

# Systemprogrammierung

## Teil 1: Einführung

# Systemsoftware versus Anwendungssoftware

---

Systemsoftware dient dem Betrieb von Rechnern

- Verwaltung der Hardware-Ressourcen und Steuerung der internen Abläufe  
*Prozessor- und Speicherverwaltung, Kommunikation mit angeschlossenen Geräten usw.*
- Bereitstellen einer komfortablen Ablaufumgebung für Anwendungssoftware  
*Verbergen von speziellen Hardware-Eigenschaften usw.*
- Beispiele:  
Betriebssysteme, Datenbanksysteme, Firmware, JVM (Java Virtual Machine), ...

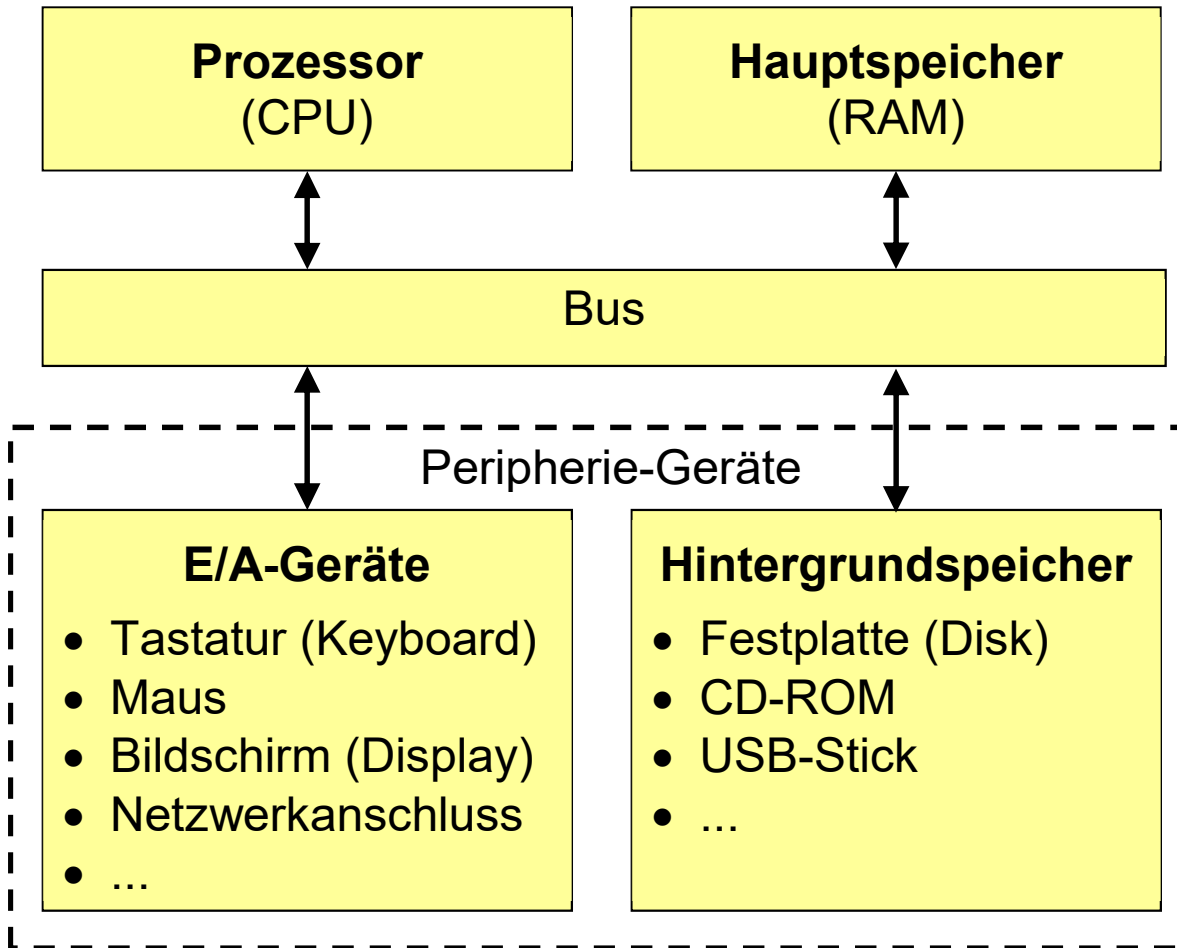
Anwendungssoftware stellt Funktionalität für Endbenutzer eines Rechners bereit

- Verwaltung und Verarbeitung von Anwenderdaten
- Bereitstellen komfortabler Bedienoberflächen
- Beispiele:  
Browser, Textverarbeitung, Computerspiele, ...

Die Grenze zwischen Systemsoftware und Anwendungssoftware ist fließend.

# Systemsoftware und Hardware

---



- Systemsoftware muss die Hardware-Ressourcen möglichst optimal nutzen  
*sie muss insbesondere die Begrenztheit der Ressourcen beachten*
- die Programmiersprache darf deshalb nicht zu stark von der Hardware abstrahieren  
*das gilt insbesondere für die Ressource Speicher*



Ursprünglich wurde Systemsoftware vollständig in Assemblersprachen erstellt

*eine Assemblersprache bietet lediglich lesbare Namen für Maschinenbefehle*

*die Programmierung ist dadurch mühsam und fehleranfällig*

*die Software ist an den Befehlssatz einer Prozessorfamilie gekoppelt*

Heute wird Systemsoftware überwiegend in den Hochsprachen C und C++ erstellt:

- C ist eine übersichtliche Sprache mit sehr guter Werkzeugunterstützung, die in den 1970er-Jahren als Programmiersprache von Unix entstanden ist:

1978: Kernighan & Ritchie - *"The C Programming Language"*

1989: **ANSI-C**, lange der am breitesten unterstützte Standard (= ISO-Standard C90)

1999: ISO-Standard C99 mit einigen (teils auch umstrittenen) Spracherweiterungen

2011: konsolidierter ISO-Standard **C11** mit *"bugfix release"* C18

- C++ ist eine sehr mächtige aber auch komplexe Sprache, die in den 1980er-Jahren als Erweiterung von C entstanden ist:

1985: Bjarne Stroustrup - *"The C++ Programming Language"*

1998: erster ISO-Standard C++98

2011: ISO-Standard **C++11** mit Konsolidierungen C++14 und C++17 (*"modern C++"*)

# Systemprogrammierung: Inhalt der Lehrveranstaltung

---

## Einführung in die Sprachen C und C++

- Sprachkonzepte: Datentypen, Anweisungen, Funktionen, Übersetzungseinheiten
- Standardbibliothek: Speicherverwaltung, Ein-/Ausgabe, Dateien, ...

## Werkzeuge

- Compiler: `gcc`, `g++`
- statische Codeanalyse: `cppcheck`
- Debugger: `ddd`, `valgrind`
- Automatisierung der Programmerstellung: `make`

## Programmorganisation

- ausführbare Dateien
- statische und dynamische Bibliotheken
- Archive
- Prozesse