Moduldokumentation

Modul Systemprogrammierung (syspr)

Simon Wächter

2017

Inhalt

[1 Einleitung 3](#_Toc478062128)

[1.1 Einleitung 3](#_Toc478062129)

[1.2 Lernziele 3](#_Toc478062130)

[1.3 Prüfungen 3](#_Toc478062131)

[2 Woche 1 4](#_Toc478062132)

[2.1 TIOBE Index 4](#_Toc478062133)

[2.2 Grundzüge der Programmierung in einer prozeduralen Sprache 4](#_Toc478062134)

[2.3 Kontrollflüsse und Schleifen 5](#_Toc478062135)

[2.4 Zusammengesetzte Datentypen und Zeiger 5](#_Toc478062136)

[2.5 Aufbau einer Programmierumgebung 5](#_Toc478062137)

[2.6 Gründe für C 6](#_Toc478062138)

[2.7 Dateien in C 6](#_Toc478062139)

[2.8 Das erste C Programm 6](#_Toc478062140)

[2.9 Ein komplexeres C Programm 7](#_Toc478062141)

[2.10 ASCII Tabelle 7](#_Toc478062142)

[3 Woche 2 8](#_Toc478062143)

[3.1 Präprozessor 8](#_Toc478062144)

[3.2 Symbolische Konstanten, Namen und Markos 8](#_Toc478062145)

[3.3 Header Datei als Beispiel 8](#_Toc478062146)

[3.4 Operationen break und continue 8](#_Toc478062147)

[3.5 Übungsbeispiel 9](#_Toc478062148)

[3.6 Operation switch 9](#_Toc478062149)

[3.7 Datentyp struct 10](#_Toc478062150)

[3.8 Funktionsaufrufe 10](#_Toc478062151)

[3.9 Pointer auf ein Objekt im Speicher 11](#_Toc478062152)

[3.10 Übung 11](#_Toc478062153)

[3.11 Fehlersuche und Fehlerbehebung 12](#_Toc478062154)

[4 Woche 3 13](#_Toc478062155)

[5 Woche 4 14](#_Toc478062156)

[5.1 Einführung in die Win32 API 14](#_Toc478062157)

[5.2 Bemerkung zu Standard beim Programmieren 14](#_Toc478062158)

[5.3 Win32 API und C Syntax 14](#_Toc478062159)

[5.4 WinMain als Einstiegspunkt für Windows-Programme 15](#_Toc478062160)

[5.5 Win32 Konsolenprogramm mit Visual Studio 15](#_Toc478062161)

[5.6 Fehlermeldungen mit FormatMessage und GetLastError 16](#_Toc478062162)

[5.7 Win32 Applikation mit Visual Studio 17](#_Toc478062163)

[5.8 Prozesszeiten 18](#_Toc478062164)

[6 Woche 5 21](#_Toc478062165)

[6.1 Kontext von Dateisystemen 21](#_Toc478062166)

[6.2 Anforderungen an ein Dateisystem 21](#_Toc478062167)

[6.3 Aufbau eines Unix Dateisystems 21](#_Toc478062168)

[6.4 Beispiel Linux Dateisystem ext2 22](#_Toc478062169)

[6.5 Gliederung von ext2 22](#_Toc478062170)

[6.6 Dateien unter Linux 22](#_Toc478062171)

[6.7 Index Nodes als sogenannte Inodes 23](#_Toc478062172)

[6.8 Design und Aufbau einer Inode 23](#_Toc478062173)

[6.9 Verzeichnisse 23](#_Toc478062174)

[6.10 Hard- und Symbolic Links 24](#_Toc478062175)

[6.11 Virtuelles Dateisystem 24](#_Toc478062176)

[6.12 VFS Objekte und Datenstrukturen 24](#_Toc478062177)

[6.13 Superblock 25](#_Toc478062178)

[6.14 Operationen auf Superblocks 25](#_Toc478062179)

[6.15 Inodes 25](#_Toc478062180)

[6.16 Struktur struct inode 26](#_Toc478062181)

[6.17 Operationen auf inode 26](#_Toc478062182)

[6.18 Struktur struct dentry 26](#_Toc478062183)

[6.19 Struktur struct file 27](#_Toc478062184)

[6.20 Operationen auf file 27](#_Toc478062185)

[6.21 Grundlegende Dateioperationen 27](#_Toc478062186)

[6.22 System Calls im Linux Dateisystem 28](#_Toc478062187)

[6.23 Library Calls zur Verzeichnismanipulation 28](#_Toc478062188)

[6.24 Beispiel einer Verzeichnismanipulation 28](#_Toc478062189)

[6.25 System Calls zur Dateimanipulation 29](#_Toc478062190)

[6.26 Beispiel einer Dateimanipulation 30](#_Toc478062191)

[6.27 System Calls zur Arbeit im Dateisystembaum 32](#_Toc478062192)

[6.28 Grundlegende API Funktionen in Windows 32](#_Toc478062193)

[7 Woche 6 34](#_Toc478062194)

# Einleitung

## Einleitung

Dieses Dokument stellt die Moduldokumentation für das Modul syspr dar. Allfällige Unterlagen sind im Modulordner zu finden.

## Lernziele

Das Modul beinhaltet folgende Lernziele:

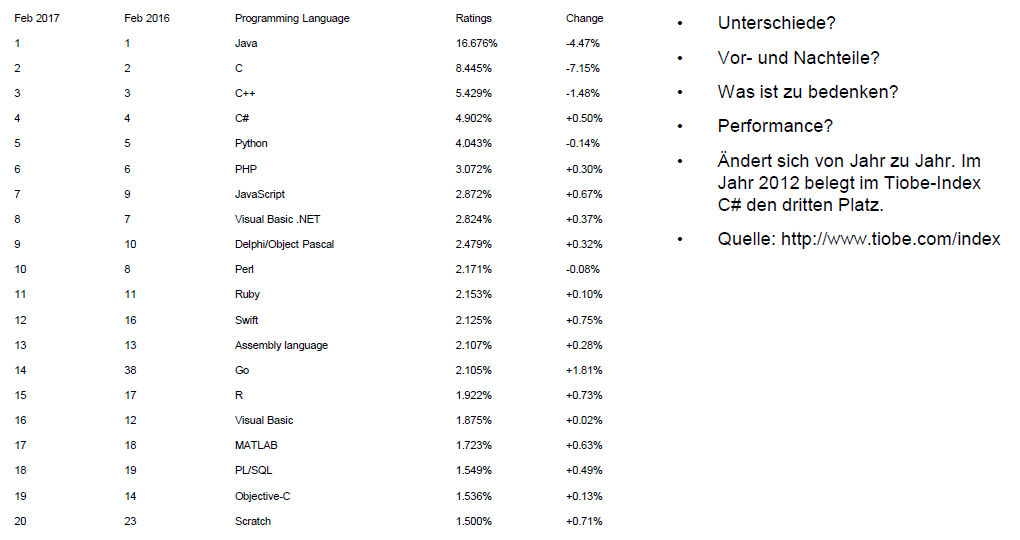
* Kennenlernen des Betriebssystems, beziehungsweise der wesentlichen Systemressourcen aus Sicht des Programmierers:
  + Dateisystem
  + Prozesssteuerung
  + Interprozesskommunikation
  + Synchronisation
* Kennenlernen der Windows API
* Zusammenhang der Windows API mit anderen Programmiersprachen
* Kennenlernen der Programmiersprache C

## Prüfungen

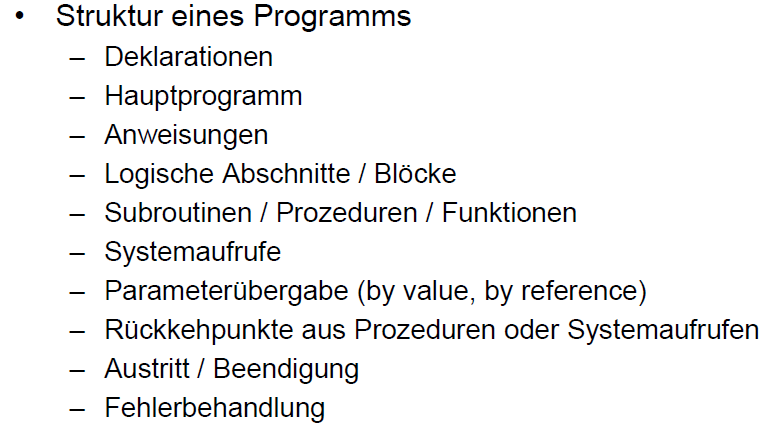
Die Modulnote setzt sich zu 100% aus zwei Semesterprüfungen zu je 50% zusammen.

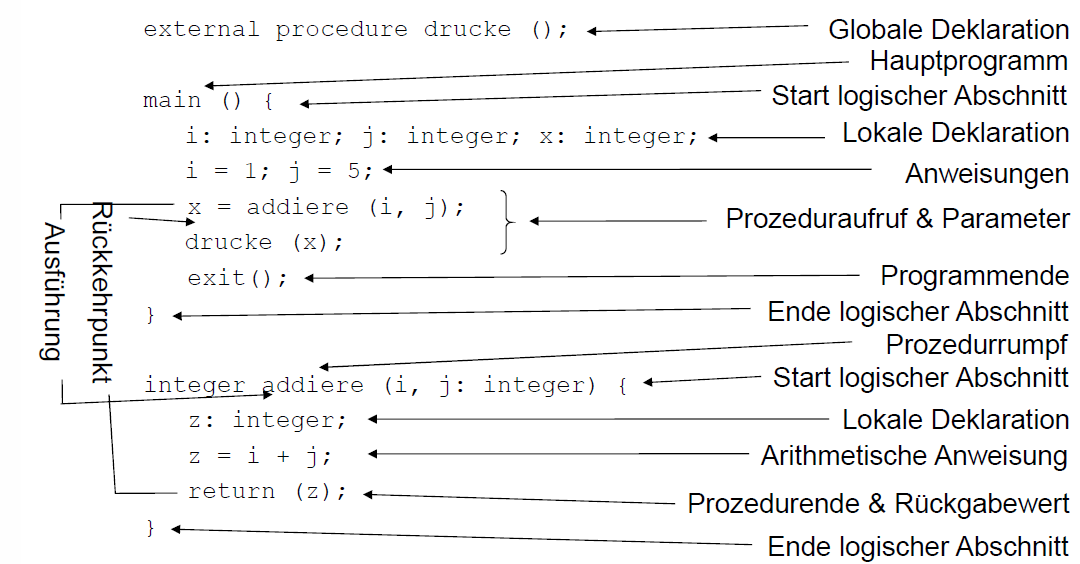
# Woche 1

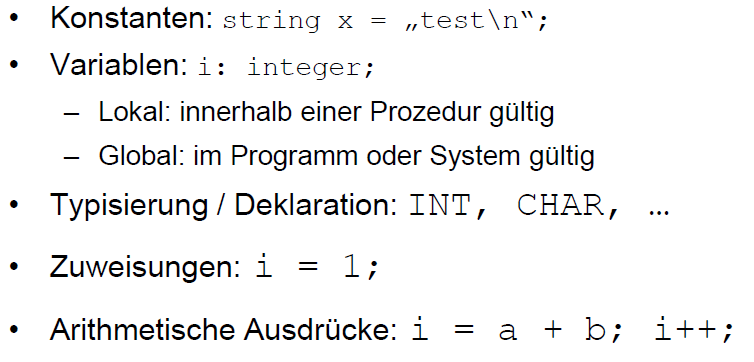
## TIOBE Index



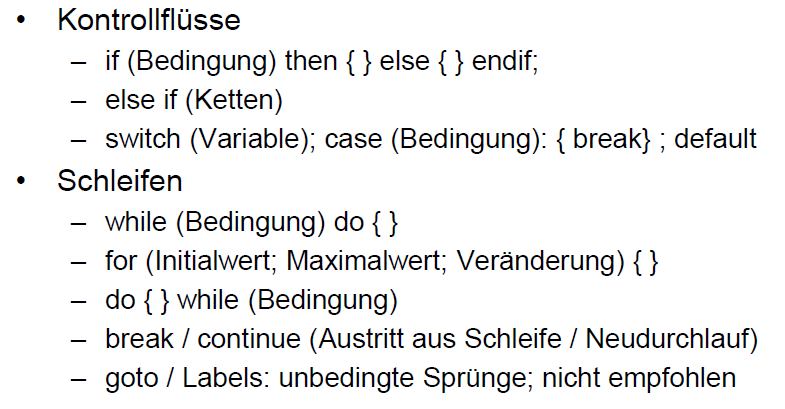
## Grundzüge der Programmierung in einer prozeduralen Sprache



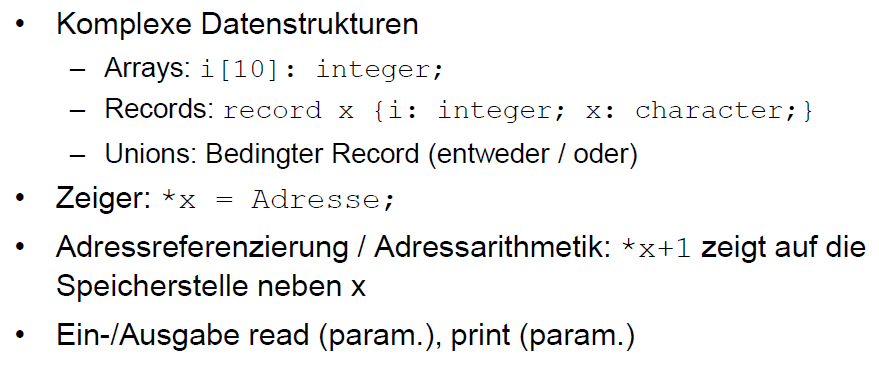




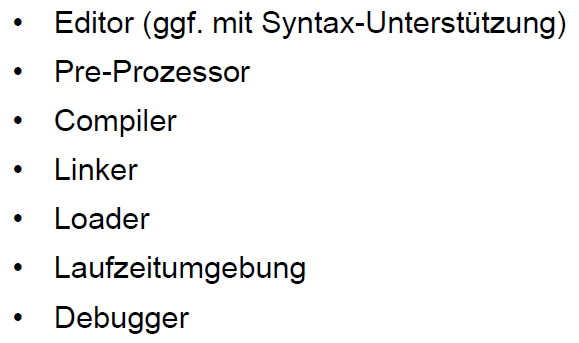
## Kontrollflüsse und Schleifen



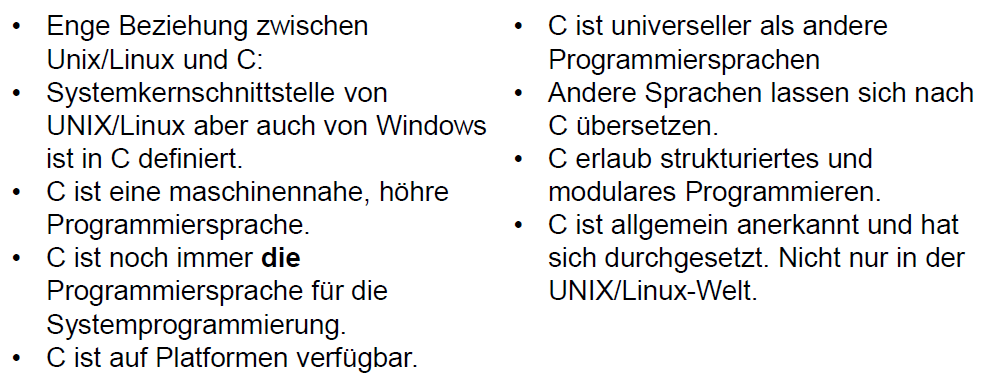
## Zusammengesetzte Datentypen und Zeiger



