
# Kompilieren & Interpretieren

# Arten von Code
## Quellcode

- lesbarer Text eines Computerprogrammes oder einer Webseite
- wird in einer Programmiersprache verfasst
- kann aus mehreren Dateien bestehen
- Syntax und Befehle hängt von der Sprache ab
- vom Programmierer geschrieben
- über Editor/Entwicklungsumgebung erstellt

# Arten von Code
## Maschinencode

Quelle: http://evlearners.com/binary-arithmetic-tutorial/ [Stand: 08.09.2022]

- die „Programmiersprache eines Computers“
- Bits und Bytes
  - Beispiel: 1011 1100 1101 0101 0000
- Stark abhängig von der Hardware
- für Menschen unverständlich
- wird vom Computer ausgeführt

# Arten von Code
## Bytecode

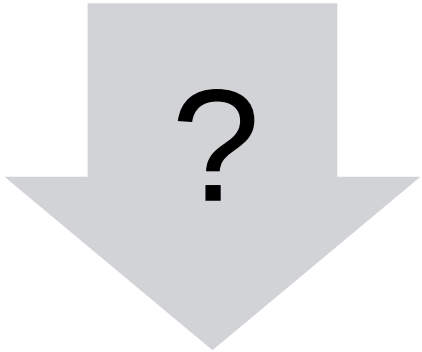
Quelle: http://evlearners.com/binary-arithmetic-tutorial/ [Stand: 08.09.2022]

- Zwischencode
- Bits und Bytes
  - Beispiel: 1101 0001 1111 1110 0001
- Sammlung von Befehlen für eine virtuelle Maschine
- unabhängig von Hardware
- für Menschen unverständlich

# Vom Quellcode zum Maschinencode

Quellcode

Vom Menschen lesbar




Quellcode wird übersetzt.

1. Compiler
2. Interpreter

Maschinencode

Vom Computer lesbar

# Compiler

- Übersetzt kompletten Quellcode in Maschinencode
  - Genannt: Kompilieren
- Erst dann kann Code ausgeführt werden !

```
IF X >= 5 THEN
MSGBOX «Geben Sie eine Zahl <5 ein»
ELSE
MSGBOX «Perfekt»
END IF
```

# Compiler

- Übersetzt kompletten Quellcode in Maschinencode
  - Genannt: Kompilieren
- Erst dann kann Code ausgeführt werden !

01101010

MSGBOX «Geben Sie eine Zahl <5 ein»

ELSE

MSGBOX «Perfekt»

END IF

# Compiler

- Übersetzt kompletten Quellcode in Maschinencode
  - Genannt: Kompilieren
- Erst dann kann Code ausgeführt werden !

01101010

10001111

ELSE

MSGBOX «Perfekt»

END IF

# Compiler

- Übersetzt kompletten Quellcode in Maschinencode
  - Genannt: Kompilieren
- Erst dann kann Code ausgeführt werden !

01101010

10001111

11000010

MSGBOX «Perfekt»

END IF

# Compiler

- Übersetzt kompletten Quellcode in Maschinencode
  - Genannt: Kompilieren
- Erst dann kann Code ausgeführt werden !

01101010

10001111

11000010

01000001

END IF

# Compiler

- Übersetzt kompletten Quellcode in Maschinencode
  - Genannt: Kompilieren
- Erst dann kann Code ausgeführt werden !

