

Praktische Informatik I

WS 2005/2005

Prof. Dr. Wolfgang Effelsberg

Lehrstuhl für Praktische Informatik IV
Universität Mannheim

Inhaltsverzeichnis (1)

1. Einführung

1.1 Was ist Informatik?

1.3 Grundstruktur des Computers

2. Die Programmiersprache Java

2.1 Was ist Java?

2.2 Erste Beispiele

2.3 Lexikalische Konventionen

2.4 Typen und Werte

2.5 Konstanten und Variable

2.6 Typumwandlungen, Ausdrücke und Operatoren

Inhaltsverzeichnis (2)

2.7 Anweisungen

2.8 Felder

2.9 Klassen und Objekte

2.10 Subklassen, Superklassen und Vererbung

2.11 Pakete, Geltungsbereiche und Zugreifbarkeit

2.12 Interfaces

2.13 Eingebettete Klassendeklarationen

Inhaltsverzeichnis (3)

3. Entwurf von Algorithmen

3.1 Algorithmen, Programmiersprachen und Programme

3.2 Systematischer Entwurf von Algorithmen

3.3 Schrittweise Verfeinerung

3.4 Ablaufsteuerung (Kontrollstrukturen)

3.5 Modularität

3.6 Rekursion

3.7 Daten und Datenstrukturen

3.8 Bäume

3.9 Endliche Automaten

Inhaltsverzeichnis (4)

4. Exkurs: Einführung in die Logik

5. Theorie der Algorithmen

5.1 Berechenbarkeit

5.2 Komplexität

5.3 Korrektheit und Verifikation

6. Alternative Programmierkonzepte

6.1 Ereignisgesteuerte Programmierung

6.2 Programmierung mit Logik

6.3 Funktionale Programmierung

Literatur

1. Basis der Vorlesung

Martin Schader, Lars Schmidt-Thieme: Java - Eine Einführung, 4. Auflage, Springer-Verlag, Berlin/Heidelberg/New York, 2003

Heinz-Peter Gumm, Manfred Sommer: Einführung in die Informatik, 6. Auflage, Oldenbourg-Verlag, 2004

2. Weitere Empfehlungen

Harvey M. Deitel, Paul J. Deitel: Java: How to program, 6th edition, Prentice Hall, 2004 (1600 Seiten!)

Uwe Schöning: Logik für Informatiker, 5. Auflage, Spektrum Akademischer Verlag, 2000

Danksagung

Die Kollegen Hesser, Schader, Schmeck und Ottmann haben mir zur Vorbereitung dieser Vorlesung Folien und andere Unterlagen überlassen. Für ihre Unterstützung möchte ich mich herzlich bedanken.

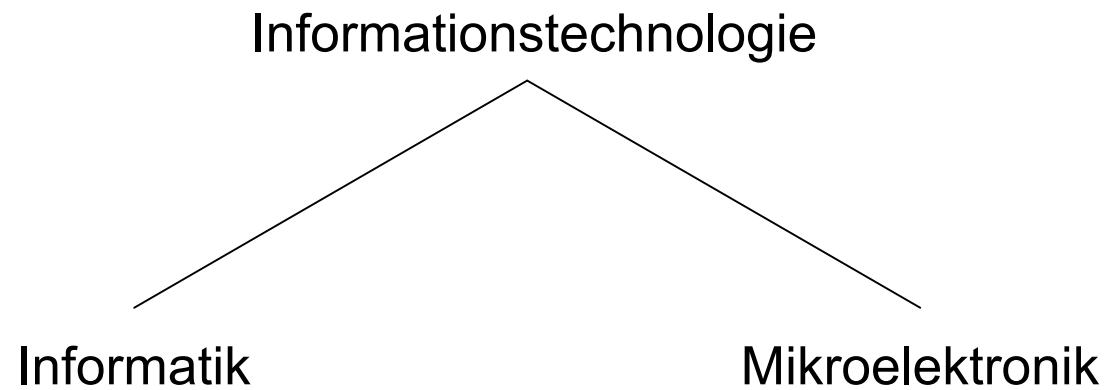
1.1 Was ist Informatik?

Informatik ist die Wissenschaft von der systematischen Verarbeitung von Informationen, insbesondere der automatisierten Verarbeitung mit Hilfe von Digitalrechnern.