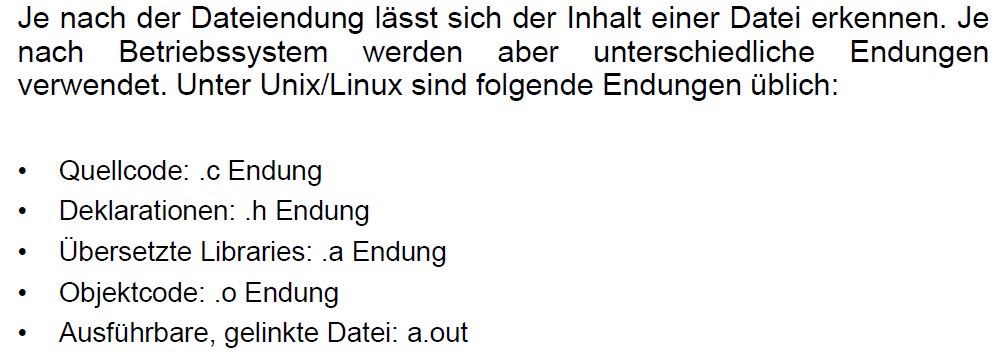
## Aufbau einer Programmierumgebung



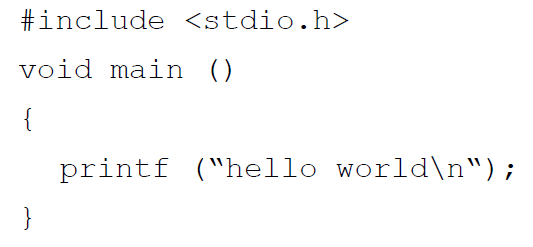
## Gründe für C



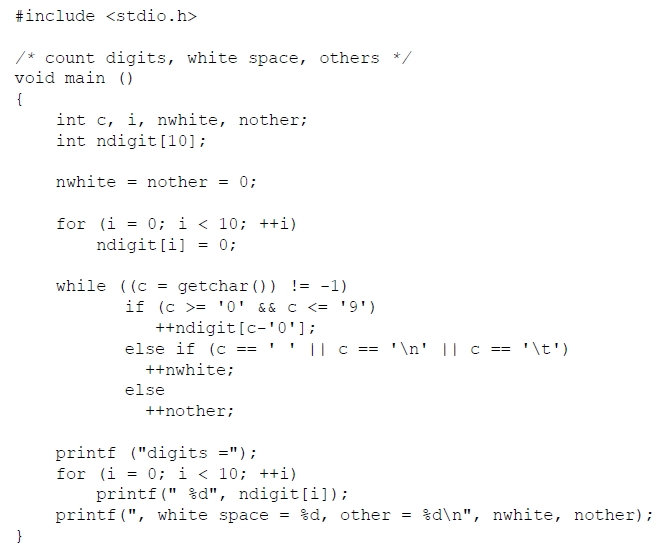
## Dateien in C



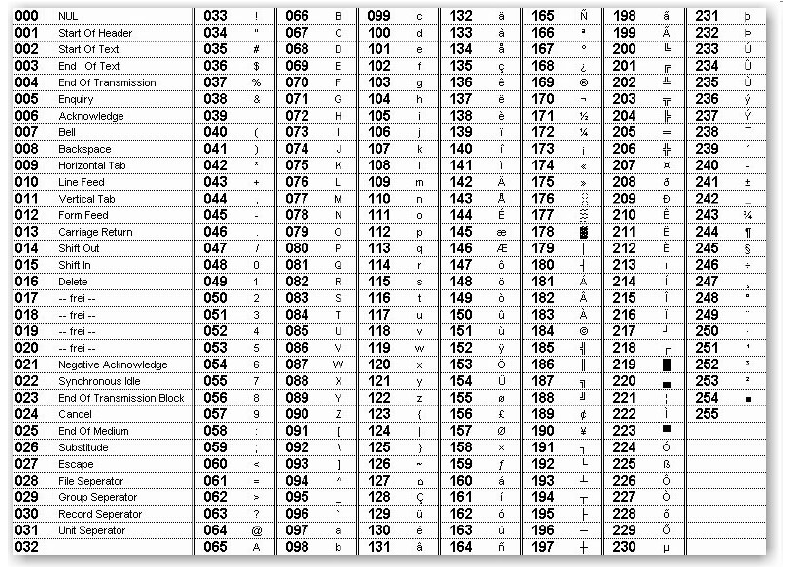
## Das erste C Programm



## Ein komplexeres C Programm

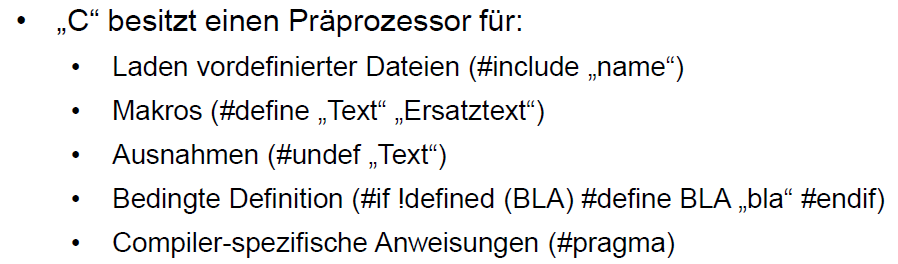


## ASCII Tabelle



# Woche 2

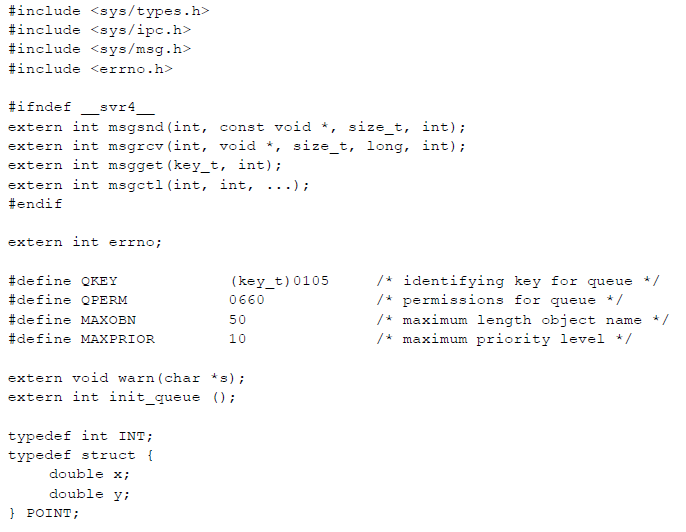
## Präprozessor



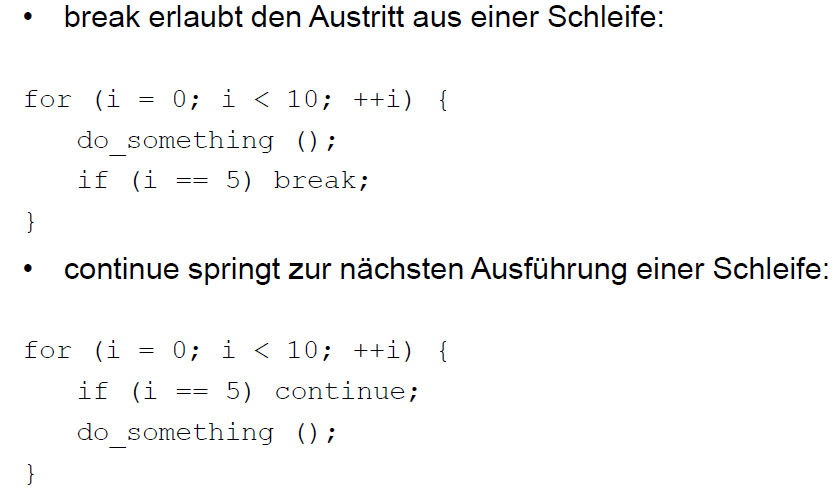
## Symbolische Konstanten, Namen und Markos