01101010

10001111

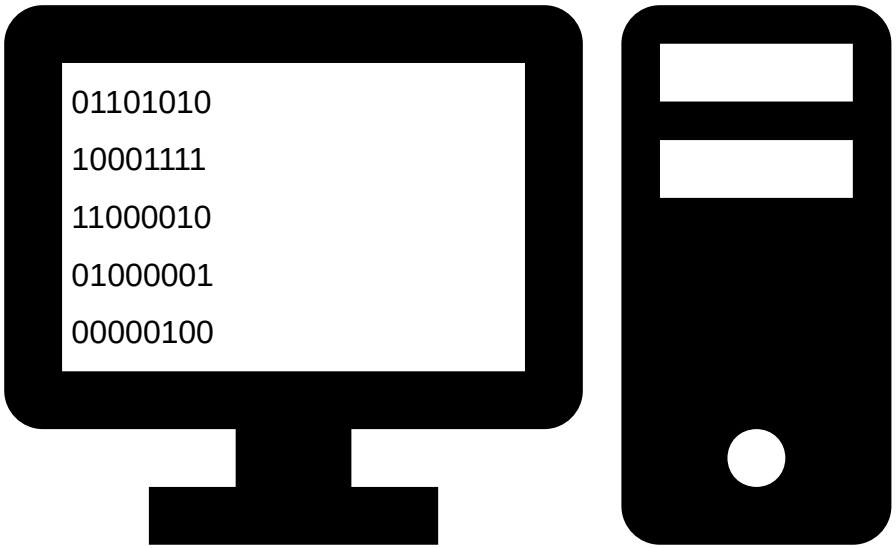
11000010

01000001

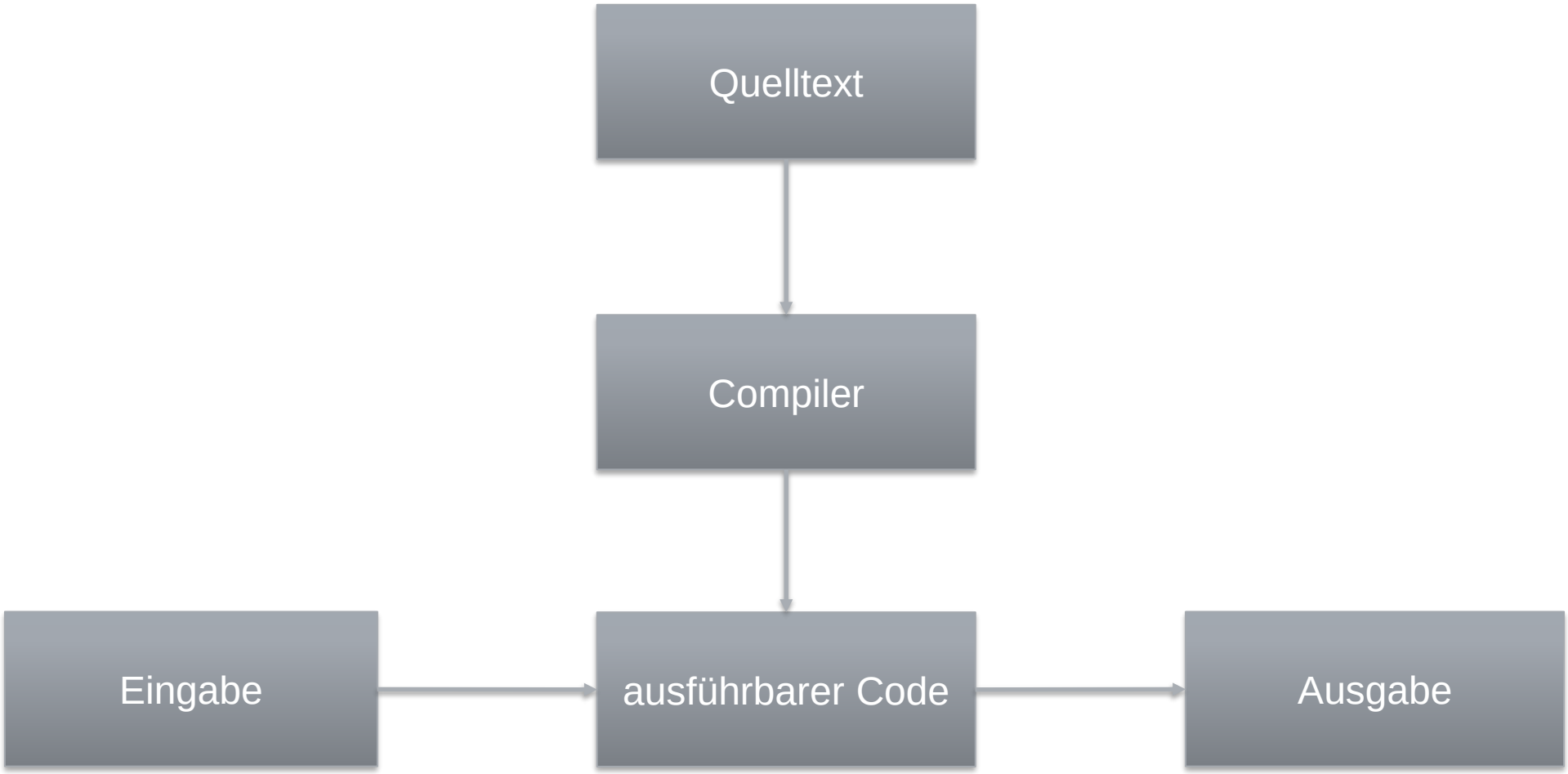
00000100

# Compiler

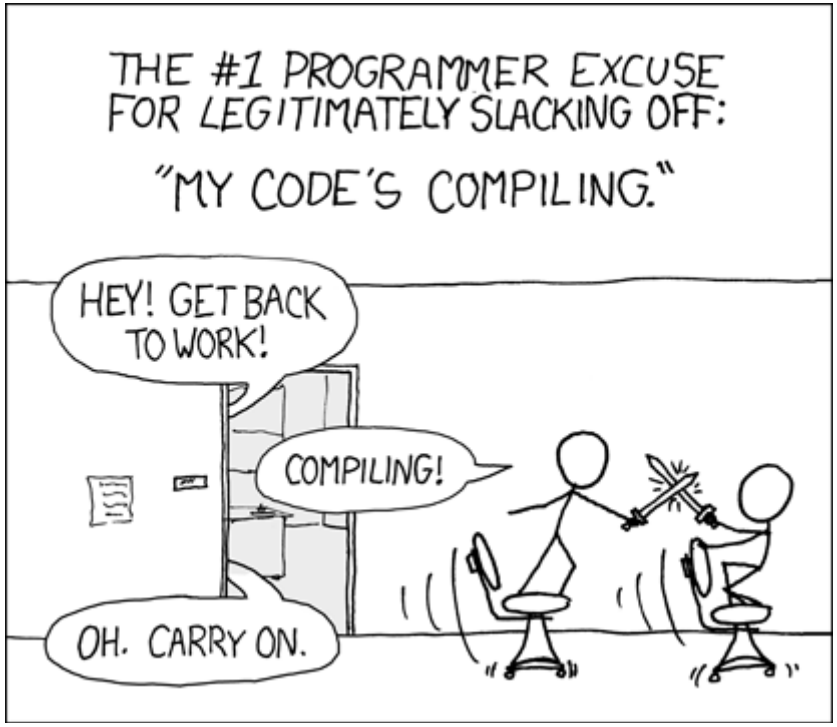
- Übersetzt kompletten Quellcode in Maschinencode
  - Genannt: Kompilieren
- Erst dann kann Code ausgeführt werden !