Sie befasst sich mit

- der Struktur, der Wirkungsweise, den Fähigkeiten und den Konstruktionsprinzipien von Informationsverarbeitungssystemen
- Strukturen, Eigenschaften und Beschreibungsmöglichkeiten von Informationen und von Informationsverarbeitungsprozessen (Datenstrukturen und Algorithmen)
- Möglichkeiten der Strukturierung, Formalisierung und Mathematisierung von Anwendungen sowie der Modellbildung und Simulation.

Einordnung der Informatik



Teilgebiete der Informatik

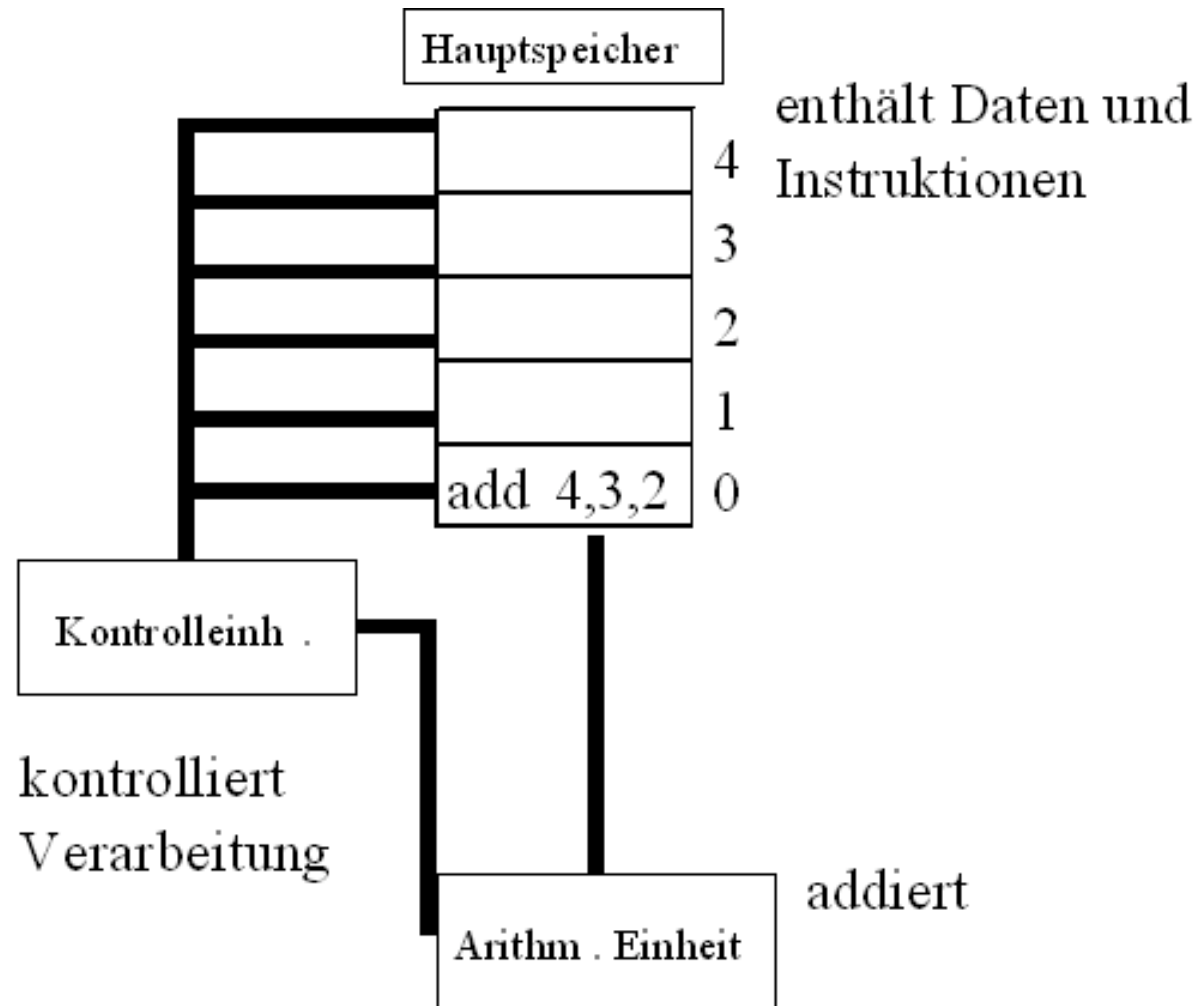
Theoretische Informatik	Technische Informatik (Hardware)	Praktische Informatik	Angewandte Informatik
Automatentheorie	Rechner-architektur	Programmiersprachen	Wirtschaftsinformatik
Formale Sprachen	Schaltkreis-entwurf	Datenstrukturen	Ingenieursinformatik
Komplexitäts- theorie	Prozessoren	Datenbanken	Computerlinguistik
usw.	Speicher	Betriebssysteme	naturwiss. Anwendungen
	periphere Geräte	Rechnernetze	geisteswiss. Anwendungen
	Kommunika- tionsgeräte	Künstliche Intelligenz	juristische Anwendungen
	usw.	usw.	usw.