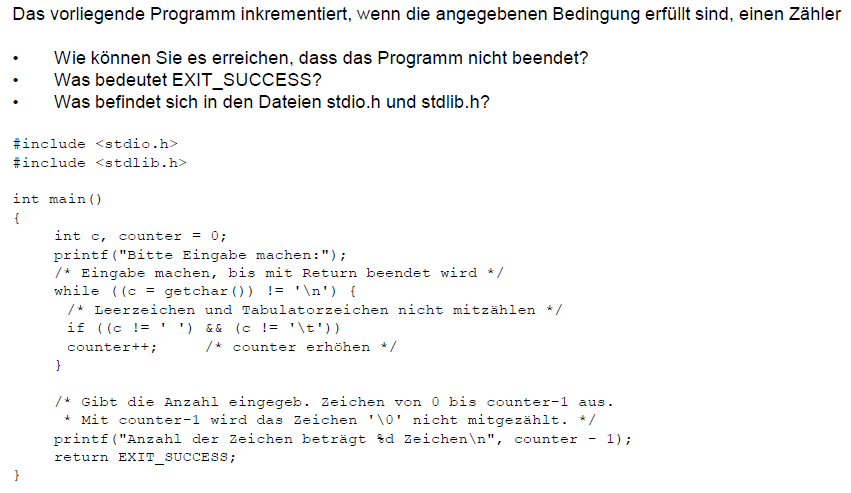
## Header Datei als Beispiel

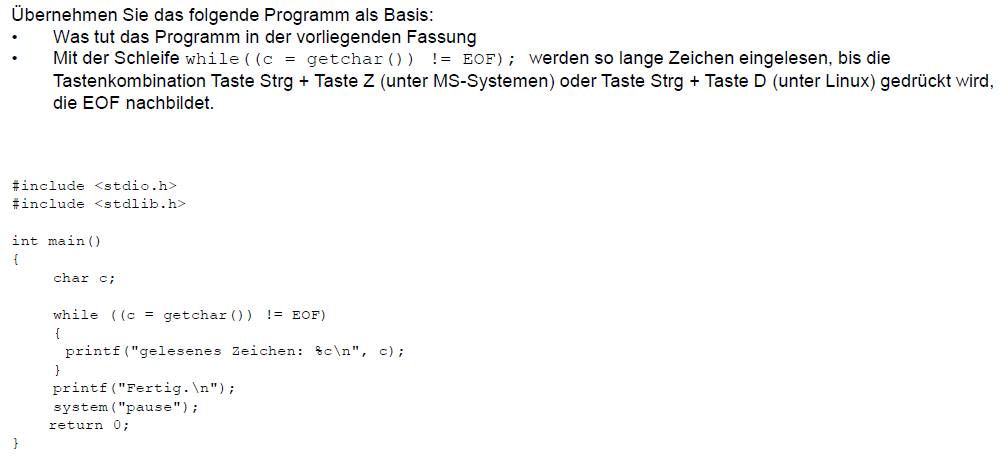


## Operationen break und continue

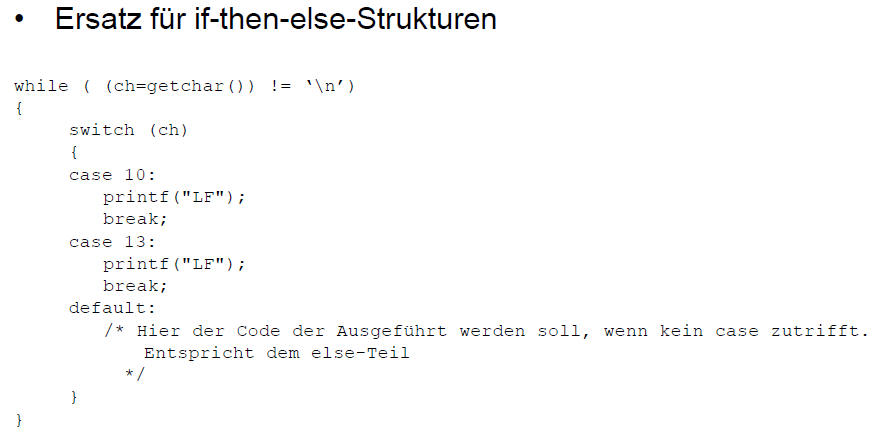


## Übungsbeispiel

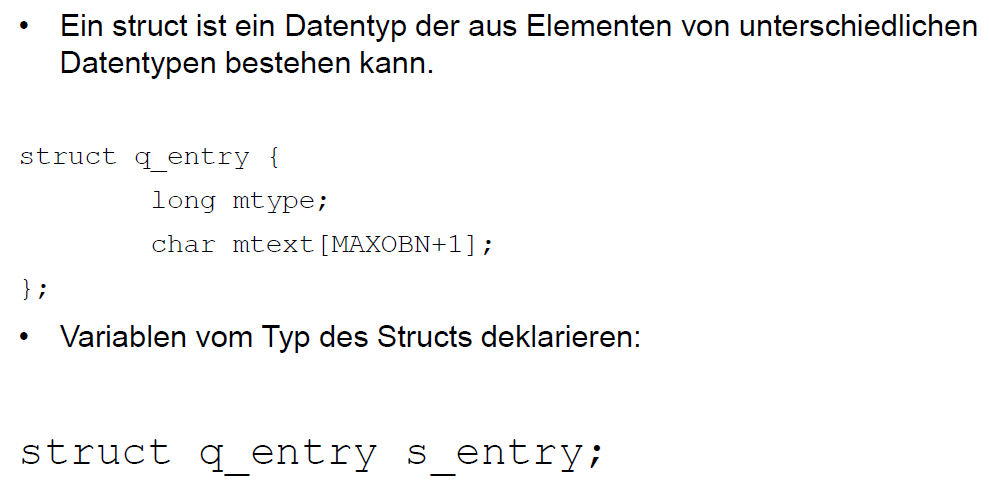


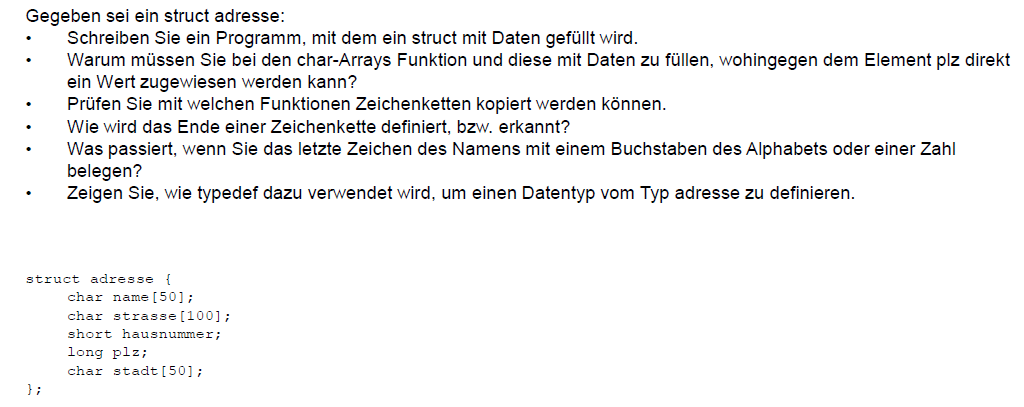


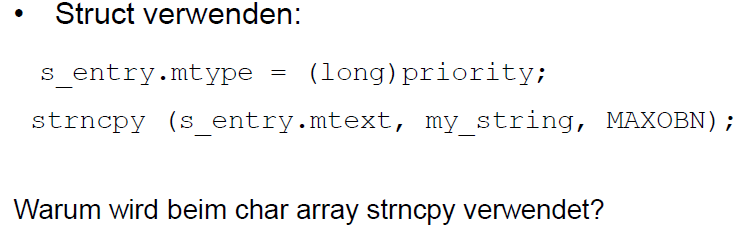
## Operation switch



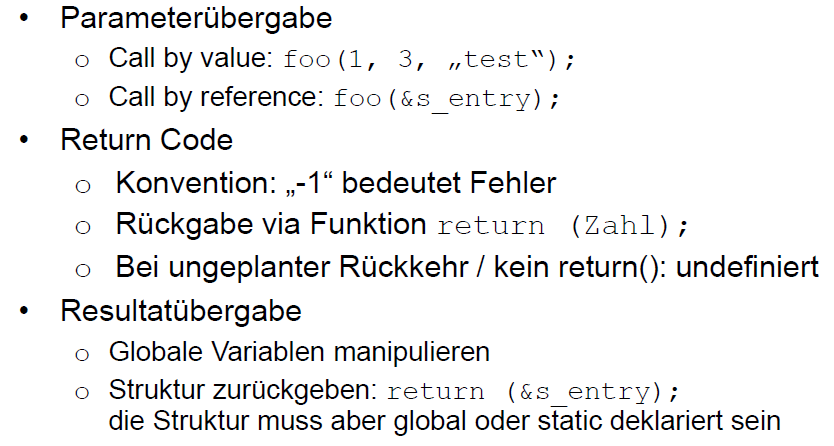
## Datentyp struct



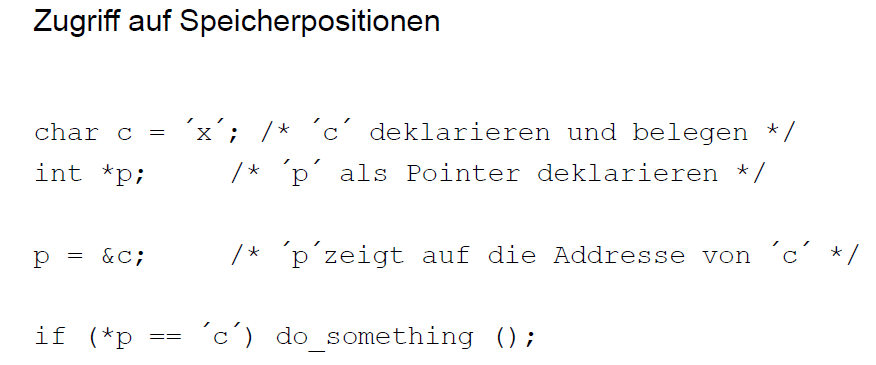


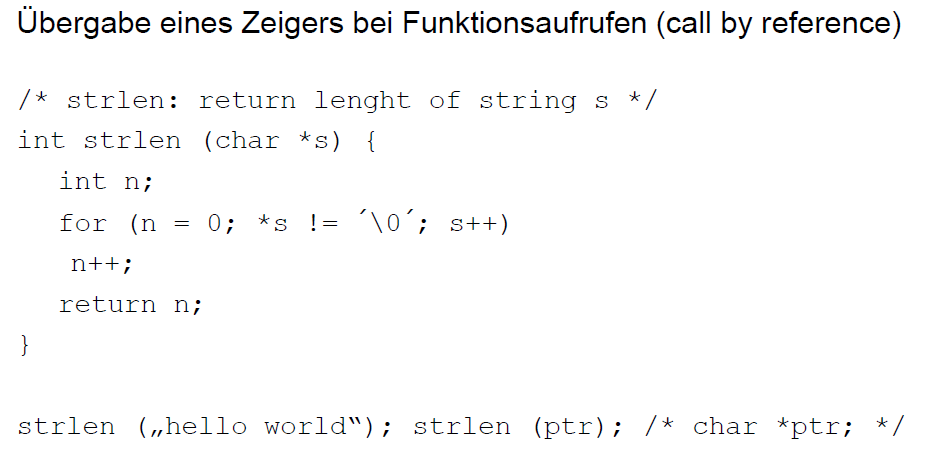


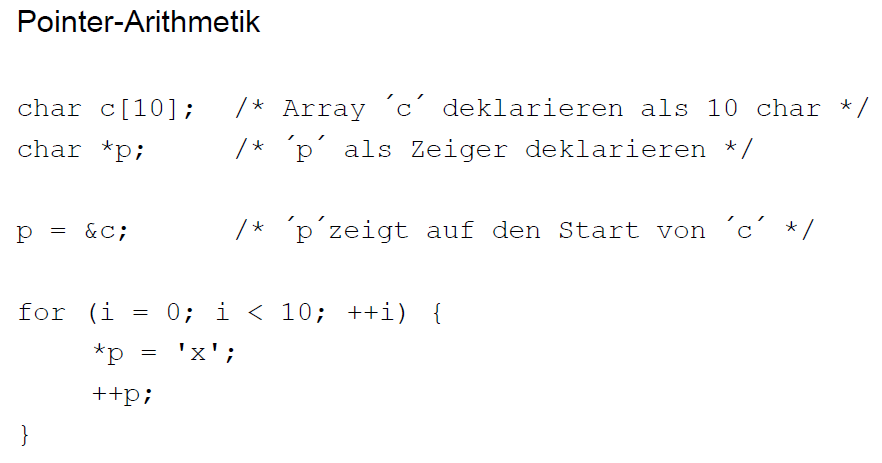
## Funktionsaufrufe



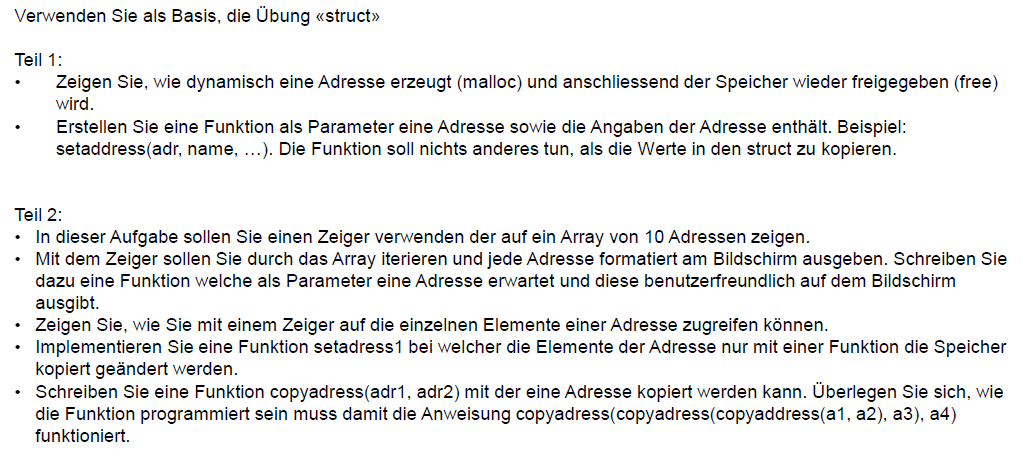
## Pointer auf ein Objekt im Speicher



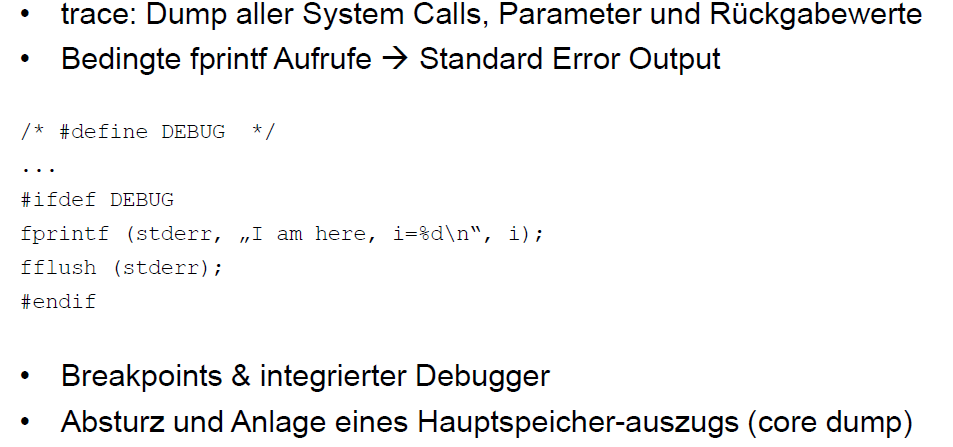




## Übung



## Fehlersuche und Fehlerbehebung

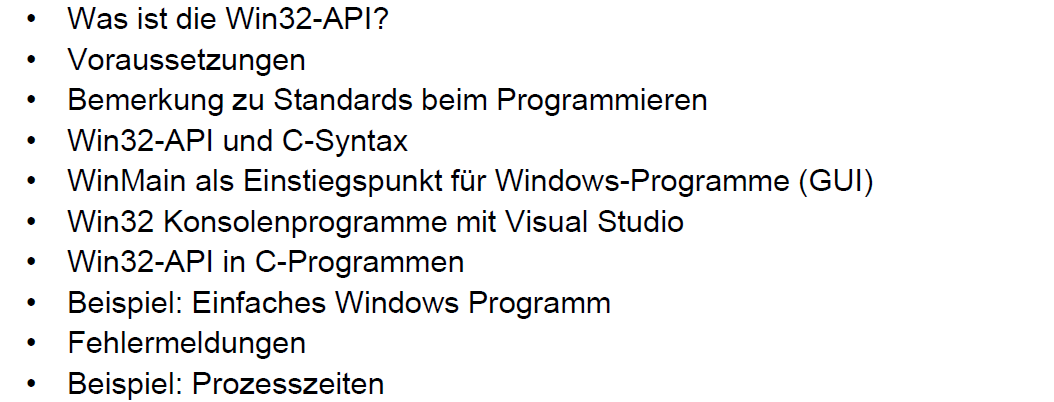


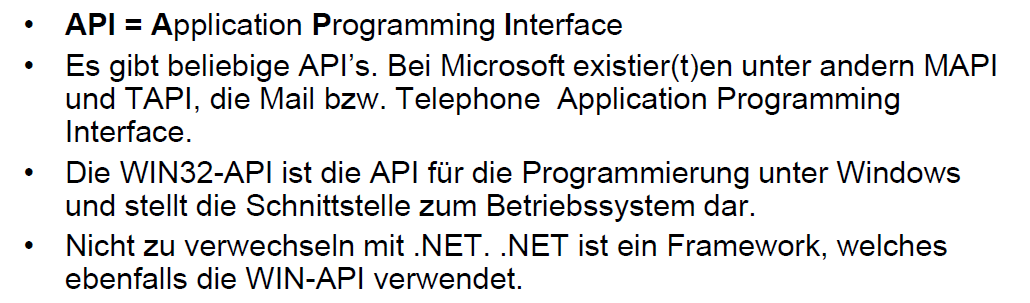
# Woche 3

In Woche 3 fiel krankheitsbedingt der Unterricht aus.

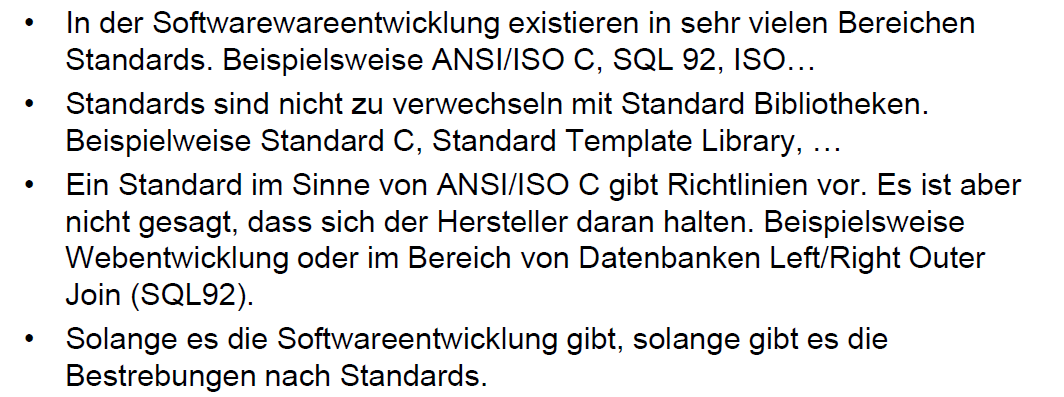
# Woche 4

## Einführung in die Win32 API

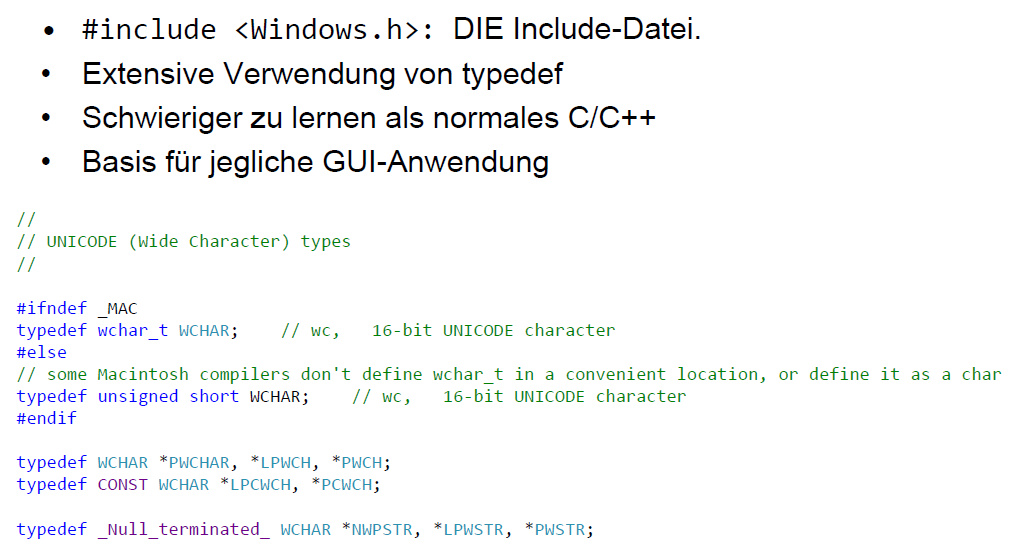




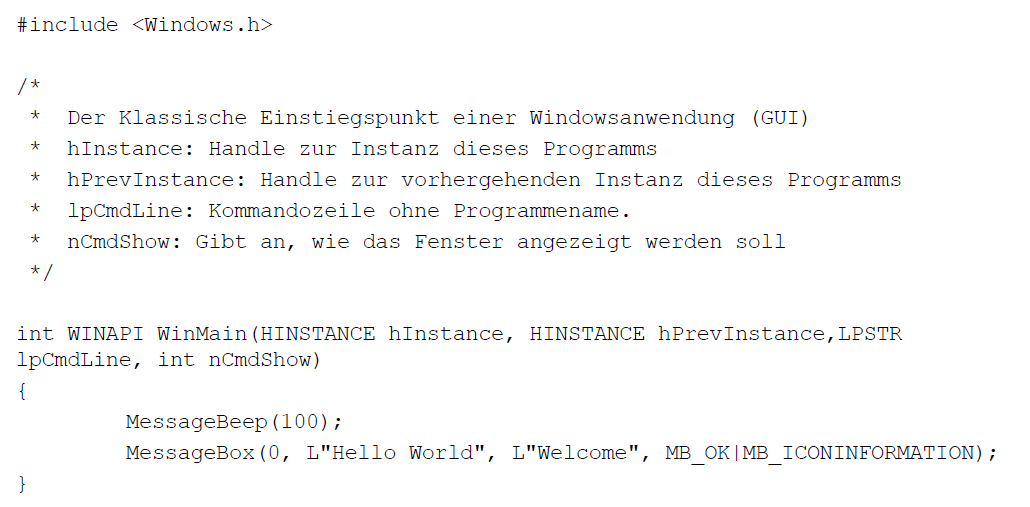
## Bemerkung zu Standard beim Programmieren



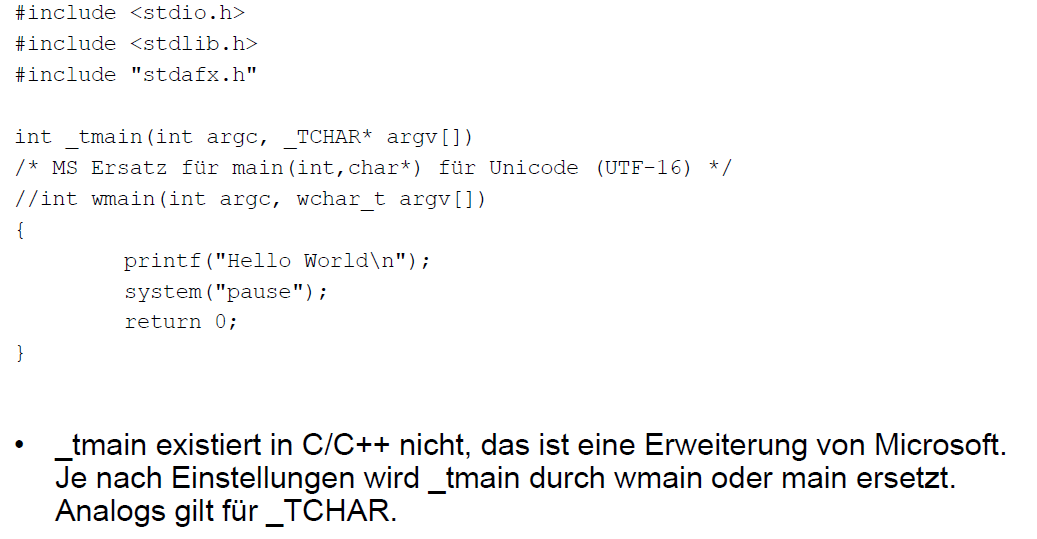
## Win32 API und C Syntax

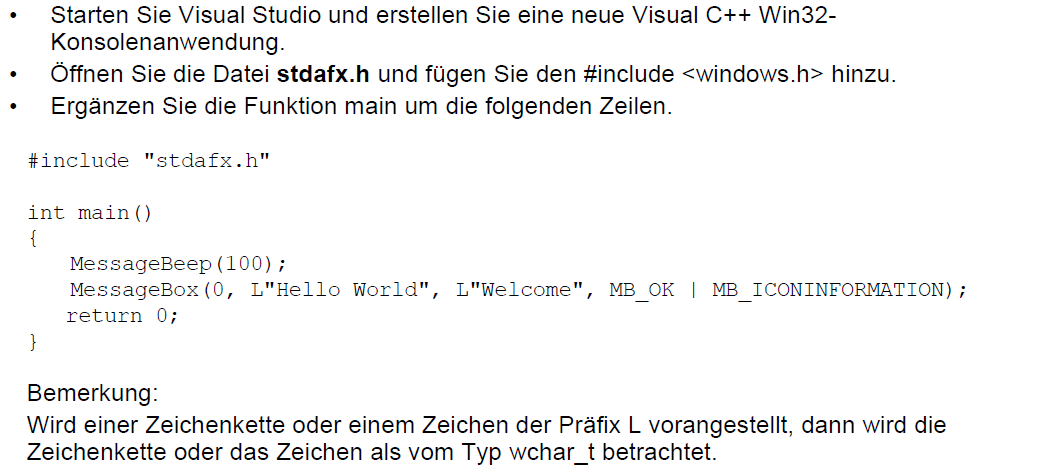


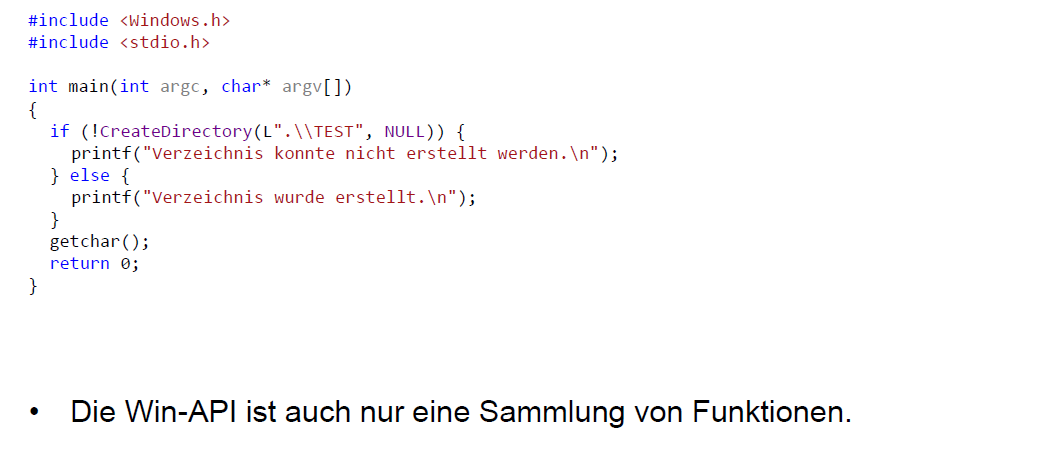
## WinMain als Einstiegspunkt für Windows-Programme



