

Programmieren in C: Entwicklungswerkzeuge (tools)

19.11.2004

Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagerd



Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 1

Entwicklungswerkzeuge für C

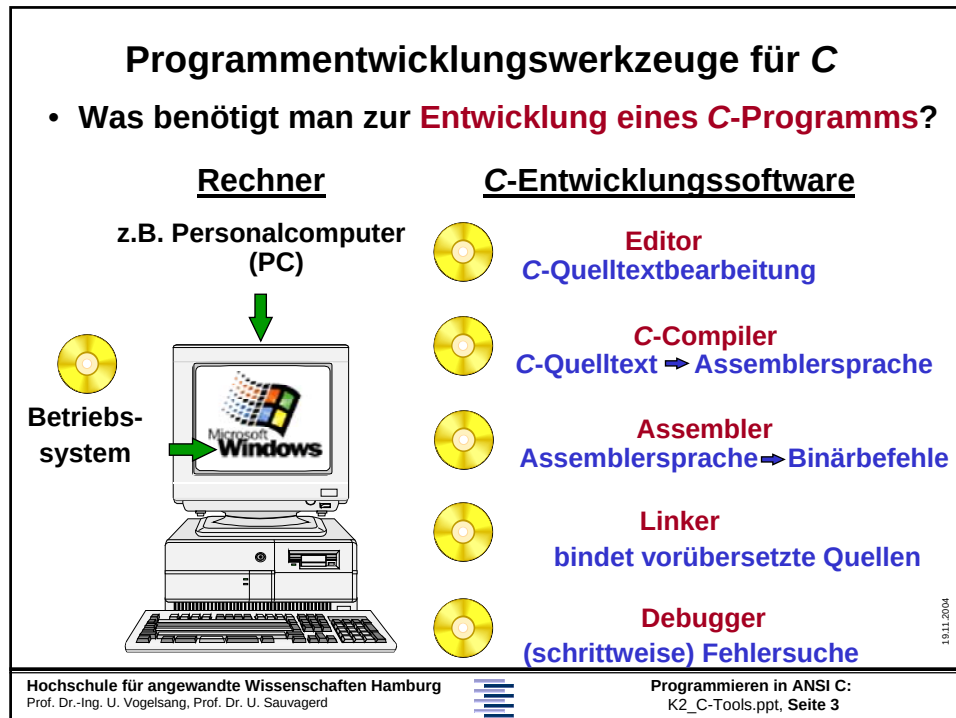
- In diesem Kapitel wollen wir zeigen,
 - ➡ mit welchen Hilfsmitteln C-Programme entwickelt werden
 - ➡ wie ein C-Programm aufgebaut ist

19.11.2004

Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagerd



Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 2



Erstes C-Programm (1)

```
/* Begrueßung  
/* Datei: Wellcome.c  
/* U. Vogelsang, HAW, FB E&I 10.9.03  
#include <stdio.h>  
int main (void)  
{  
    printf("\n\n Herzlich Willkommen im C-Kurs! \n\n\n");  
    return 0 ;  
}
```

```
Kommentar: Erläuterung */  
Kommentar: Dateiname */  
Kommentar: Verfasser, etc.*/  
/* Bibliothek für die Eingabe/Ausgabe */  
/* Vereinbaren eines Hauptprogramms */  
/* Anfang des Hauptprogramms */  
/* Ausgabe: 'Herzl ... */  
/* Rueckgabewert des Hauptprogramms */  
/* Ende des Hauptprogramms */
```

- Das Programm gibt die **ASCII-Zeichenfolge**
Herzlich Willkommen im C-Kurs!

aus.

Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagerd

Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 4

19.11.2004

Erstes C-Programm (2)

```

/* Begrueßung                                Kommentar: Erläuterung */
/* Datei: Wellcome.c                          Kommentar: Dateiname */
/* U. Vogelsang, HAW, FB E&I 10.9.03           Kommentar: Verfasser, etc.*/
#include <stdio.h>                             /* Bibliothek für die Eingabe/Ausgabe */
int main (void)                               /* Vereinbaren eines Hauptprogramms */
{                                              /* Anfang des Hauptprogramms */
    printf("Herzlich Willkommen im C-Kurs! \n"); /* Ausgabe: "Herzl ... */
    return 0;                                /* Rueckgabewert des Hauptprogramms */
}                                              /* Ende des Hauptprogramms */

```

- Der Text zwischen den Zeichen `/*` und `*/` kennzeichnet einen **Kommentar**.
- Ein **Kommentar** wird vom C-Compiler **ignoriert**.
- Außer in Zeichenketten (`" .. "`) werden Leerzeichen und Tabulatoren vom C-Compiler **ignoriert**.

Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagard



Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 5

19.11.2004

Erstes C-Programm (3)

```

/* Begrueßung                                Kommentar: Erläuterung */
/* Datei: Wellcome.c                          Kommentar: Dateiname */
/* U. Vogelsang, HAW, FB E&I 10.9.03           Kommentar: Verfasser, etc.*/
#include <stdio.h>                             /* Bibliothek für die Eingabe/Ausgabe */
int main (void)                               /* Vereinbaren eines Hauptprogramms */
{                                              /* Anfang des Hauptprogramms */
    printf("Herzlich Willkommen im C-Kurs! \n"); /* Ausgabe: "Herzl ... */
    return 0;                                /* Rueckgabewert des Hauptprogramms */
}                                              /* Ende des Hauptprogramms */

```

- Die `#include <stdio.h>` -Anweisung (*include* = *einfügen*) bewirkt, dass die **Bibliotheksfunktionen** für die **Standard Eingabe/Ausgabe (input/output) `stdio.h`** verwendet werden können, wie z.B. `printf`.
- **Bibliotheken** enthalten vorübersetzte Programme, die der **Linker** dann mit einbindet.

Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagard



Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 6

19.11.2004

Erstes C-Programm (4)

```

/* Begrueßung                                Kommentar: Erlaeuterung */
/* Datei: Wellcome.c                          Kommentar: Dateiname */
/* U. Vogelsang, HAW, FB E&I 10.9.03  Kommentar: Verfasser, etc.*/
#include <stdio.h>                            /* Bibliothek für die Eingabe/Ausgabe */
int main (void)                               /* Vereinbaren eines Hauptprogramms */
{                                              /* Anfang des Hauptprogramms */
    printf("Herzlich Willkommen im C-Kurs! \n"); /* Ausgabe: "Herzl ... */
    return 0;                                /* Rueckgabewert des Hauptprogramms */
}                                              /* Ende des Hauptprogramms */

```

- **main** vereinbart ein **Hauptprogramm**.
- **int** zeigt an, dass das Hauptprogramm einen **integer Parameter** (ganze Zahl) an das Betriebssystem zurückgibt.
- **void** bedeutet, dass das Hauptprogramm **beim Start** **keinen** Parameter vom Betriebssystem übernimmt.

Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagerd



Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 7

19.11.2004

Erstes C-Programm (5)

```

/* Begrueßung                                Kommentar: Erlaeuterung */
/* Datei: Wellcome.c                          Kommentar: Dateiname */
/* U. Vogelsang, HAW, FB E&I 10.9.03  Kommentar: Verfasser, etc.*/
#include <stdio.h>                            /* Bibliothek für die Eingabe/Ausgabe */
int main (void)                               /* Vereinbaren eines Hauptprogramms */
{                                              /* Anfang des Hauptprogramms */
    printf("Herzlich Willkommen im C-Kurs! \n"); /* Ausgabe: "Herzl ... */
    return 0;                                /* Rueckgabewert des Hauptprogramms */
}                                              /* Ende des Hauptprogramms */

```

- Die **geschweiften Klammern { }** fassen die dazwischen liegenden Anweisungen zu einem **Block** zusammen.
- Die Klammern zeigen dem Compiler an, wo **das Hauptprogramm beginnt** und wo es **endet**.

Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagerd



Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 8

19.11.2004

Erstes C-Programm (6)

```

/* Begrueßung                                Kommentar: Erläuterung */
/* Datei: Wellcome.c                          Kommentar: Dateiname */
/* U. Vogelsang, HAW, FB E&I 10.9.03           Kommentar: Verfasser, etc.*/
#include <stdio.h>                             /* Bibliothek für die Eingabe/Ausgabe */
int main (void)                               /* Vereinbaren eines Hauptprogramms */
{                                              /* Anfang des Hauptprogramms */
    printf("Herzlich Willkommen im C-Kurs! \n"); /* Ausgabe: "Herzl ... */
    return 0;                                /* Rueckgabewert des Hauptprogramms */
}                                              /* Ende des Hauptprogramms */

```

- Die Anweisung **printf()** (*print formatted*) bewirkt, dass der in **Anführungszeichen** eingeschlossene Text (*string*) auf dem Bildschirm ausgegeben wird.
- Die Zeichenfolge **\n** (*new line*) bewirkt einen **Zeilenvorschub** nach der Ausgabe des Textes.
- Jede **Anweisung** wird mit Semikolon **;** abgeschlossen.

Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagerd



Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 9

19.11.2004

Erstes C-Programm (7)

```

/* Begrueßung                                Kommentar: Erläuterung */
/* Datei: Wellcome.c                          Kommentar: Dateiname */
/* U. Vogelsang, HAW, FB E&I 10.9.03           Kommentar: Verfasser, etc.*/
#include <stdio.h>                             /* Bibliothek für die Eingabe/Ausgabe */
int main (void)                               /* Vereinbaren eines Hauptprogramms */
{                                              /* Anfang des Hauptprogramms */
    printf("Herzlich Willkommen im C-Kurs! \n"); /* Ausgabe: "Herzl ... */
    return 0;                                /* Rueckgabewert des Hauptprogramms */
}                                              /* Ende des Hauptprogramms */

```

- Die Anweisung **return 0** beendet das Programm und führt zur Rückkehr in das Betriebssystem.
- Mit der **ganzzahligen Konstanten (integer constant)** **0** wird dem Betriebssystem angezeigt, dass im Programm kein Fehler aufgetreten ist.

Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagerd



Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 10

19.11.2004

Grundlegende Begriffe in C (1)

➡ C-Anweisungen und Präprozessoranweisungen:

- Man unterscheidet C-Anweisungen (Befehle) von Präprozessoranweisungen.
- Präprozessoranweisungen fangen immer mit **#** an, C-Anweisungen nie.

➡ Leerzeichen (*blank*):

- Leerzeichen außer in Variablennamen, 2-fach-Operatoren (z.B. +=) strings und in gewissen Präprozessoranweisungen werden vom C-Compiler ignoriert.
- Sie dürfen also Leerzeichen einsetzen, um Ihr Programm übersichtlich zu gestalten.

19.11.2004



Grundlegende Begriffe in C (2)

➡ **Ausdruck (*expression*):**
eine Kombination von zulässigen Namen, Konstanten, Operatoren und Klammern.

➡ **Anweisung (*statement*):**

- Aus einem Ausdruck wird eine Anweisung, wenn ein Semikolon folgt.
- Anweisungen werden – in aller Regel – nacheinander (sequentiell) ausgeführt.
- Wenn der Compiler das Semikolon erkennt, so überesetzt er diese Anweisung.
- Danach sind die Vorrangregeln für die nächste Anweisung nicht mehr gültig.

19.11.2004



Grundlegende Begriffe in C (3)

➔ **Block, Verbundanweisung** (*compound statement*):

- In C darf jede Anweisung durch einen Block aus mehreren Anweisungen ersetzt werden.
- Blöcke werden in geschweifte Klammern eingeschlossen:

```
{  
    ...  
}
```

- Auf einen Block folgt kein Semikolon!

Beispiel: übersichtlicher und unübersichtlicher C-Code:



```
printf("Short-Zahl i eingeben(i>=0, i<=99): i=?");scanf("%hi",&i);
```



```
printf ( "Short-Zahl i eingeben ( i>=0, i<=99 ): i = ?" );  
scanf ( "%hi", &i ) ;
```