Verwandte Gebiete

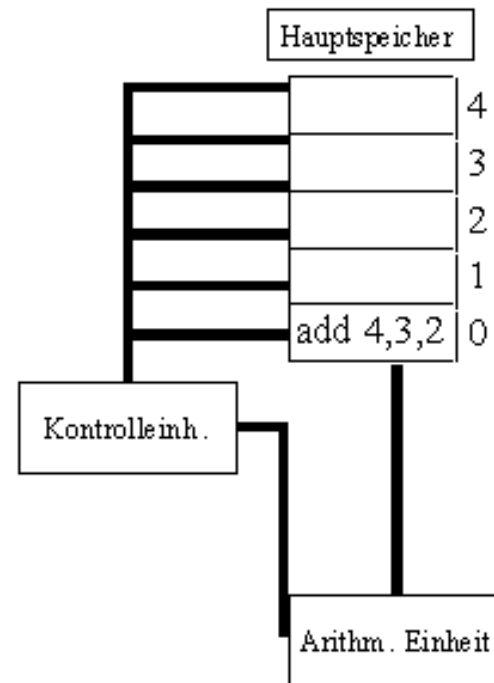
1.2 Grundstruktur des Computers

Aufbau eines Computersystems

Funktionsweise (1)



Funktionsweise (2)



Instruktion auf Adresse 0:

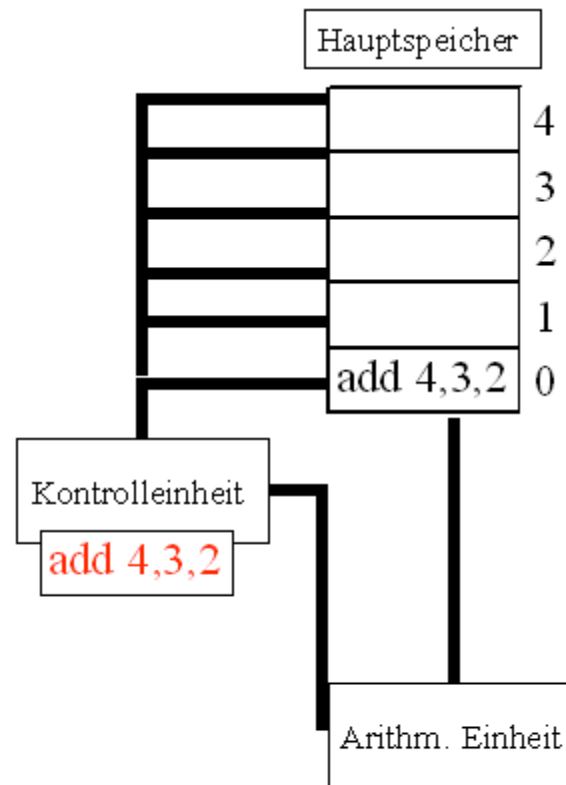
add Werte von Adressen **4** und **3** und schreibe Resultat auf
Adresse **2**

lese nächste Instruktion von Adresse **1**

Abgekürzt:

