kontrollbedingung der Schleife fortgefahren.

24.10.2003

2

Verzweigungen 2

- Häufig werden if-else-Anweisungen benutzt um eine Entscheidung unter vielen Alternativen zu formulieren

```
if (Ausdruck1)
    Anweisung1;
else if (Ausdruck2)
    Anweisung2;
else if (Ausdruck3)
    Anweisung3;
else if ... ..
```

24.10.2003

3

Switch-Verzweigung

- Die Switch-Anweisung

```
<switch> ::= switch (<Ausdruck>) { {<case-Anweisung>}* [<default>] }
<case-Anweisung> ::= case <Konstante>: {<Anweisung>;}*
<default> ::= default : {<Anweisung>;}*
```

- Beispiel

```
switch (Zeichen){
    case '0': a+=0;
               break;
    case '1': a+=1;
               break;
    ....
    default: a=0;
               break;
}
```

24.10.2003

4

Der C-Preprozessor

- Der C-Preprozessor nimmt textuelle Ersetzungen vor
- Einsatz des Preprozessors:
 - **Symbolische Konstanten**
 - **Einfügen von sogenannten Header-Dateien**
 - Bedingte Übersetzung
 - Macros
 - Vordefinierte Symbole

Der C-Preprozessor

- Die Ersetzungen werden durch sogenannte Direktiven gesteuert
 - **include**
 - **define**
 - **ifdef**
 - **ifndef**
 - **else**
 - **endif**

Der C-Preprozessor

- Vor einer Direktiven steht immer ein #
- Der Preprozessor interpretiert eine ganze Zeile als eine Anweisung
- Soll eine Anweisung für den Preprozessor länger als eine Zeile sein, so muß am Zeilenende ein \ stehen

#define

- Definition von symbolischen Konstanten
<define> ::= # define <Name> [<Ersatz-Text>]
- Beispiel

```
#define PI 3.1415926
#define ZU_LANGER_KONSTANTENNAME_FUER_EINE_ZEILE \
25
#define MAXIMUM 1000
#define M (PI/8.)
```

#include

- Die include-Direktive wird genutzt um sogenannte Header-Dateien einzufügen
- In Header-Dateien sind externe Deklarationen und Funktionen enthalten
- Syntax der include-Direktive:

```
<include> ::= #include <Anwender-Header>|<Bibl-Header>  
<Anwender-Header> ::= "<Datei-Name>"  
<Bibl-Header> ::= <<Datei-Name>>
```

#include

- Mit " " eingebundene Header-Dateien werden in dem lokalen Anwenderverzeichnis gesucht
- Mit < > eingebundene Header-Dateien werden im Installationsverzeichnis des Compilers gesucht

- Beispiel:

```
#include <stdio.h>  
#include <math.h>  
#include "meine_konstanten.h"  
main()  
{.....}
```