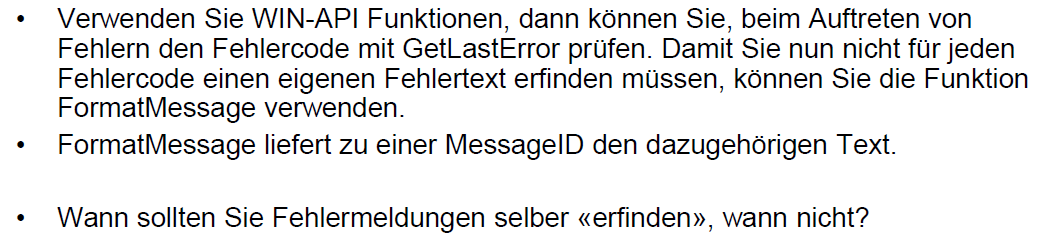
## Win32 Konsolenprogramm mit Visual Studio

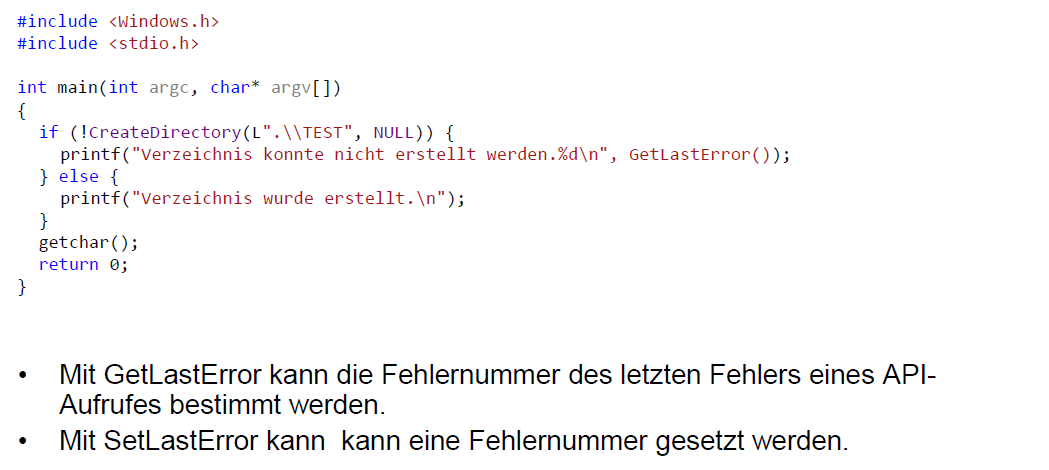


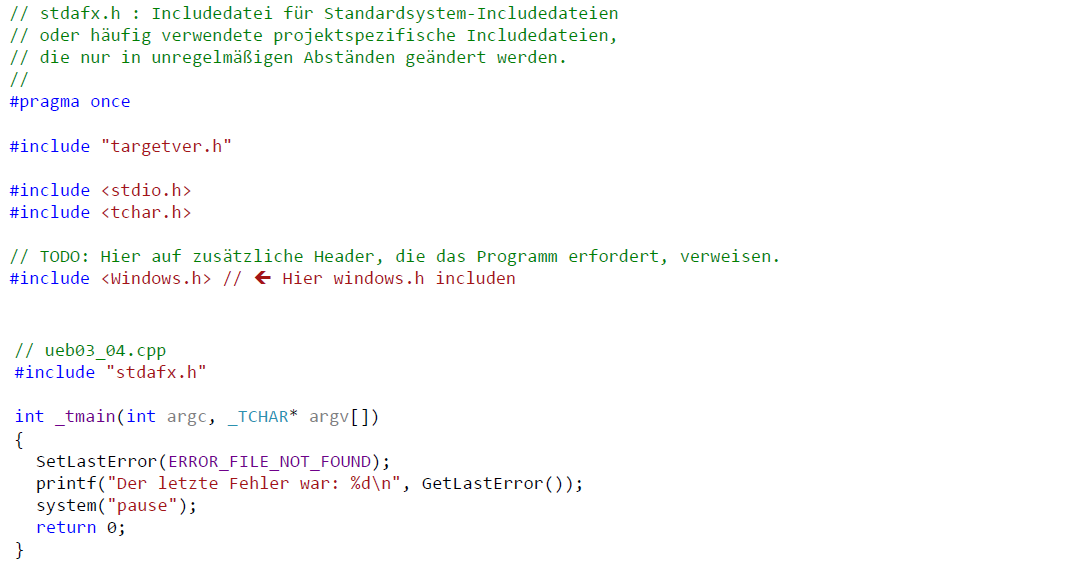




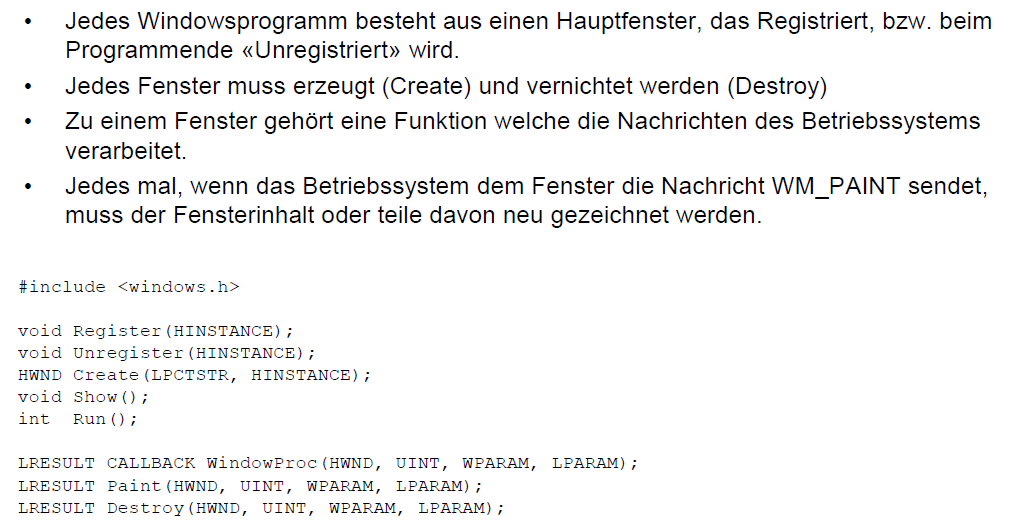
## Fehlermeldungen mit FormatMessage und GetLastError

