

Grundlagen der Programmierung

Fabian Kubek

24. Dezember 2023

Themen

1 Aufbau eines Computers

2 Programmiersprachen

- Low Level Sprache
- High Level Sprache

Aufbau eines Computers

Die von Neumann Architektur legte die Grundlage für unsere heutigen Rechner

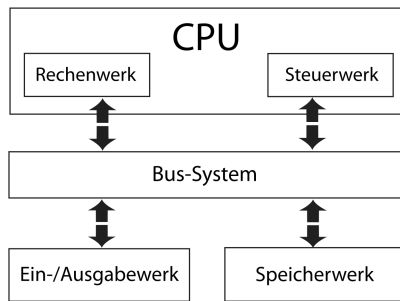


Abbildung: von Neumann Architektur

Aufbau eines Computers

- **Rechenwerk/ALU:** Hier werden die eigentlichen Rechenoperationen ausgeführt
- **Steuerwerk:** Verwaltet die Befehlsreihenfolge und ist für die Steuerung der Programme zuständig.
- **BUS-System:** Dient zur Kommunikation der einzelnen Komponenten
- **Speicherwerk (RAM):** Speichert Daten und Programme. Ist auch für das Rechenwerk zugänglich
- **Ein-/Ausgabewerk:** Leitet die Daten an andere Komponenten weiter (z.B. Tastatur, Bildschirm, etc.)

Befehlssatz

- Eine CPU hat eine vorgegebene Menge an Befehlen die diese ausführen kann.
- Je nach Art des Prozessors unterscheiden sich diese in ihrem Umfang.
- Diese können zum Beispiel sein ADD, STORE, MULT, etc.
- Der meist verbreitetste in Computern ist x86 bzw. x86_64, oder in den neuen Produkten von Apple der ARM Instruktionssatz

Arten von Programmiersprachen

Man unterscheidet grundsätzlich zwischen zwei Arten von Programmiersprachen:

- High-Level Programmiersprachen
- Low-Level Programmiersprachen

Maschinensprache/Assembler

- Maschinensprache oder Assembly sind beides Low-Level Programmiersprachen
- Diese erlauben es direkt mit den einzelnen Hardwarekomponenten zu interagieren
- Maschinensprache besteht eigentlich nur aus 0 und 1
- Assembler hingegen ist für Menschen lesbarer aber beinhaltet auch nur die einzelnen Instruktionen eines Prozessors

High Level Programmiersprachen

- Abgrenzung zu Low Level klar definiert: Ein Prozessor kann den Code nicht ohne Kompilierung ausführen
- Kompilieren bedeutet, dass der Code erst noch für den Prozessor "verständlich" gemacht werden muss
- Wann das Kompilieren stattfindet hängt von der Programmiersprache ab. Python z.B. zur Ausführungszeit, C++ muss erst kompiliert werden bevor der Code ausgeführt werden kann.

High Level Programmiersprachen

In High Level Sprachen unterscheidet man zwischen verschiedenen Paradigmen:

- **Strukturelle Programmierung:** beschränkt sich auf folgende Kontrollstrukturen: Sequenz, Auswahl, Wiederholung

High Level Programmiersprachen

In High Level Sprachen unterscheidet man zwischen verschiedenen Paradigmen:

- **Strukturelle Programmierung:** beschränkt sich auf folgende Kontrollstrukturen: Sequenz, Auswahl, Wiederholung
- **Imperative Programmierung:** In dieser Programmierung wird festgelegt was wann wie zu tun ist, dazu werden Kontrollstrukturen aus der strukturellen Programmierung verwendet

High Level Programmiersprachen

In High Level Sprachen unterscheidet man zwischen verschiedenen Paradigmen:

- **Strukturelle Programmierung:** beschränkt sich auf folgende Kontrollstrukturen: Sequenz, Auswahl, Wiederholung
- **Imperative Programmierung:** In dieser Programmierung wird festgelegt was wann wie zu tun ist, dazu werden Kontrollstrukturen aus der strukturellen Programmierung verwendet
- **Deklarative Programmierung:** Sind häufig Datenbanksprachen, man beschreibt was gemacht werden soll, aber nicht wie.

High Level Programmiersprachen

In High Level Sprachen unterscheidet man zwischen verschiedenen Paradigmen:

- **Strukturelle Programmierung:** beschränkt sich auf folgende Kontrollstrukturen: Sequenz, Auswahl, Wiederholung
- **Imperative Programmierung:** In dieser Programmierung wird festgelegt was wann wie zu tun ist, dazu werden Kontrollstrukturen aus der strukturellen Programmierung verwendet
- **Deklarative Programmierung:** Sind häufig Datenbanksprachen, man beschreibt was gemacht werden soll, aber nicht wie.
- **Objektorientierte Programmierung:** Lernen wir noch was das ist :)

High Level Programmiersprachen

In High Level Sprachen unterscheidet man zwischen verschiedenen Paradigmen:

- **Strukturelle Programmierung:** beschränkt sich auf folgende Kontrollstrukturen: Sequenz, Auswahl, Wiederholung
- **Imperative Programmierung:** In dieser Programmierung wird festgelegt was wann wie zu tun ist, dazu werden Kontrollstrukturen aus der strukturellen Programmierung verwendet
- **Deklarative Programmierung:** Sind häufig Datenbanksprachen, man beschreibt was gemacht werden soll, aber nicht wie.
- **Objektorientierte Programmierung:** Lernen wir noch was das ist :)
- **Funktionale Programmierung:** Lernen wir auch noch :)

Was davon kann Python?

- Wie viele andere Programmiersprachen vereint Python viele der eben genannten Paradigmen
- Python erlaubt es Strukturell, Imperativ, Objektorientiert und Funktional zu programmieren
- Python ist eine sogenannte Allzweck Programmiersprache