

Programmieren

Heute: Was heißt es eigentlich zu programmieren

Programmieren: Arbeitsschritte für einen Vorgang (z.B. Primzahlen ausrechnen, Schach spielen, Gesichter erkennen) so erklären, dass eine Maschine sie versteht

Historisches

Rechenmaschinen sind sehr alt: Antikythera (150-100 BC)
berechnet astronomische Bewegungen

Noch älter: Stonehenge etc.

Rechenmaschinen haben nur eine fixe Zahl von Funktionen.

Wann gab es Maschinen, denen man etwas erklären kann?

- Hero von Alexandria (10-70 AD):
Programmierbarer Roboter, für's Theater
"Programm" besteht aus Schnur, die um die Achsen gewickelt wird.
- 1725 Bouchon baut automatisierten Webstuhl
"Programm" ist als Lochband kodiert

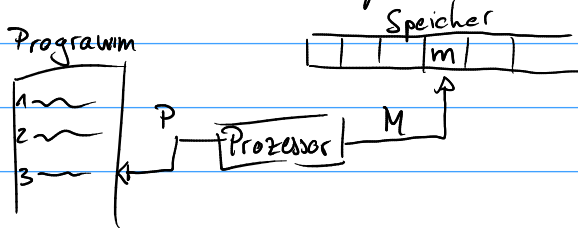
Bis hier waren "Programme" feste Abläufe. Die Ausführung läuft immer gleich.

- 1833: Babbage: Analytical Engine
Der erste echte Computer (leider nie gebaut)
Lochkarten für Programm und Daten
Alles rein mechanisch
- 1941: Konrad Zuse Z3, der erste elektronische Computer

Programmiersprachen

Computer sind im Prinzip einfache Maschinen. Der einfachstmögliche Computer hat zwei Zähler:

- Wo bin ich im Programm
- Wo bin ich im Speicher



Also genau wie die Turing-Maschine!

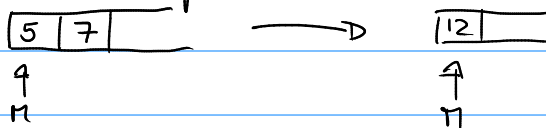
Und versteht 5 Anweisungen

- | | | |
|------|--|------------------------|
| 1. | Erhöhe M oder m um 1 | INC(M), DEC(M) |
| 2. | Vermindere M o. m um 1 | INC(m), DEC(m) |
| 3.4. | Gib m aus / Lies einen Wert in m ein | READ, WRITE |
| 5. | Solange $m \neq 0$: (wiederhole) | REPEAT |

Das ist schon eine gewaltige Abstraktion von dem, was physikalisch passiert!

Man kann diese Befehle jetzt benutzen um kompliziertere Anweisungen zu bauen.

Zum Beispiel: Addition



INC(M)

REPEAT

DEC(M)

INC(m)

INC(M)

DEC(M)

Übung:

$|x|$ → $|0|x|$ MOVE

$|x|$ → $|x|x|$ COPY

$|a|b|$ → $|a|b|a+b|$ ADD

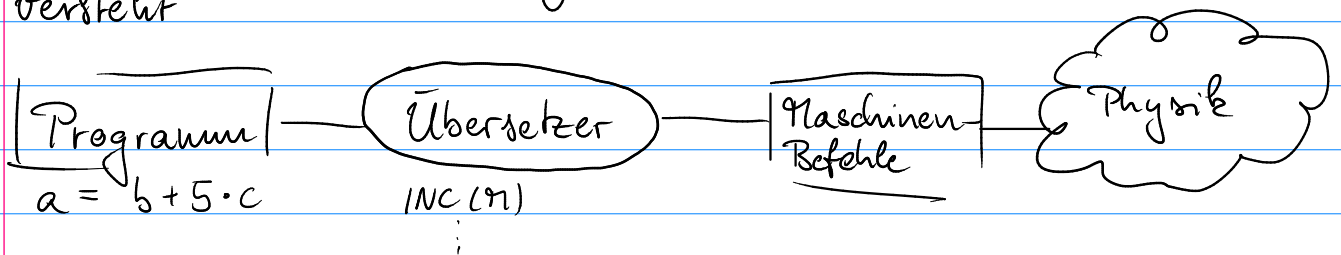
Für Wagemutige:

$|a|b|$ → Ausgabe 1 wenn $a > b$, sonst Ausgabe 0

Niemand kann sinnvolle Programme schreiben, wenn schon Addition so kompliziert ist.

Man könnte die physikalische Maschine klüger machen, Addition, Listenverarbeitung etc, das stößt aber schnell an Grenzen.

Stattdessen schreibt man Programme, die komplizierte Anweisungen in Folgen von Anweisungen übersetzen, die der Computer versteht



Das ist eine weitere Abstraktionsebene:

Programmiersprache
Maschinenbefehle
Elektrotechnik
Quantenmechanik

Jede Abstraktionsebene erlaubt Lösung viel komplizierterer Probleme

Sorting

Genau wie niemand Addition programmieren möchte, möchte niemand Sortieren programmieren.

Bibliotheken von Lösungen zu verschiedenen Problemen erlauben Wiederverwendung von Code.

Bibliothek
Programmiersprache
Maschinenbefehle
Elektrotechnik
Quantenmechanik

Modernes Programmieren besteht größtenteils aus dem Verbinden verschiedener Bibliotheken

Catpictures