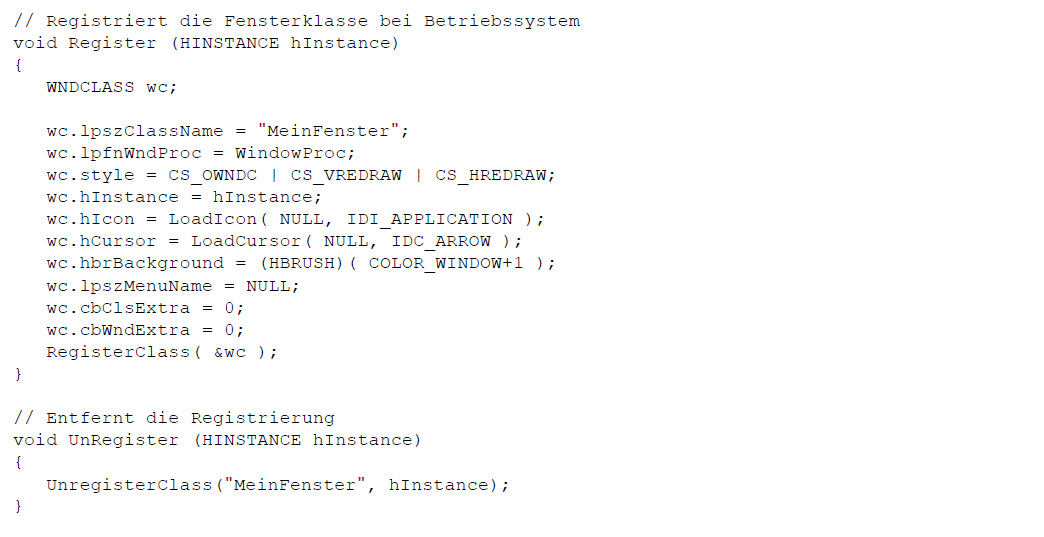






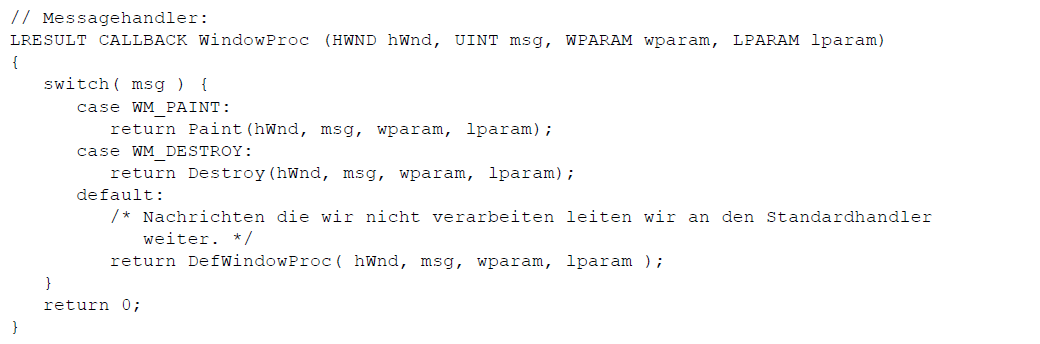
## Win32 Applikation mit Visual Studio

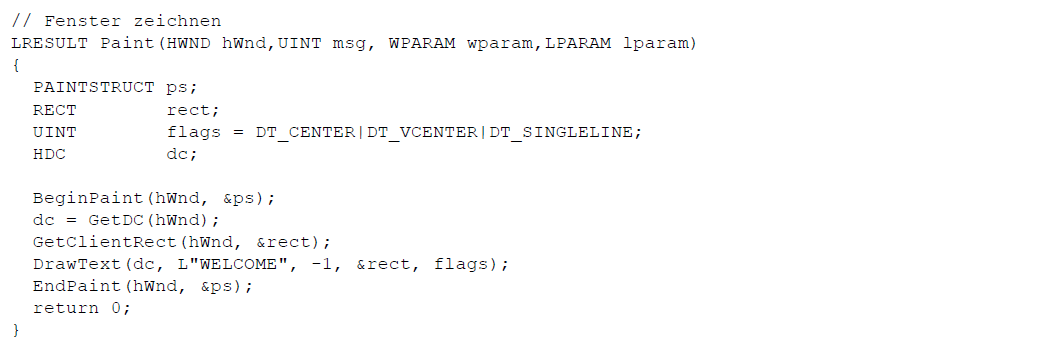


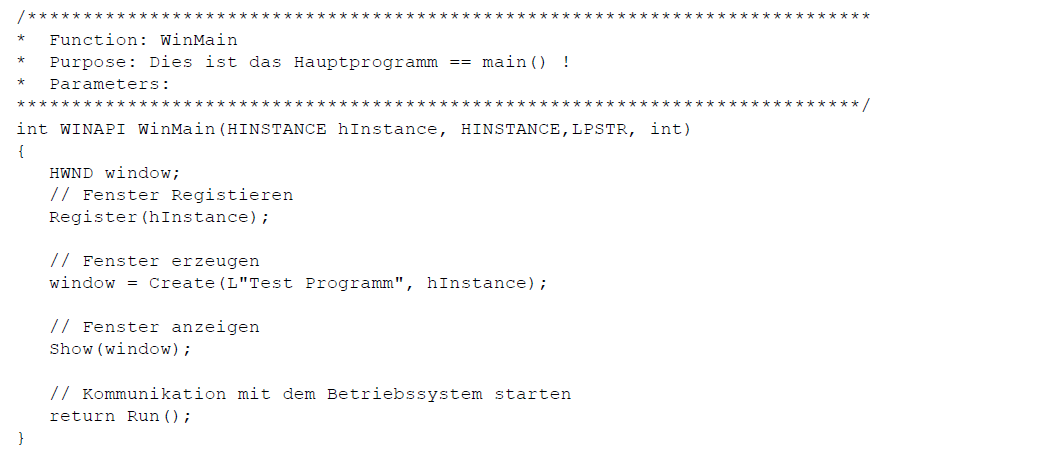




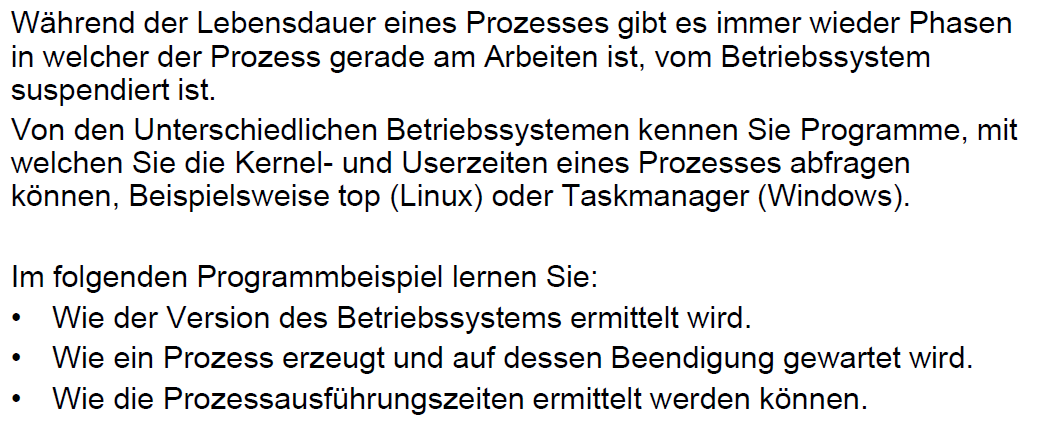


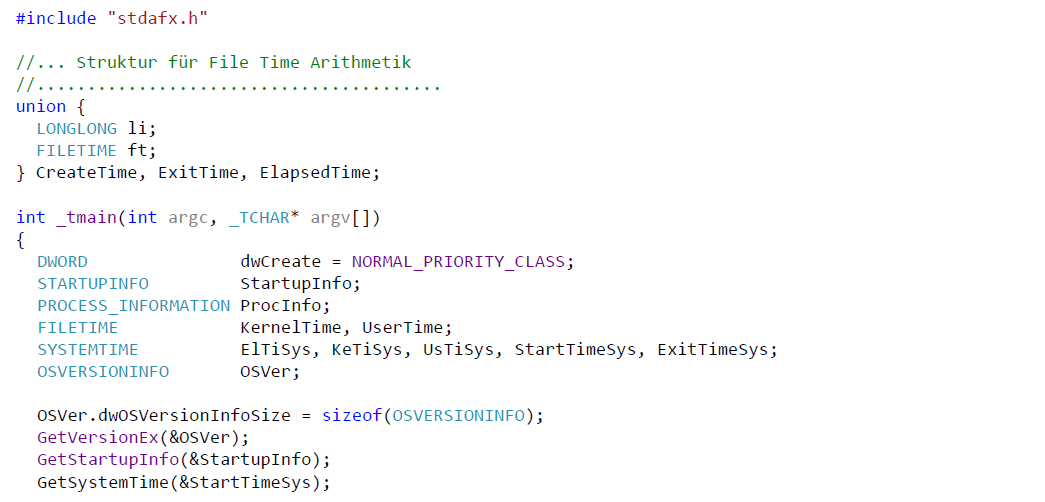


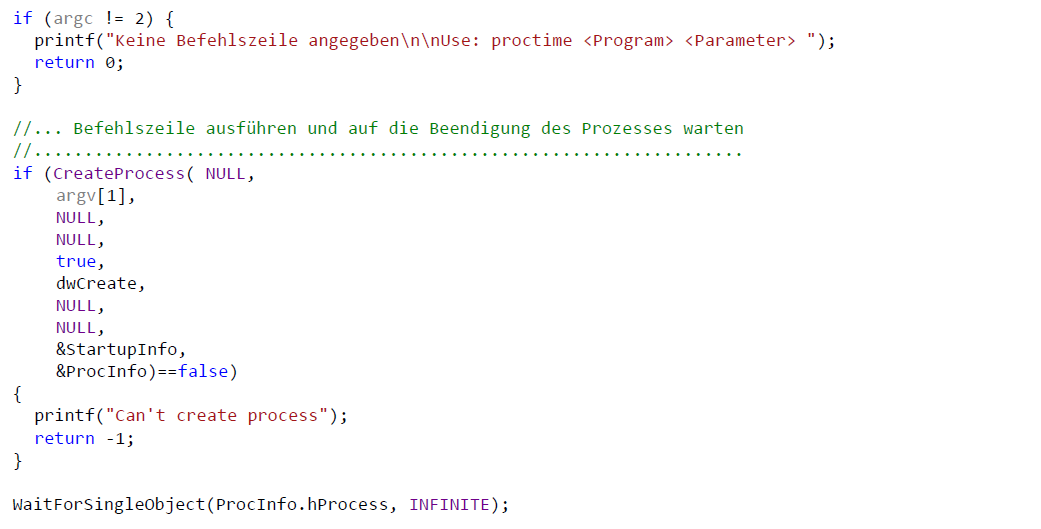


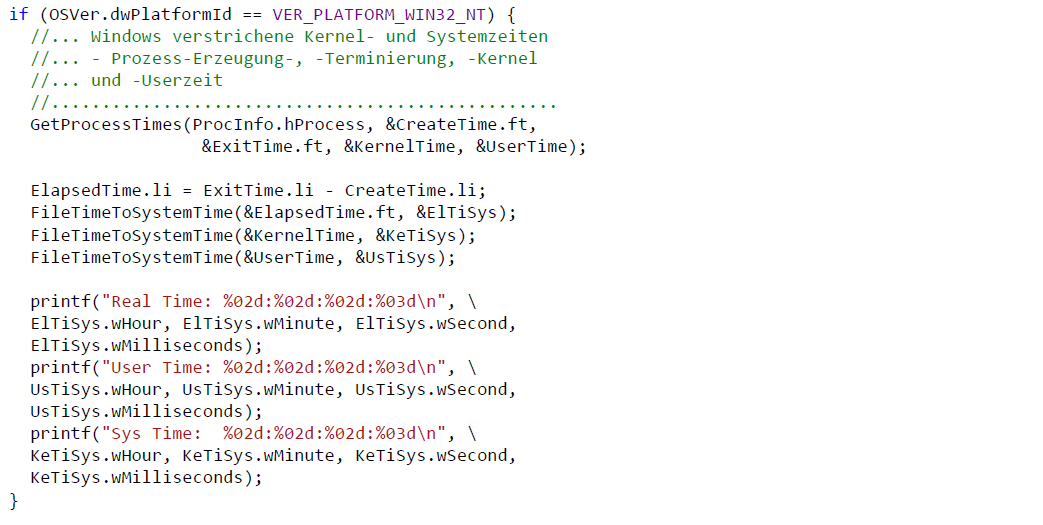


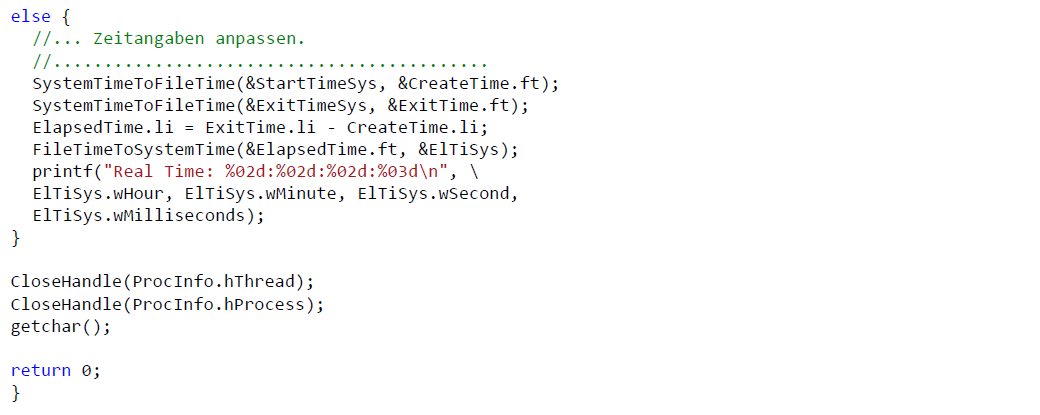
## Prozesszeiten





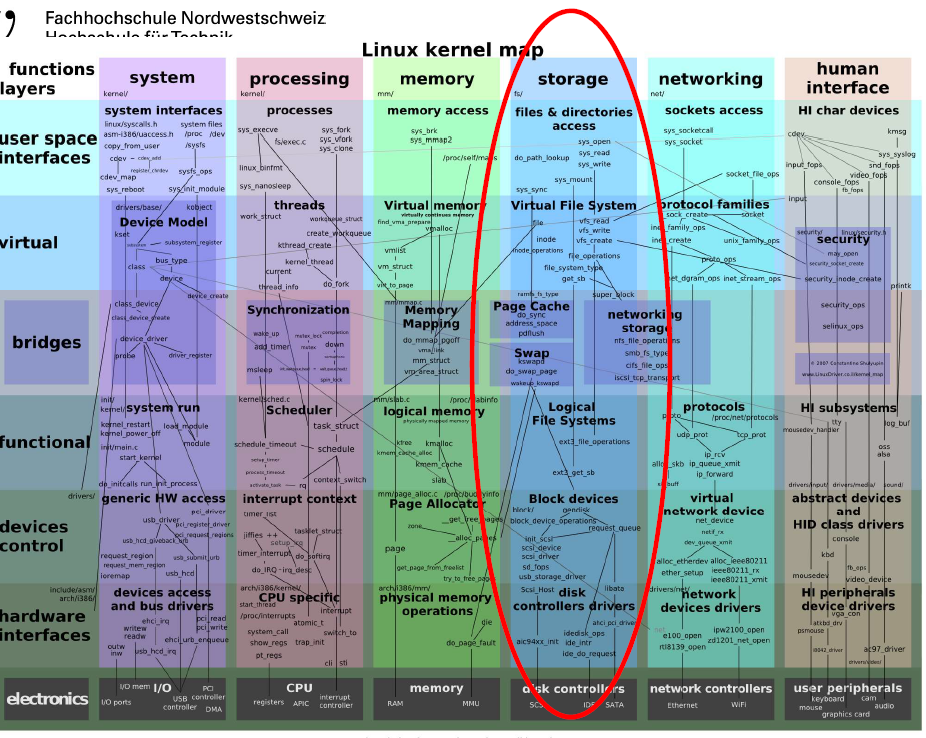




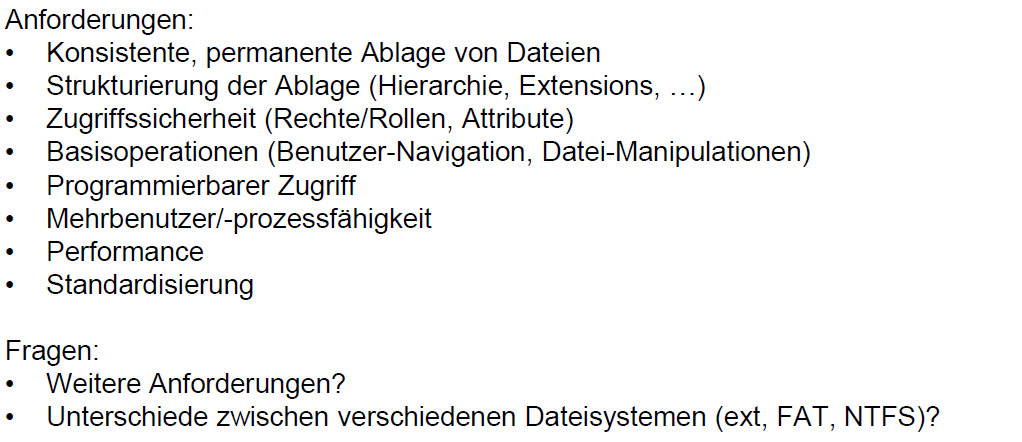


# Woche 5

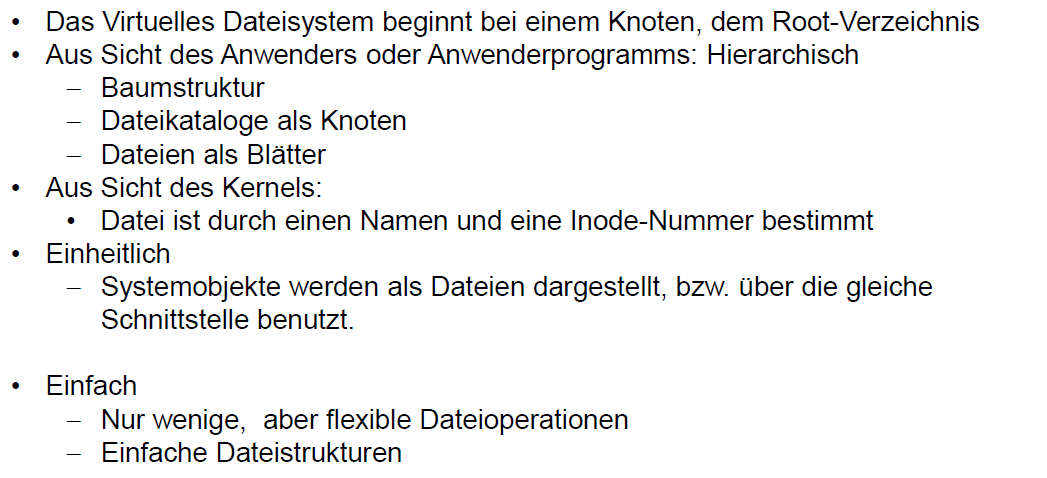
## Kontext von Dateisystemen



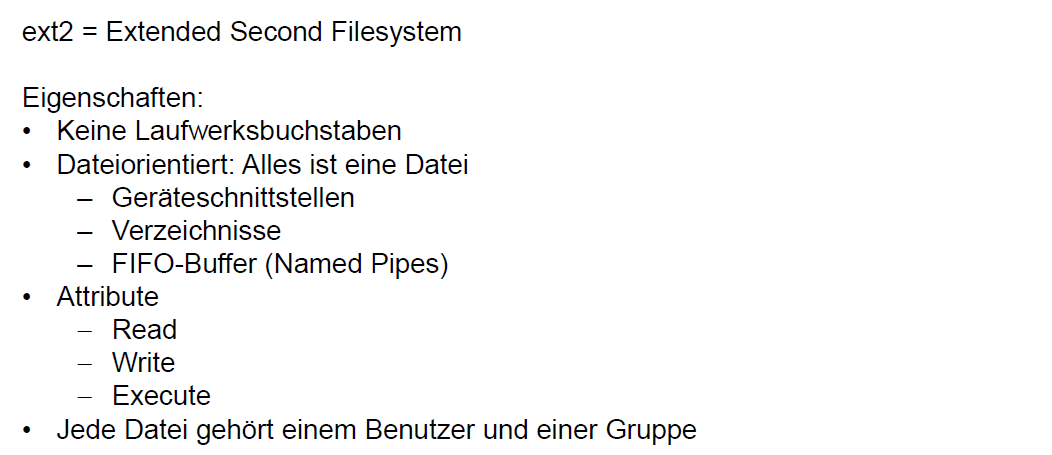
## Anforderungen an ein Dateisystem



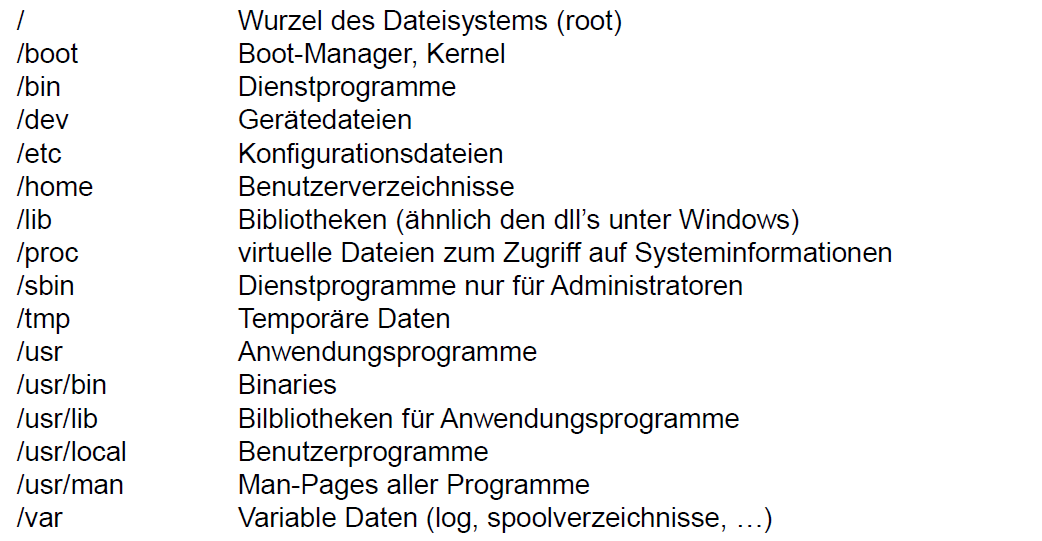