19.11.2004



Integrierte Entwicklungsumgebung *Microsoft Visual C++*

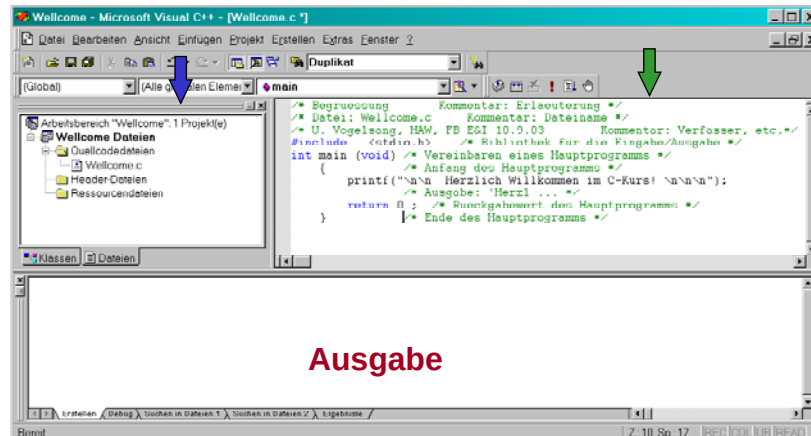
- Wir verwenden im Praktikum die
➔ **Integrierte Entwicklungsumgebung**
(*integrated development environment = IDE*)
Microsoft Visual C++ 6.0
- Bei einer *IDE* werden die Vorgänge
Programmtexteingabe, Compilieren, Linken und Debuggen
effizient innerhalb ein und desselben Programms ausgeführt.
- *Visual C++* erlaubt die Erstellung von Programmen in den **Programmiersprachen C** und in **C++**.
Wir verwenden in dieser Veranstaltung **ausschließlich C**.

19.11.2004



IDE Microsoft Visual C++: Aufteilung

- Das Programm *MS Visual C++* verwendet die 3 Fenster
Projektarbeitsbereich **Quellcode-Editor**

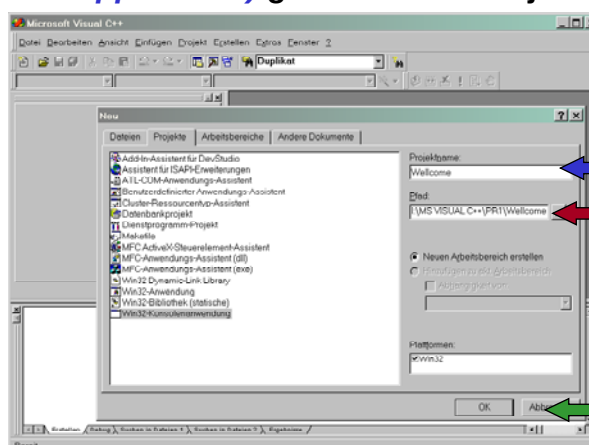


Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagere

Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 15

Microsoft Visual C++: neues Projekt

- Mit **Datei (file) → Neu (new) → Projekte (project) → Win32-Konsolanwendung (WIN32 console application)** gibt man den Projektnamen ein:



Projektname
hier: **Wellcome**

Der Pfad ist
im Praktikum
ein anderer!

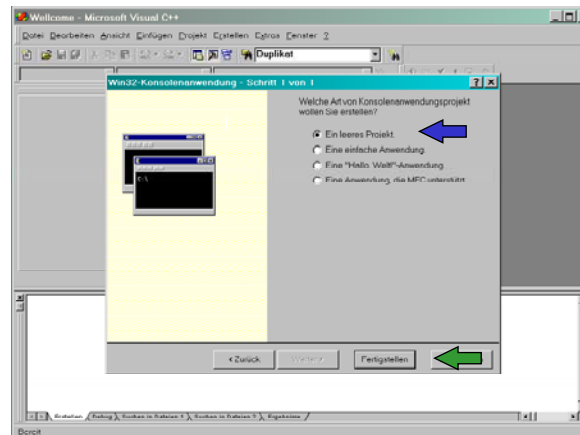
OK beendet
die Eingabe!

Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagere

Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 16

Microsoft Visual C++: Projekt fertigstellen

- Im folgenden Fenster wird **Ein leeres Projekt** (*empty project*) markiert und die Eingabe mit **Fertigstellen** (*finish*) abgeschlossen.



19.11.2004

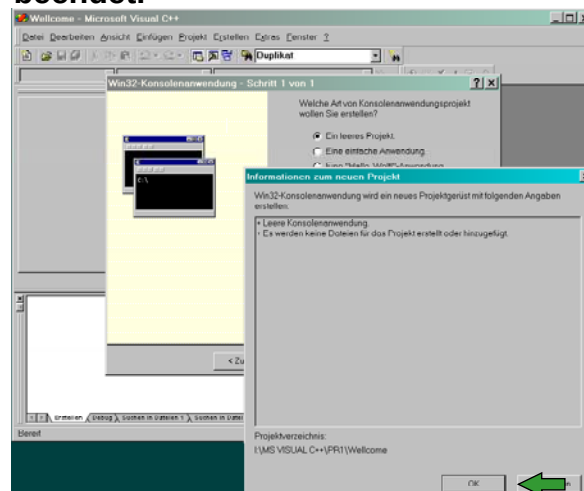
Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagerd



Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 17

Microsoft Visual C++: Informationsfenster

- Das folgende Informationsfenster wird mit **(OK)** beendet.