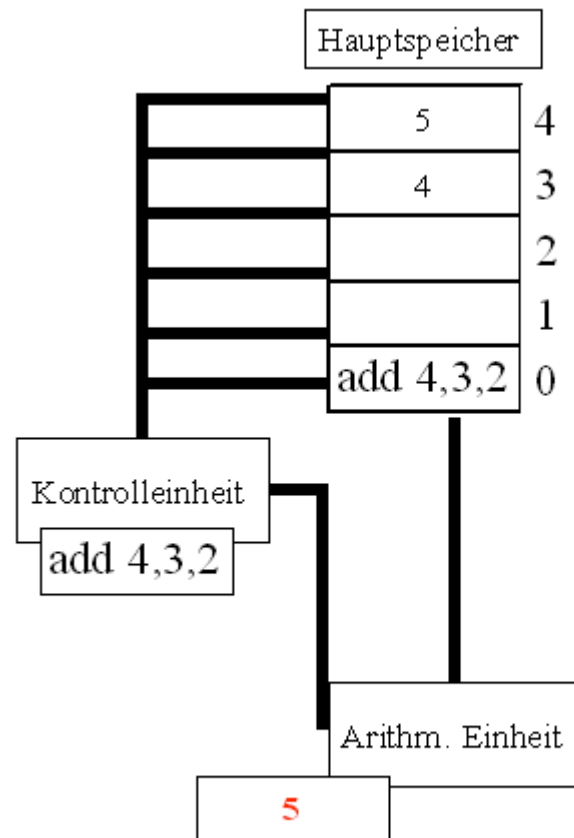
add 4,3,2

Funktionsweise (3)



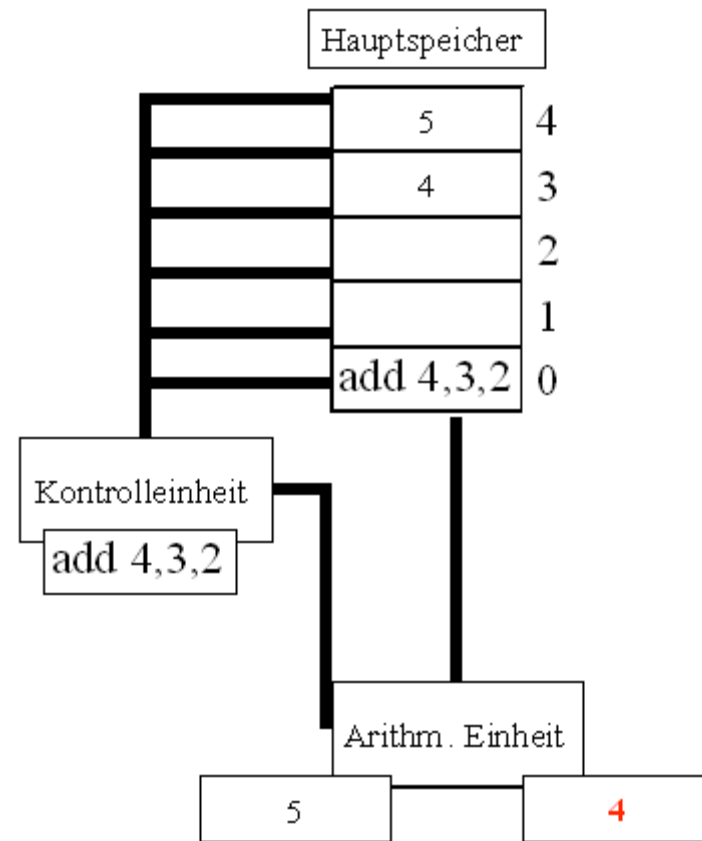
lese Instruktion von Adresse 0

Funktionsweise (4)



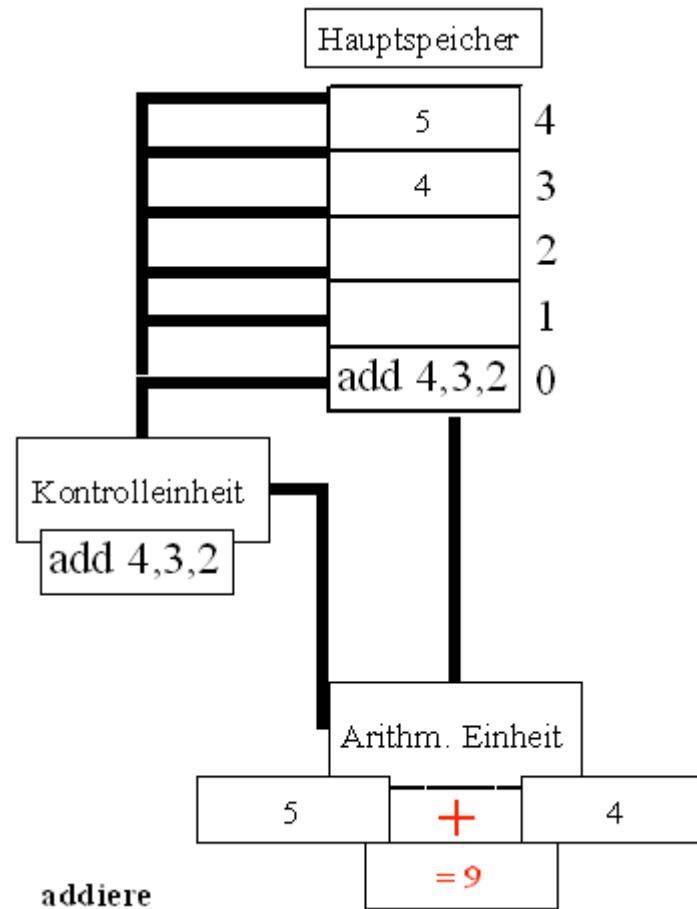
schreibe Wert von Adresse 4 in arithm. Einheit