## Aufbau eines Unix Dateisystems



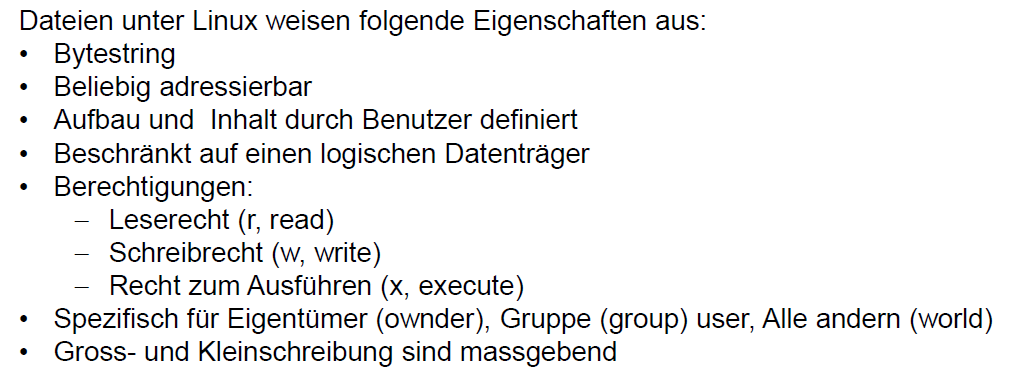
## Beispiel Linux Dateisystem ext2



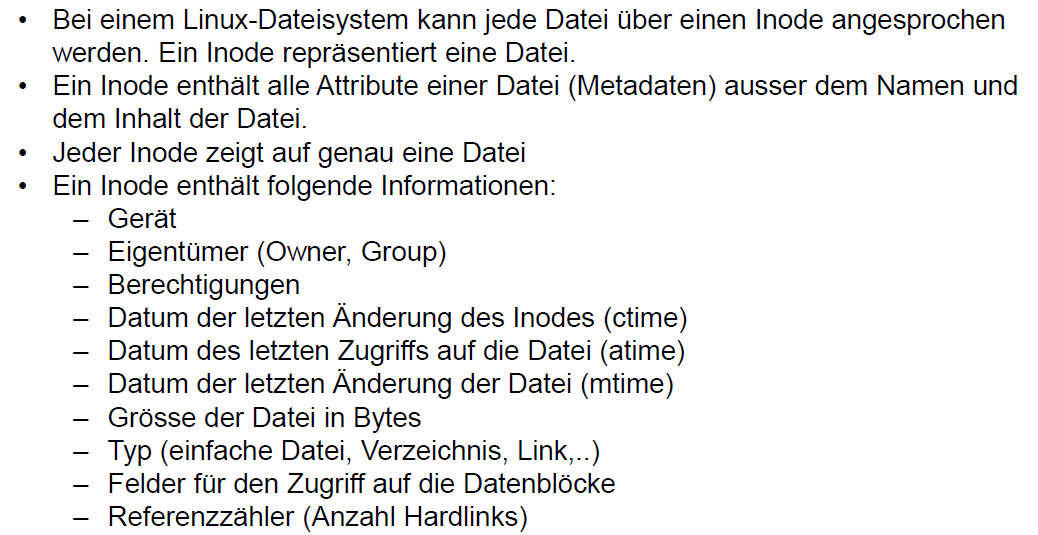
## Gliederung von ext2



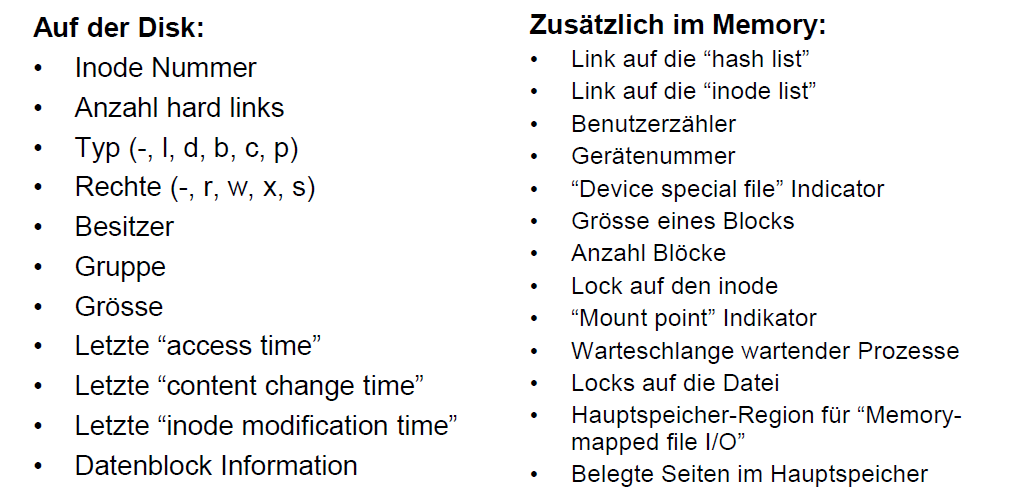
## Dateien unter Linux



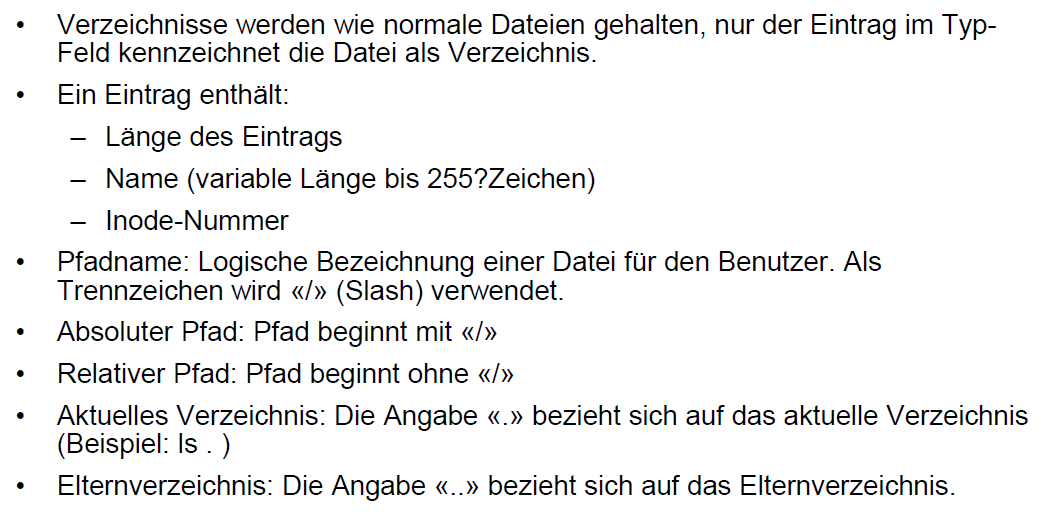
## Index Nodes als sogenannte Inodes



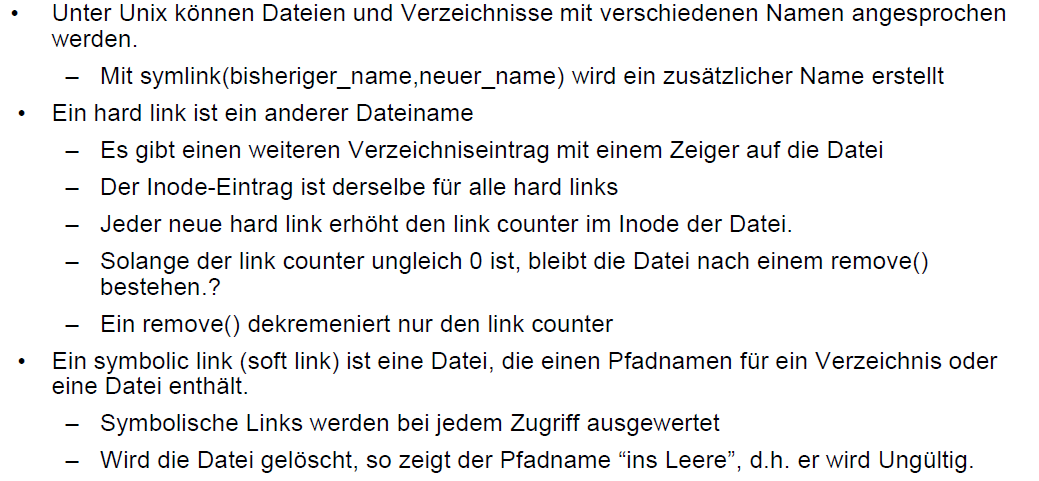
## Design und Aufbau einer Inode



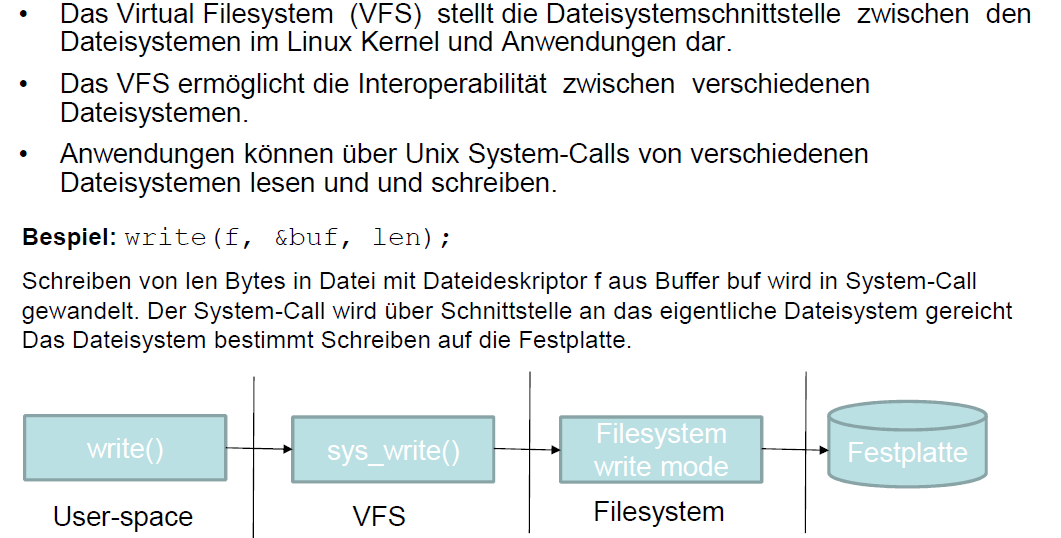
## Verzeichnisse



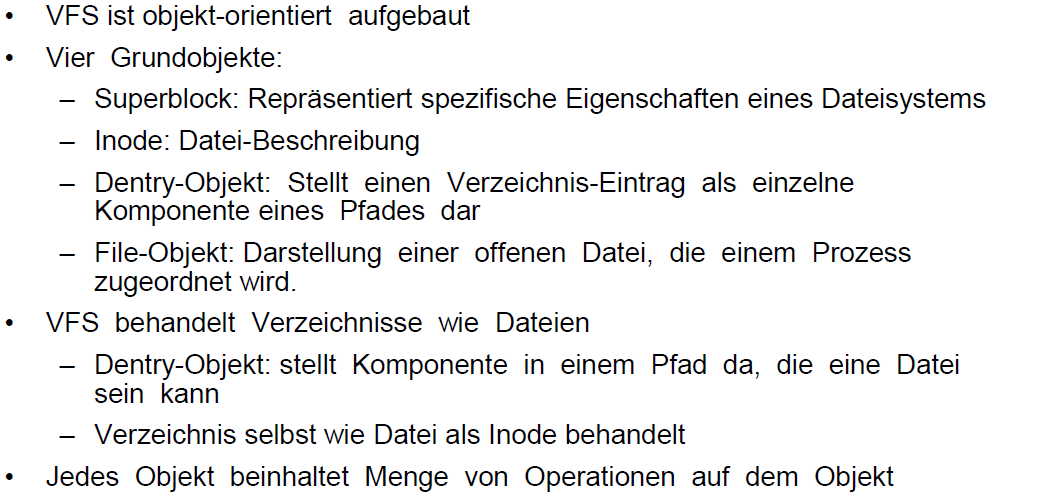
## Hard- und Symbolic Links



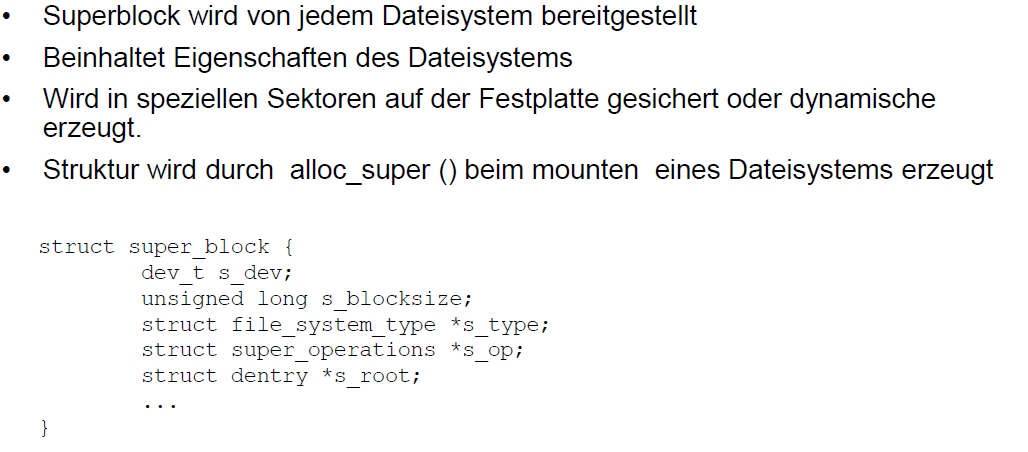
## Virtuelles Dateisystem



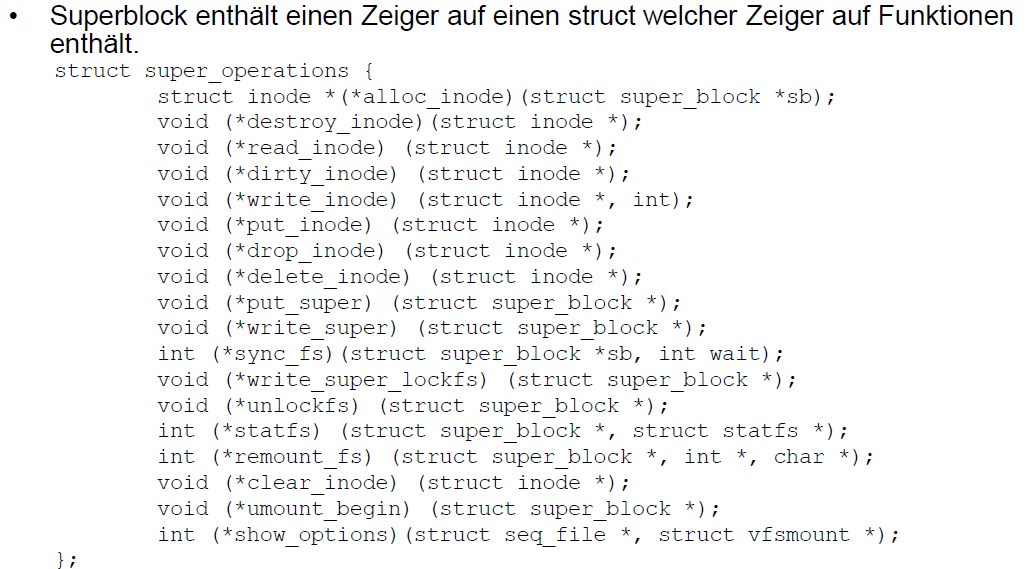
## VFS Objekte und Datenstrukturen



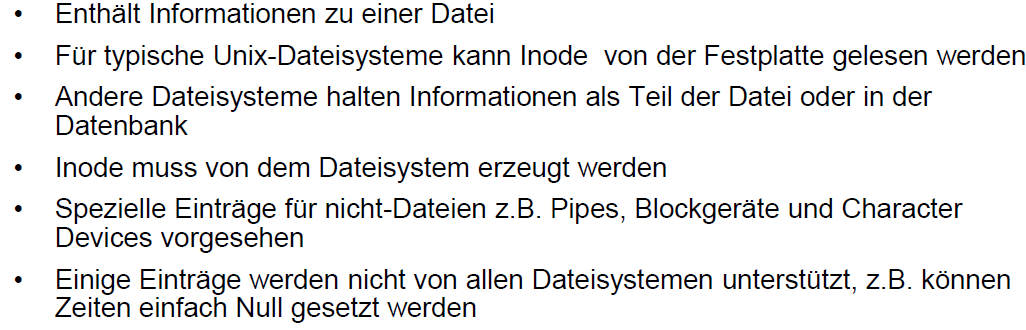
## Superblock



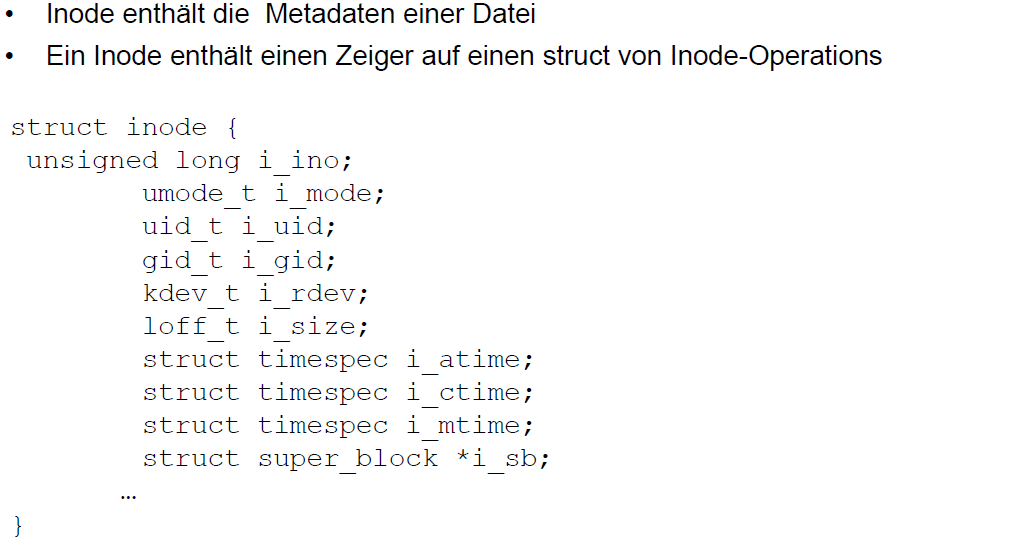