19.11.2004

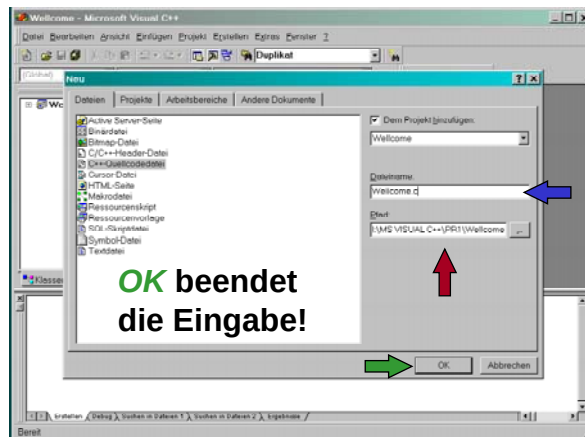
Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagerd



Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 18

Microsoft Visual C++: neuer C-Quellcode

- Nach der Befehlsfolge **Datei** → **Neu** → **Datei** → **C++ Quelldatei** gibt man den Namen der C-Quelldatei mit der Namenserverweiterung .c (!) ein.
Damit wird die Datei **Wellcome.c** erzeugt.



hier: **Wellcome**

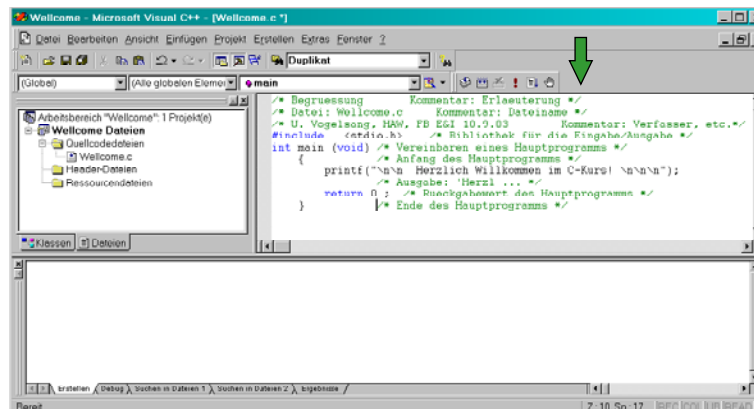
Der Pfad ist der des Projekts!

Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagard

Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 19

Microsoft Visual C++: C-Quellcode eingeben

- Im **Quellcode-Editor-Fenster** gibt man den Programmcode ein: **keine Umlaute, kein ß!**
- Das Programm erkennt die **Kommentare** und **C-Schlüsselwörter** und stellt sie **grün** bzw. **blau** dar.

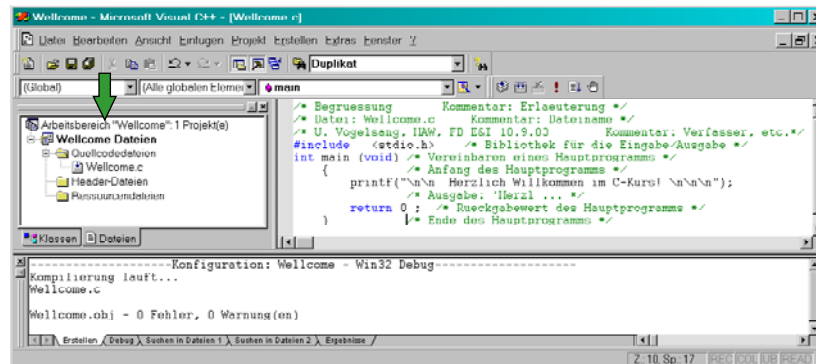


Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagard

Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 20

Microsoft Visual C++: fehlerfreies Programm

- Nach der Kompilierung mit  wird im Ausgabefenster der Erfolg der Operation angezeigt:
Wellcome.obj – 0 Fehler, 0 Warnungen
- Im Arbeitsbereich links erscheint bei korrekter Projekteingabe unter Dateien folgende **Ordnerstruktur**:

