

Programmausführung

Drei Wege bis zur
Programmausführung

Compiler

Die Programme werden direkt in die Maschinsprache des jeweiligen Rechners übersetzt. Das übersetzte Programm wird dann direkt von der Hardware interpretiert.

Interpreter

Die Programme werden nicht übersetzt, sondern direkt von einem Interpreter-Programm ausgeführt.

Compiler + Interpreter (virtuelle Maschine)

Die Programme werden in eine Zwischensprache übersetzt und von einer so genannten virtuellen Maschine (vereinfachter Interpreter) ausgeführt.

Programmausführung

Arbeitsweise eines Compiler

Programm in einer
höheren
Programmiersprache



Programm in
Maschinsprache

```
0000011101101011  
0101010111110101  
0000101010111111  
1111110101010110  
0010101010111110  
....
```

Das Programm kann
direkt von der Hardware (CPU)
ausgeführt werden

Beispiele: C, C++,

Hardware meint hier insbesondere, dass Maschinsprache-Programme nur von der CPU ausgeführt werden können für die sie auch compiliert wurden. Einschliesslich der gesamten Hardware-Nahen Komponenten, wie z.B. Speicher, Peripherie, Ein- und Ausgabegeräte.

Programmausführung

Interpreter

Programm in einer höheren Programmiersprache

```
read (a);  
read (b);  
if (a<b) then  
    a = a*a;  
else  
    a = a+b;  
print a;
```



Das Programm wird hier interpretiert

Der Interpreter enthält somit alle Hardware-nahen Komponenten, die sonst bei einem Maschinensprache-Programm durch den Compiler mit integriert werden müssen.

Der Interpreter wird direkt von der Hardware ausgeführt

Beispiel: Skriptsprachen: JavaScript, JScript, Python, Tcl/Tk, VBA usw.

Beachte: JavaScript ≠ Java!
JavaScript wird direkt im Browser interpretiert (ausgeführt)

Programmausführung

Compiler + Interpreter

Programm in einer höheren Programmiersprache

```
read (a);  
read (b);  
if (a<b) then  
  a = a*a;  
else  
  a = a+b;  
print a;
```

Übersetzer

Programm in einer Zwischensprache

```
LOAD #1 C  
LOAD #2 B  
MULT #1 #2 #3  
LOAD #4 A  
ADD #3 #4 #1  
STORE #1 C  
LOAD #4 A  
ADD #3 #4 #1
```

Der Zwischencode wird hier interpretiert

Interpreter Virtuelle Maschine

Der Interpreter wird direkt von der Hardware ausgeführt.

Beispiele: Java

Beispiel: Die JVM (Java Virtuell Machine) ist Bestandteil einer JRE (Java Runtime Environment) und führt Java-Zwischencode direkt auf jeder Hardware aus, für die eine JVM existiert.
„Write once - run everywhere“

Wir setzen nur den Interpreter im Unterricht ein. Das insbesondere didaktische Gründe. Die Handhabung ist wesentlich einfacher. Ergebnisse lassen sich direkt durch den Funktionsaufruf evaluieren. Das Erstellen von Skripten ist unkompliziert.

Programmausführung in Haskell

Compiler

Haskell (GHC)

Glasgow Haskell Compiler

Haskell-Skripte haben die Dateiendung .hs und können in jedem beliebigen Texteditor bearbeitet werden.

```
helloFU = "Welcome to FU!"
```

Beispiel für einen Aufruf des Haskell-Compilers am Prompt.

```
$ ghc --make -o hello hello.hs  
$ ./hello  
Welcome to FU!
```

Interpreter-**Arten in Haskell**

GHCI

z.B. direkt in WIN/MAC/LINUX über Kommandozeile/Terminal oder über Visual Studio Code. Manchmal auch GHCi bezeichnet.

Haskell (WinHugs)

WinHugs stellt eine komplette IDE (ohne Editor) für Windows dar. Im Grunde ein Terminalfenster in einem Windows-Fenster ;-)

HUGS

Haskell User's Gofer System



Beispiel für einen Aufruf der definierten Funktion am WinHugs-Prompt.