## Operationen auf Superblocks



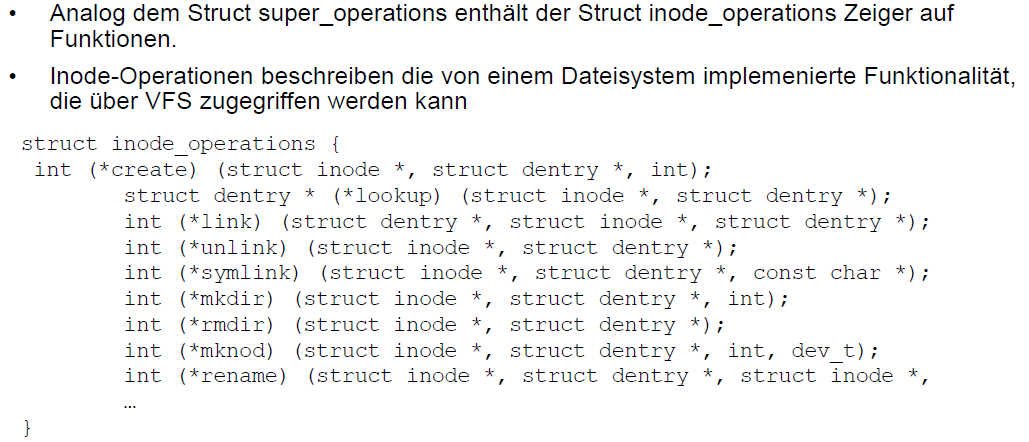
## Inodes



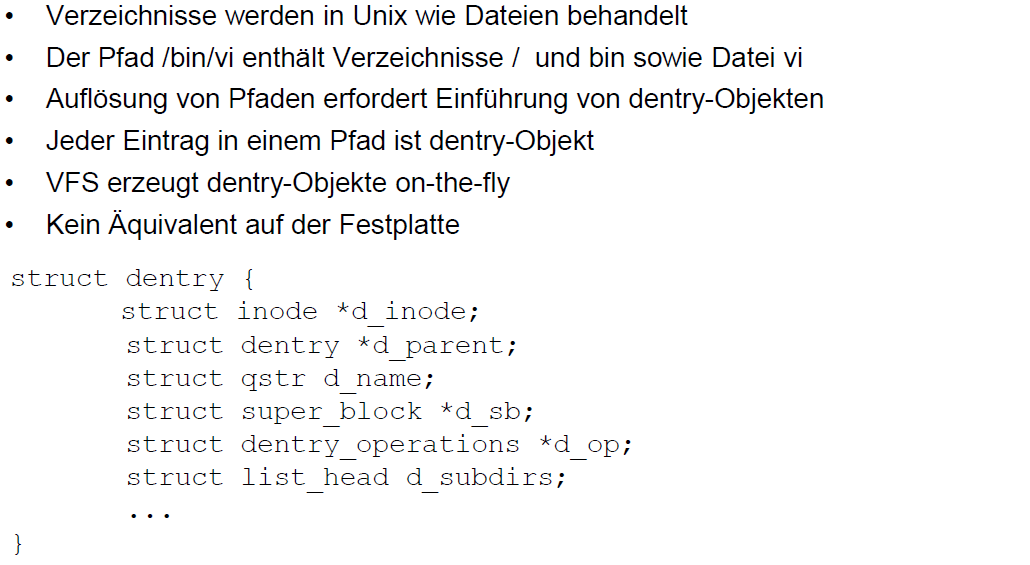
## Struktur struct inode



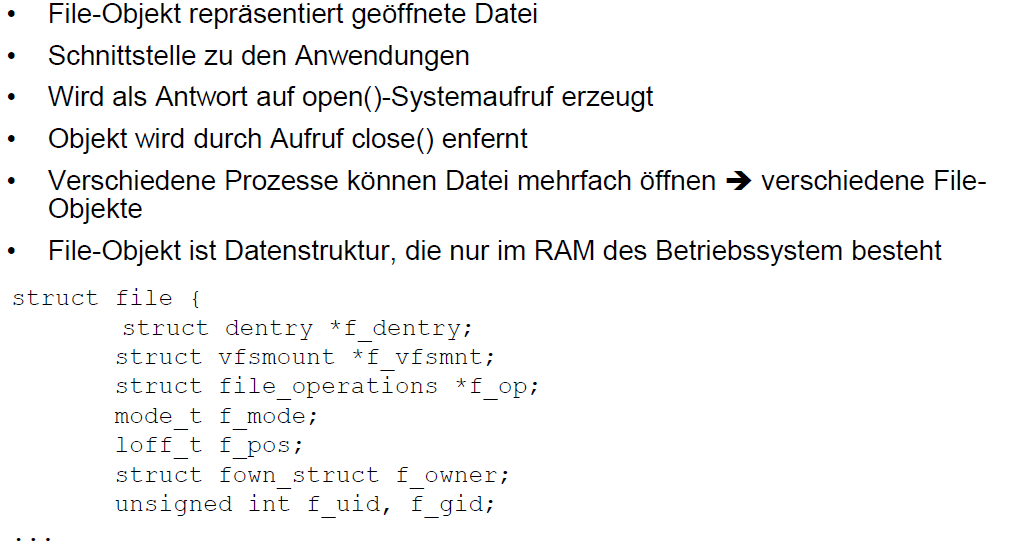
## Operationen auf inode



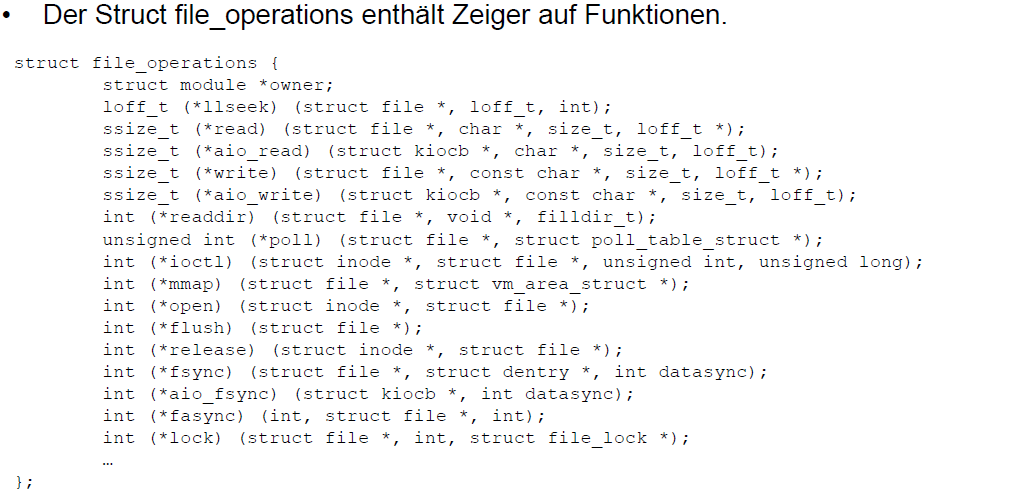
## Struktur struct dentry



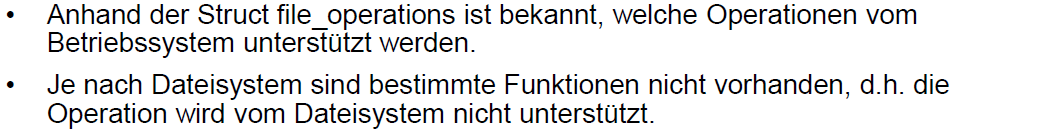
## Struktur struct file



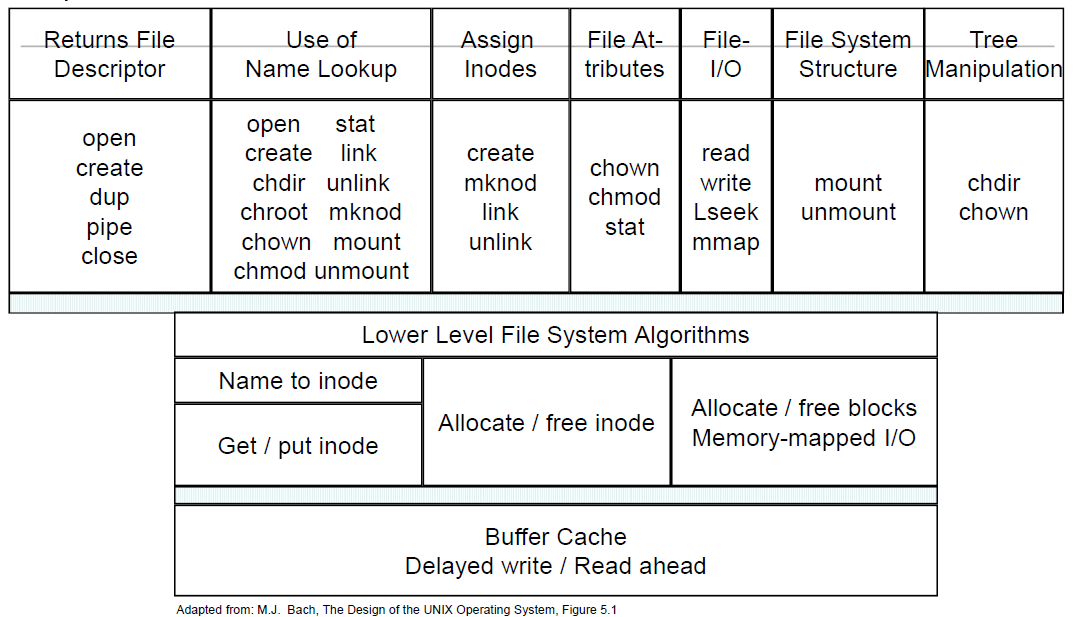
## Operationen auf file



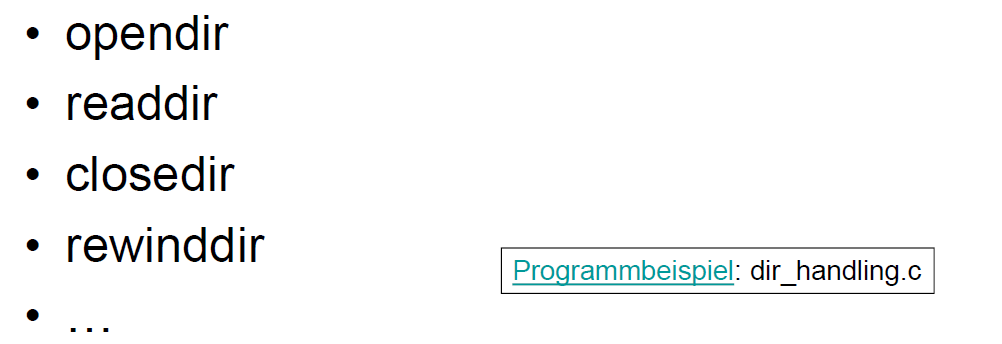
## Grundlegende Dateioperationen



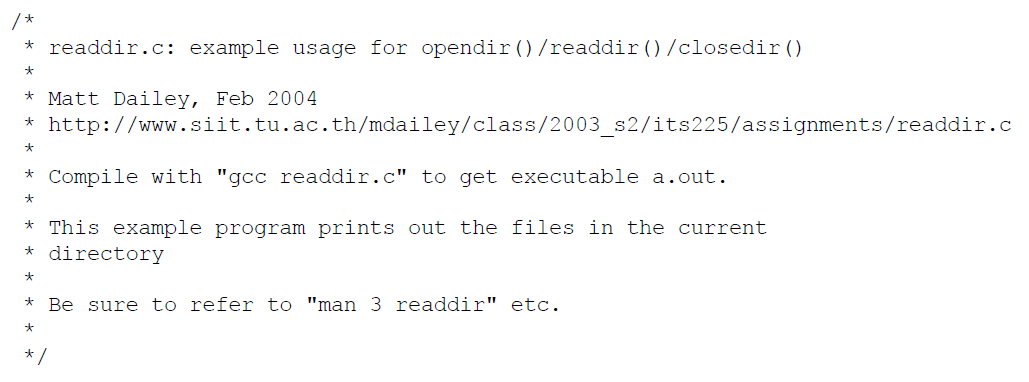
## System Calls im Linux Dateisystem

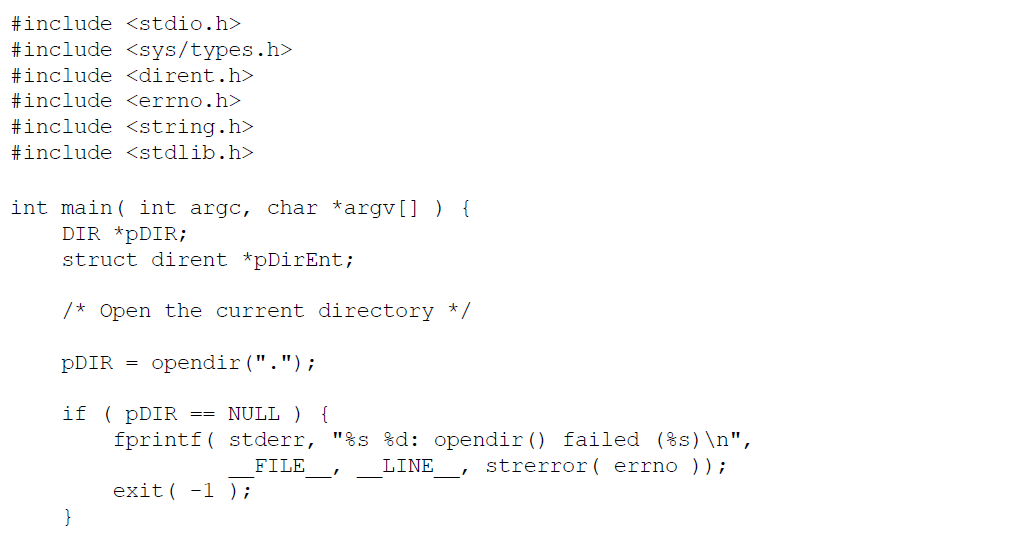


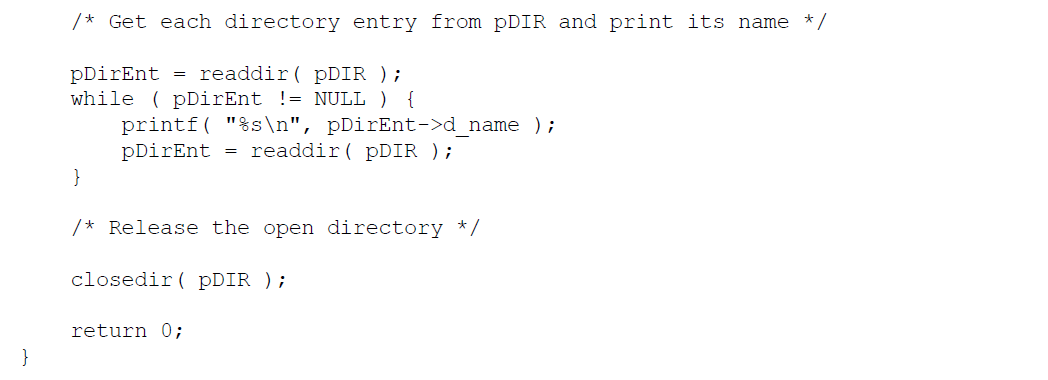
## Library Calls zur Verzeichnismanipulation



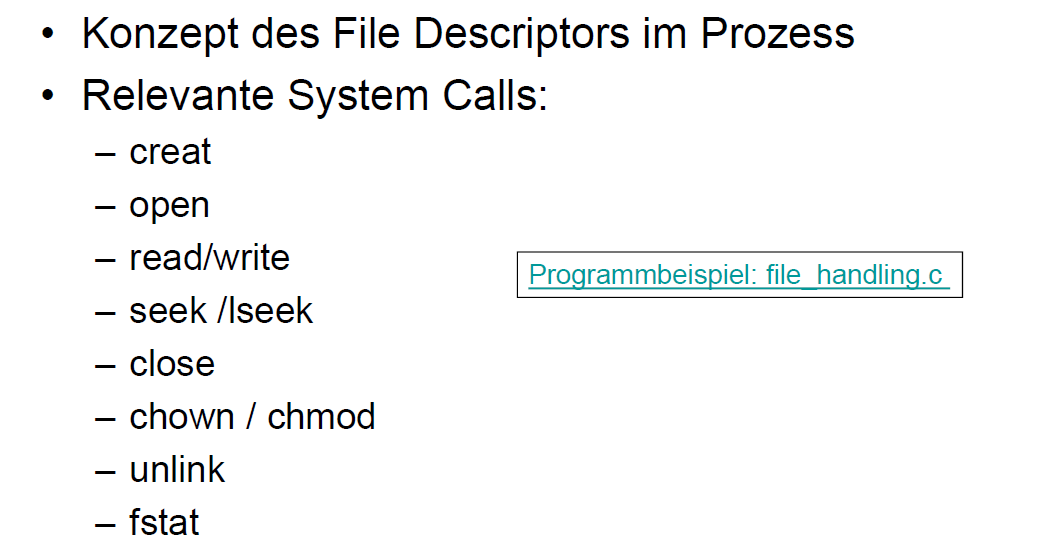
## Beispiel einer Verzeichnismanipulation