Funktionsweise (5)

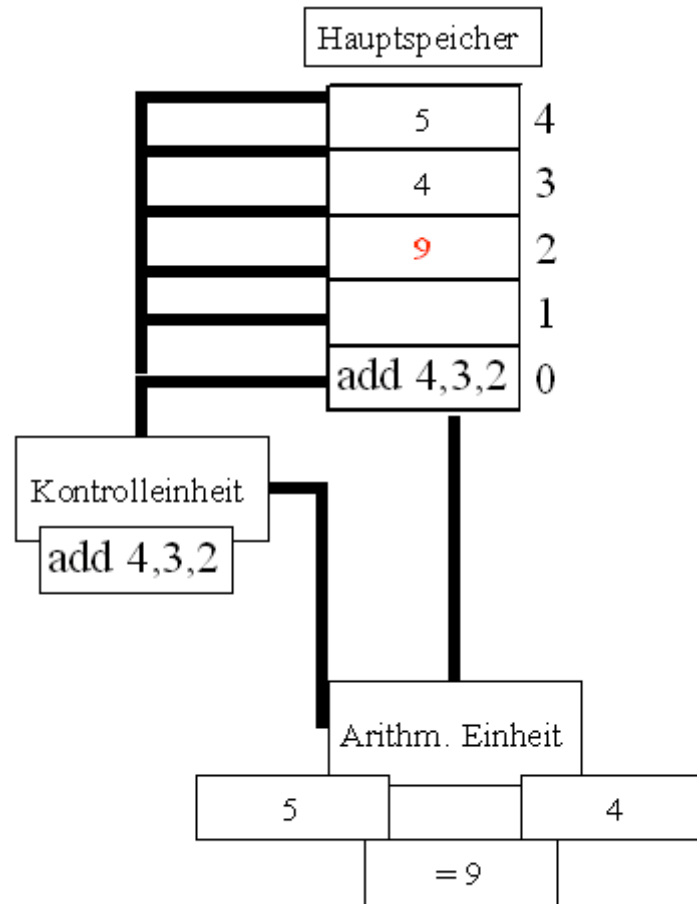


schreibe Wert von Adresse 3 auf arithm. Einheit

Funktionsweise (6)

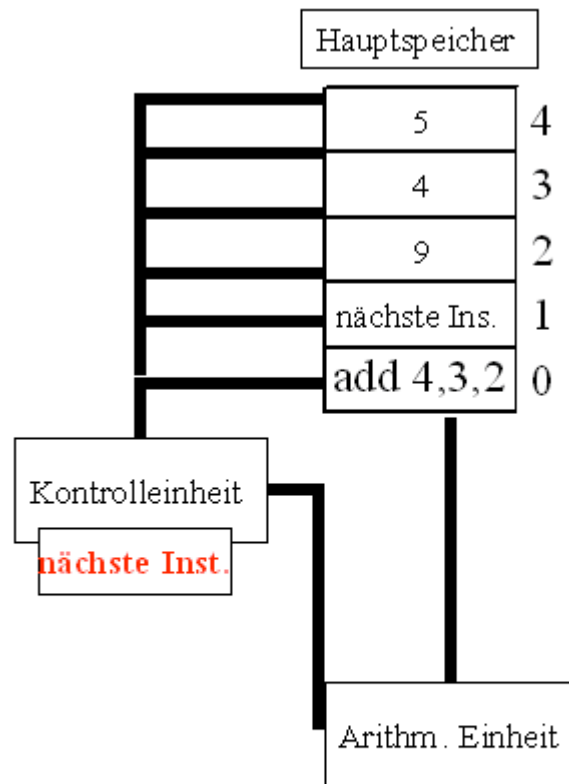


Funktionsweise (7)



schreibe Resultat auf Adresse 2

Funktionsweise (8)



lese nächste Instruktion von Adresse 0+1

Software

Software ist eine Abfolge von Instruktionen, die von einem Prozessor verarbeitet werden.