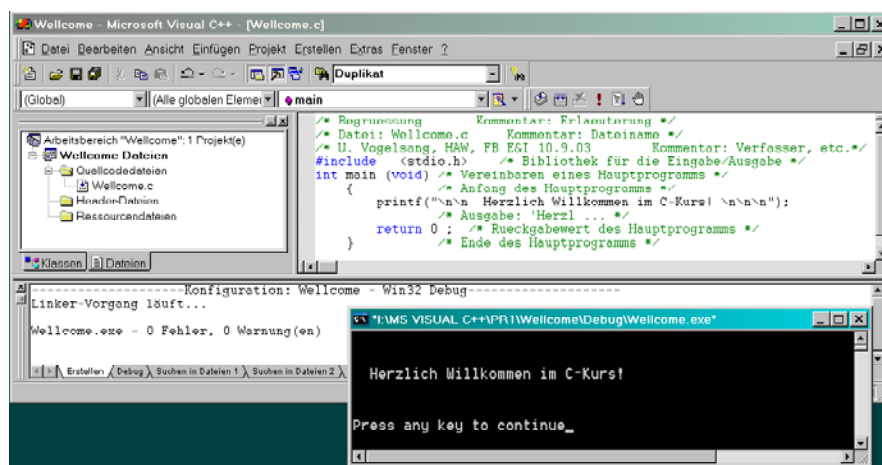


Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagerd

Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 21

Microsoft Visual C++: Programmstart

- Nach  wird **Wellcome.exe** im **DOS-Fenster** gestartet. Das DOS-Fenster wird mit beliebiger Taste beendet.



Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagerd

Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 22

Microsoft Visual C++: Programmausführung (1)

- Bei weiterer Programmausführung kann das Programm direkt gestartet werden, z. B. durch Doppelklick auf *Wellcome.exe* im *Windows Explorer* oder *Windows Arbeitsplatz-Programm*.
- In der Regel läuft das Programm dann so schnell ab, dass man auf dem Bildschirm nur ein Fenster aufflackern sieht.
- Wir werden sehen, wie man durch geeignete C-Befehle erreichen kann, dass das Fenster geöffnet bleibt.
- Häufig enthalten ersten Versionen der C-Programme Fehler.
Wir wollen sehen, wie der *Visual C++*-Kompiler darauf reagiert.

19.11.2004

Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagere



Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 23

Microsoft Visual C++: Fehler im C-Quelltext (1)

- Die folgende Variante des Programms geht nachlässig mit Semikolons um. Die Fehler sind **rot** gekennzeichnet.
- Die Zeilennummern sind **zusätzlich eingefügt** und dürfen im Quelltext **nicht auftauchen!**
- In Zeile 4 ist das Semikolon falsch; in Zeile 8 fehlt es.

```
1. /* Begrueßung           Kommentar: Erläuterung */
2. /* Datei: Wellcome.c    Kommentar: Dateiname */
3. /* U. Vogelsang, HAW, FB E&I 10.9.03  Kommentar: Verfasser,
   etc.*/
4. #include <stdio.h> ; /* Bibliothek für die Eingabe/Ausgabe */
5. int main (void)         /* Vereinbaren eines Hauptprogramms */
6. {                       /* Anfang des Hauptprogramms */
7.     printf("\n\n Herzlich Willkommen im C-Kurs! \n\n\n")
   /* Ausgabe: 'Herzl ... */
8.     return 0 /* fehlendes ; Rueckgabewert des Hauptpr... */
9. }
```

19.11.2004

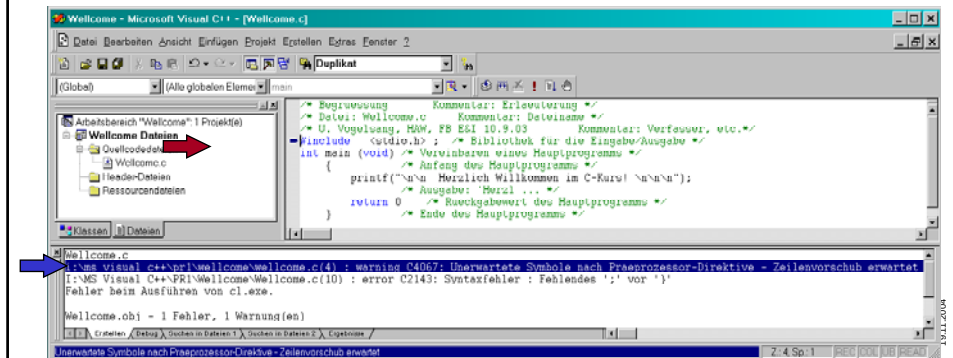
Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagere



Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 24

Microsoft Visual C++: Fehler im C-Quelltext (2)

- Das Programm meldet Fehler und Warnungen.
Alle Fehler und Warnungen müssen beseitigt werden!
- Ein **Doppelklick auf die Fehlermeldung** im Ausgabefenster blendet einen Pfeil im Quelltextfenster ein, der die **fehlerhafte Zeile** markiert.



Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagerd

Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 25

Microsoft Visual C++: Fehler im C-Quelltext (3)

- Die Fehlermeldungen:
I:\MS Visual c++\pr1\wellcome\wellcome.c (4) : warning C4067: Unerwartete Symbole nach Praeprozessor-Direktive - Zeilenvorschub erwartet
I:\MS Visual C++\PR1\Wellcome\Wellcome.c(10) : error C2143: Syntaxfehler : Fehlendes ';' vor '}'
Fehler beim Ausführen von cl.exe.
Wellcome.obj - 1 Fehler, 1 Warnung(en)
- Die Interpretation der Fehlermeldung und das Beseitigen der Fehlerursache (*debug*) erfordert **viel Erfahrung, die man nur durch vieles Üben – auch zu Hause – gewinnt**. Das Praktikum allein reicht dazu nicht aus!!
- Der *Visual C++* ist in seinen Fehlermeldungen erheblich besser als frühere C-Kompiler.

Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagerd

Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 26

Microsoft Visual C++: C- und C++-Dateien

- ☛ Bei **MSVC 6** ist **nicht gleichgültig**, ob die Programmquelltexte mit der Namensweiterung ***.c** oder mit ***.cpp** (= **C++**) angegeben werden.
- ☛ Im **1. Fall** startet **MSVC 6** den **C-Compiler**, im **2. Fall** den **C++-Compiler**.
Man erhält dann unter Umständen **unterschiedliche Fehlermeldungen** und im **schlimmsten Fall** ein **nicht lauffähiges Programm**.
- ☛ Die **Programmiersprache C++** enthält aber im Allgemeinen Erweiterungen und Vereinfachungen der C-Syntax, so dass es in einigen Fällen gleichgültig ist, ob der Quelltext mit ***.c** oder mit ***.cpp** (= **C++**) endet.
- ☛ **Verwenden Sie bei allen Quellcodedateien nur die Dateierdung (extension) *.c !!!**

19.11.2004

Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagerd



Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 27



Übung mit *MS Visual C++* an Ihrem privaten PC oder im PC-Pool: Debug eines C-Programms

Beseitigen Sie bei dem folgenden C-Programm alle Fehler und testen Sie die korrekte Programmausführung mit der Endung ***.c** und ***.cpp**!

```
/* Begrüßung */      Erläuterung */
/* Datei: Übung1.c ; */
/* U. Vogelsang, HAW, FB E&I 11.9.03.*/
#include <stdio.h>      /* Bibliothek für die Eingabe/Ausgabe */
int main (void)        /* Vereinbaren eines Hauptprogramms */
{                      /* Anfang des Hauptprogramms
    printf("\n\n Herzlich Willkommen im C-Kurs! ",
           "\n"); /* Ausgabe: 'Herzl ... */
    return 0 /* Rueckgabewert des Hauptprogramms */
}            // Ende des Hauptprogramms
```

19.11.2004

Hochschule für angewandte Wissenschaften Hamburg
Prof. Dr.-Ing. U. Vogelsang, Prof. Dr. U. Sauvagerd



Programmieren in ANSI C:
K2_C-Tools.ppt, Seite 28

Darstellung von Programmtext und Kommentaren in den Programmbeispielen der Vorlesung

- Bei den vorhergehenden Beispielen haben wir die Farbdarstellungen im C-Quellprogramm entsprechend der Voreinstellung von *MSVC* gewählt.
 - In den folgenden Beispielen wird der **Programmtext** in **blau** dargestellt.
Alle Kommentare (*/* ... */*) werden in schwarz gekennzeichnet.
- Es hat sich herausgestellt, dass diese Darstellung übersichtlicher ist.

19.11.2004