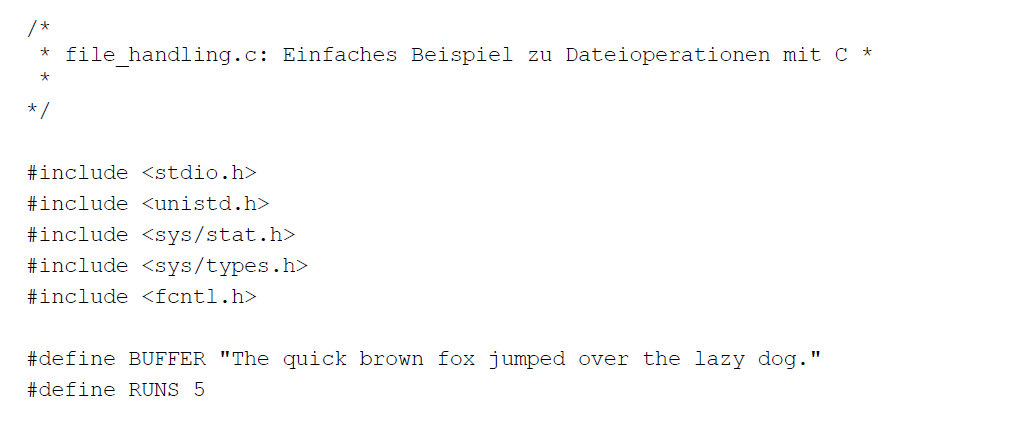


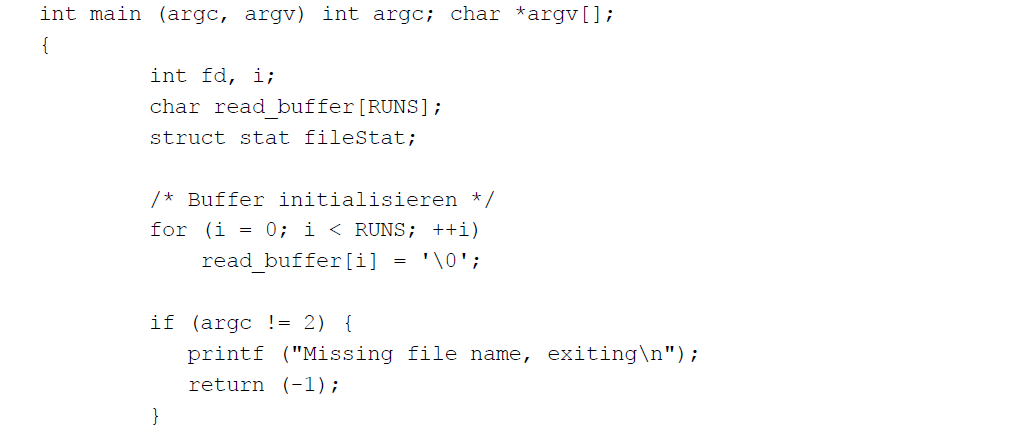


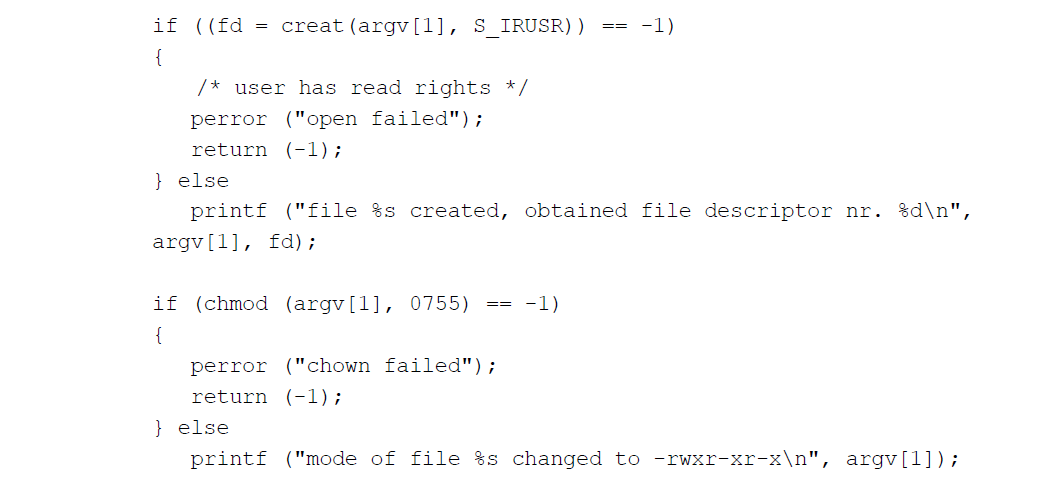
## System Calls zur Dateimanipulation

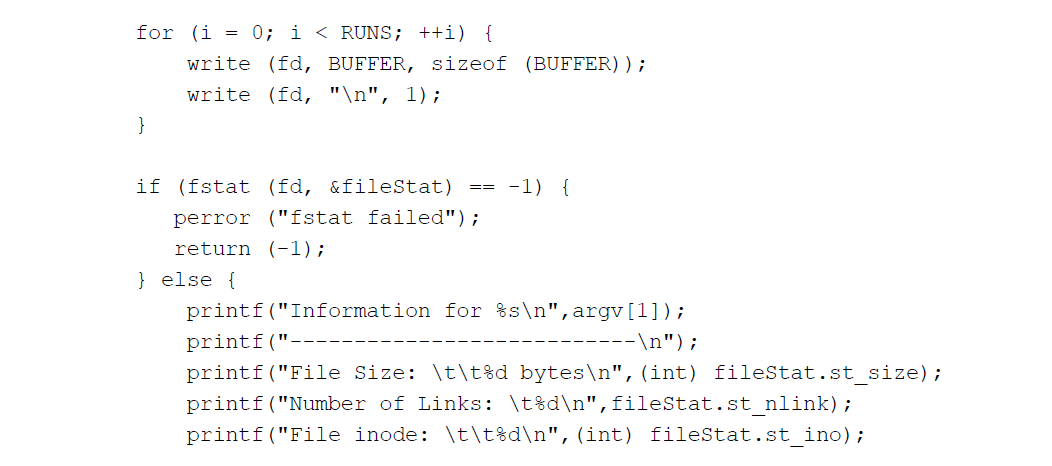


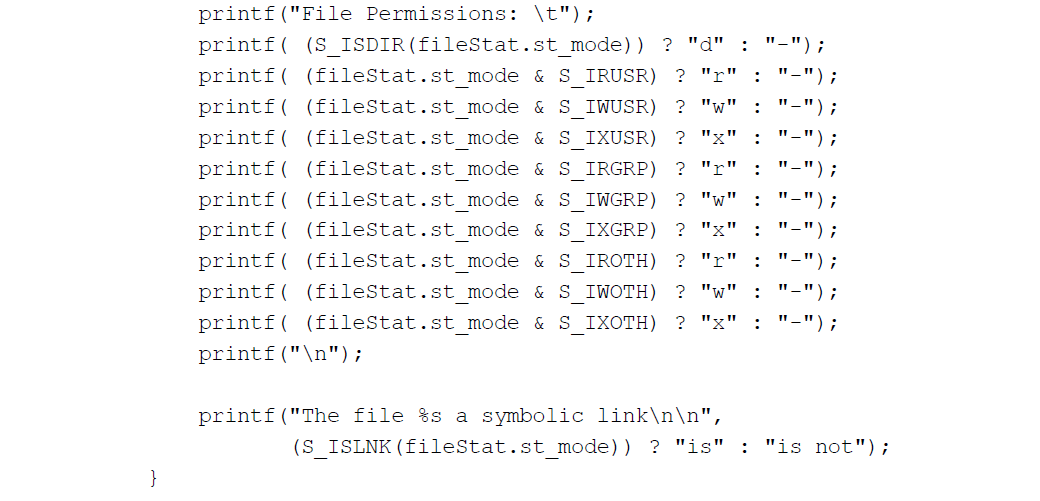
## Beispiel einer Dateimanipulation

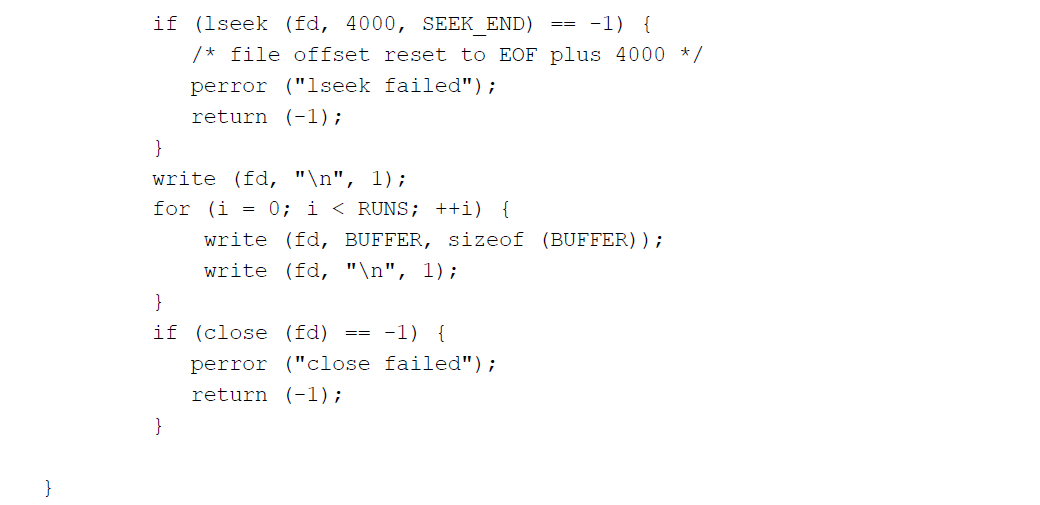








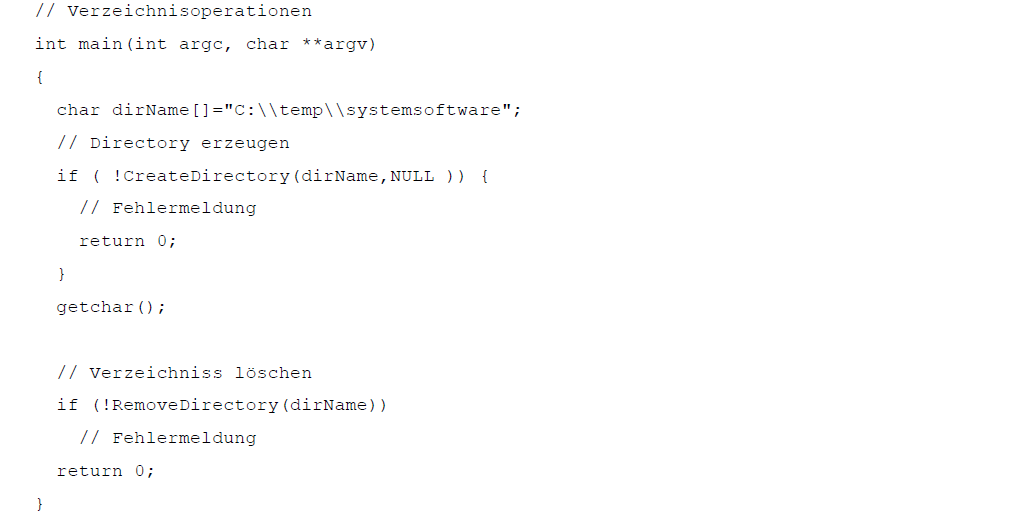


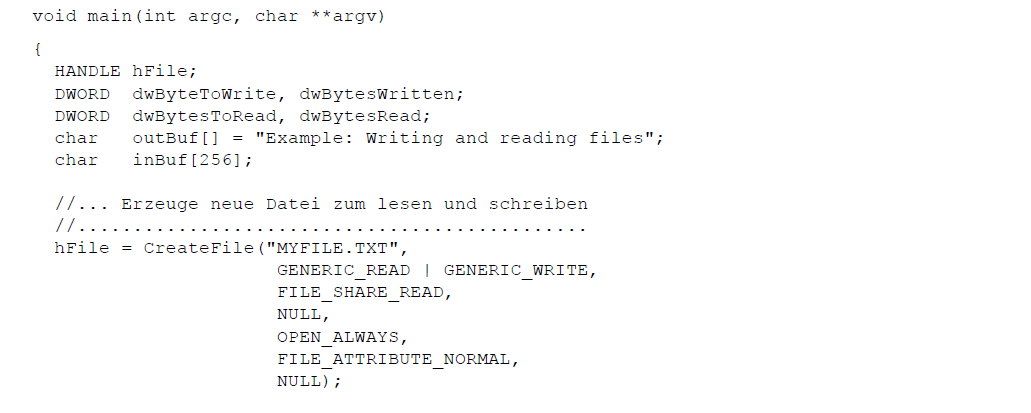


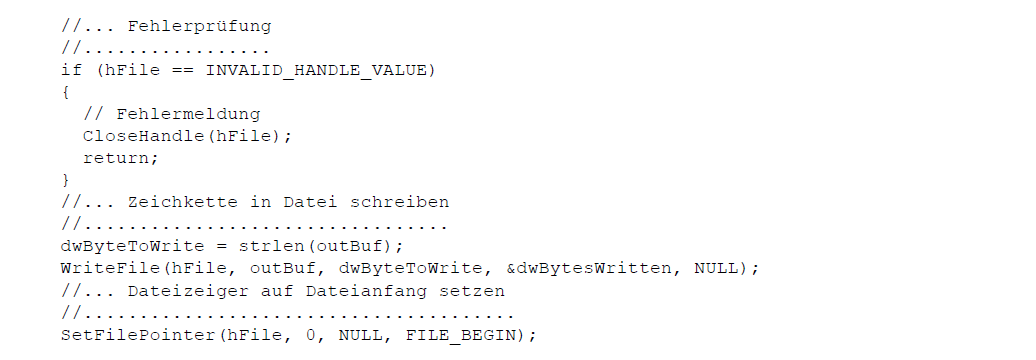
## System Calls zur Arbeit im Dateisystembaum

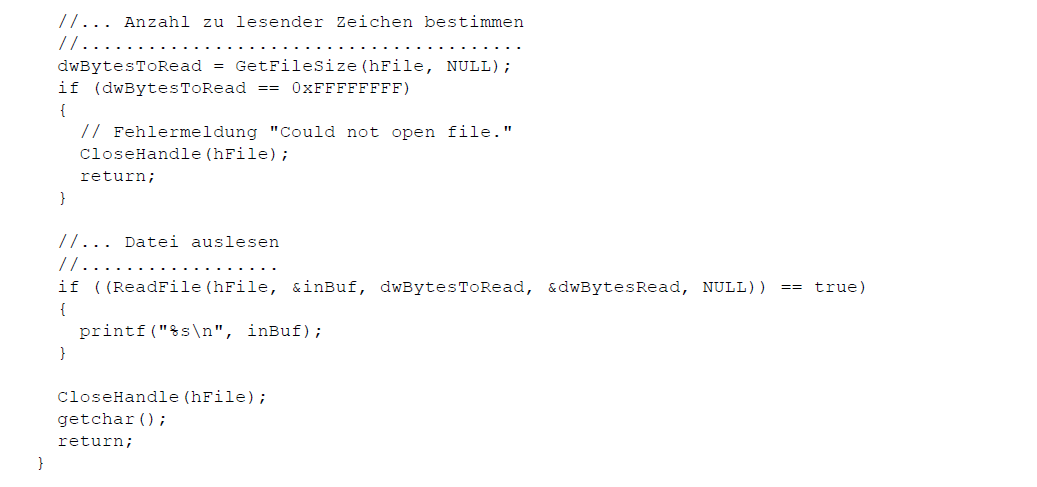


## Grundlegende API Funktionen in Windows









# Woche 6