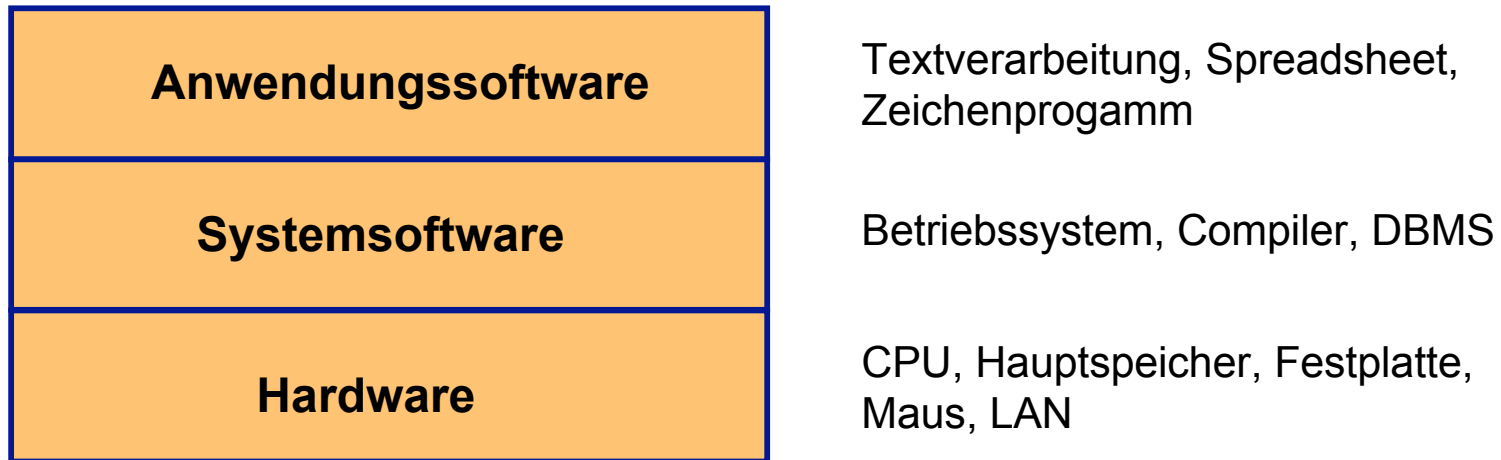
Beispiele für Instruktionen

`y = 4 + 3;` höhere Programmiersprache

`add 4,3,2` Assembler

`01110101010` Maschinensprache

Hardware- und Software-Schnittstellen



Software-Beispiele

Anwendungssoftware

- Editor
- Web-Browser
- Java-Programm (eigenes Programm)

Systemsoftware

- Betriebssystem
- Compiler
- Datenbank-Managementsystem
- Netzwerksoftware (Treiber)

Hardware-Beispiele

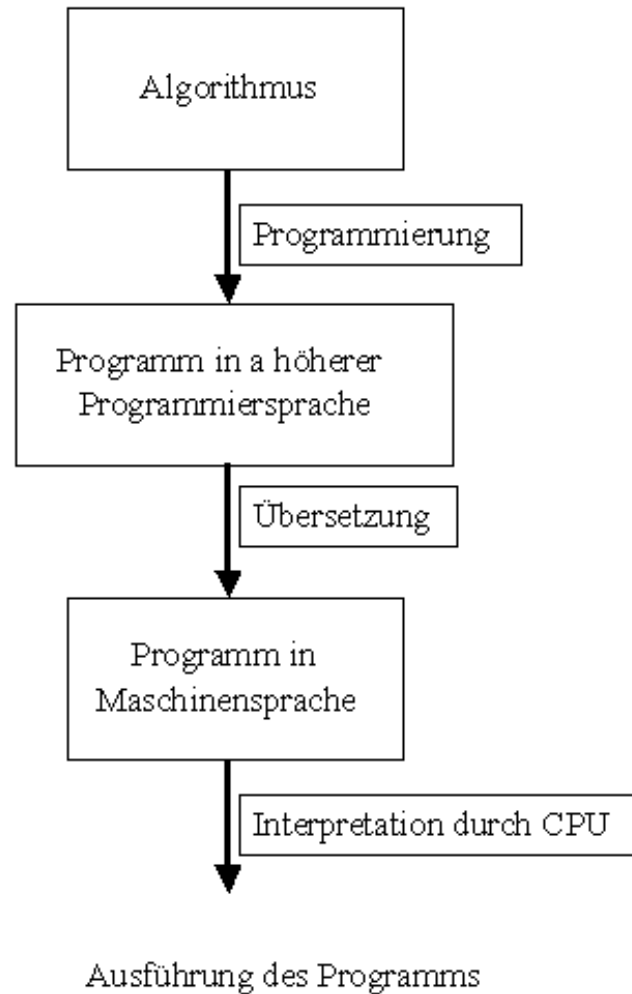
Hardware

- CPU
- Hauptspeicher
- Festplatte
- Ethernet-Karte
- Tastatur
- Monitor

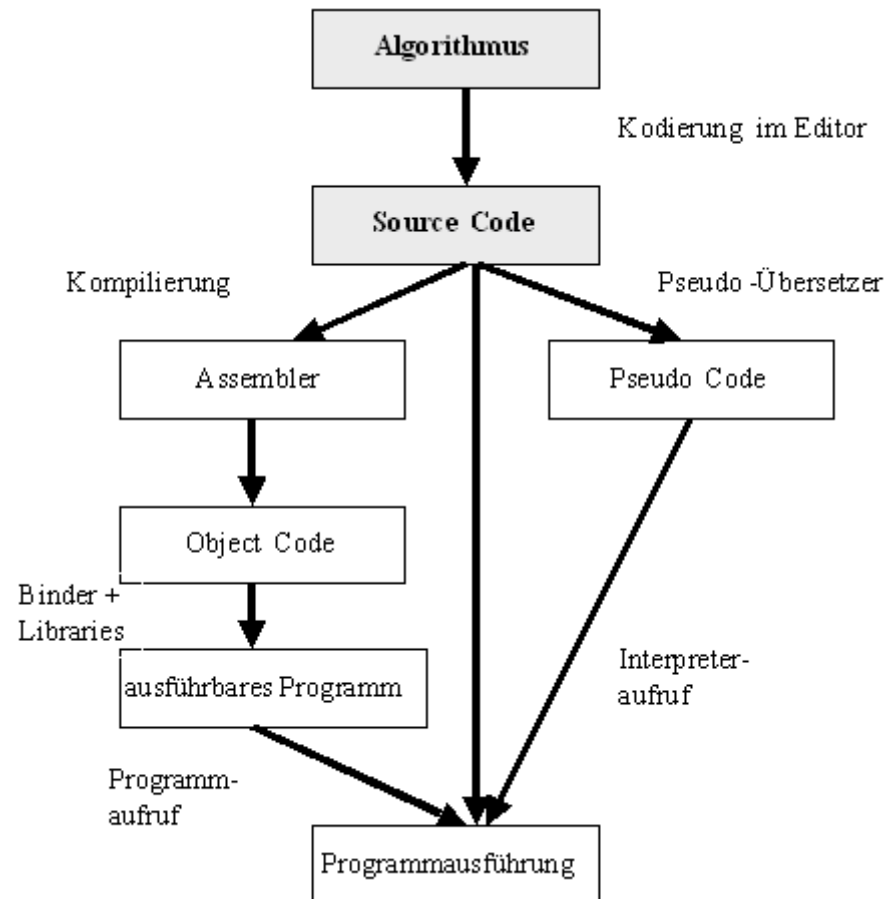
Vom Anwendungsproblem zum Programm

- Formuliere das Problem
- Skizziere die Lösung (z. B. Algorithmus in Pseudo-Code)
- Schreibe den Algorithmus in einer Programmiersprache auf
- Kompiliere, binde und starte das Programm

Vom Algorithmus zum Maschinenprogramm



Kompilierung vs. Interpretation



Typische Java-Umgebung