# Compiler



# Compiler




'Are you stealing those LCDs?' 'Yeah, but I'm doing it while my code compiles.'
Quelle: https://xkcd.com/303/ [Stand: 15.02.2022]

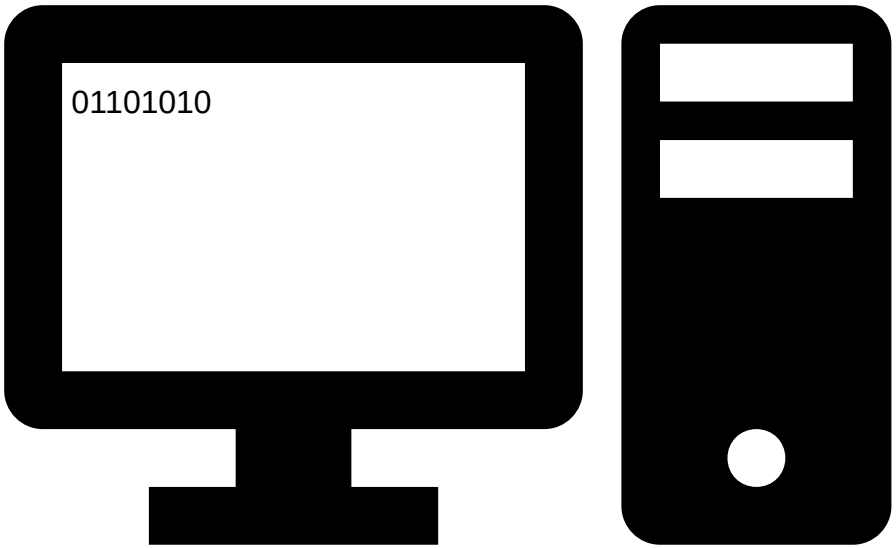
# Interpreter

- Übersetzt einzelne Anweisungen in Maschinencode während Ausführung
- Durchführung läuft „zeilenweise“ ab

```
IF X > 5 THEN
MSGBOX «Geben Sie eine Zahl <5 ein»
ELSE
MSGBOX «Perfekt»
END IF
```

# Interpreter

- Übersetzt einzelne Anweisungen in Maschinencode während Ausführung
- Durchführung läuft „zeilenweise“ ab




MSGBOX «Geben Sie eine Zahl <5 ein»

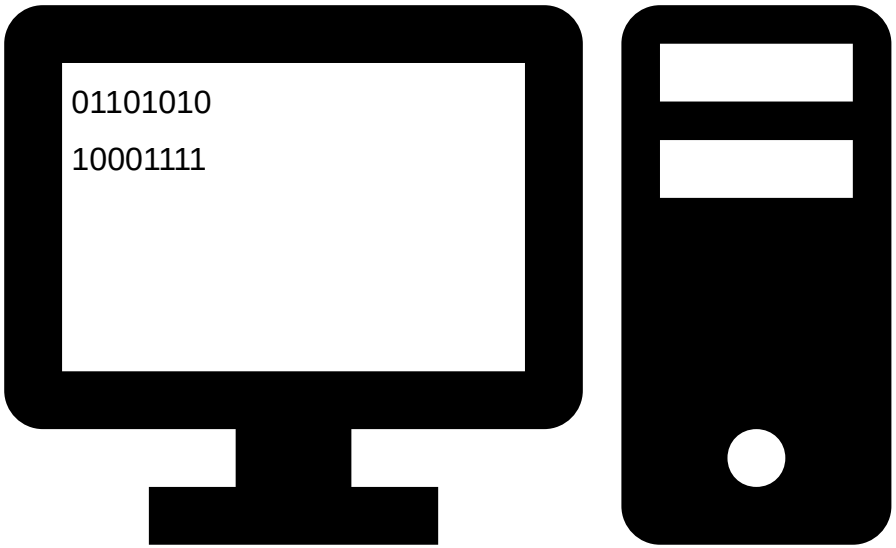
ELSE

MSGBOX «Perfekt»

END IF

# Interpreter

- Übersetzt einzelne Anweisungen in Maschinencode während Ausführung
- Durchführung läuft „zeilenweise“ ab



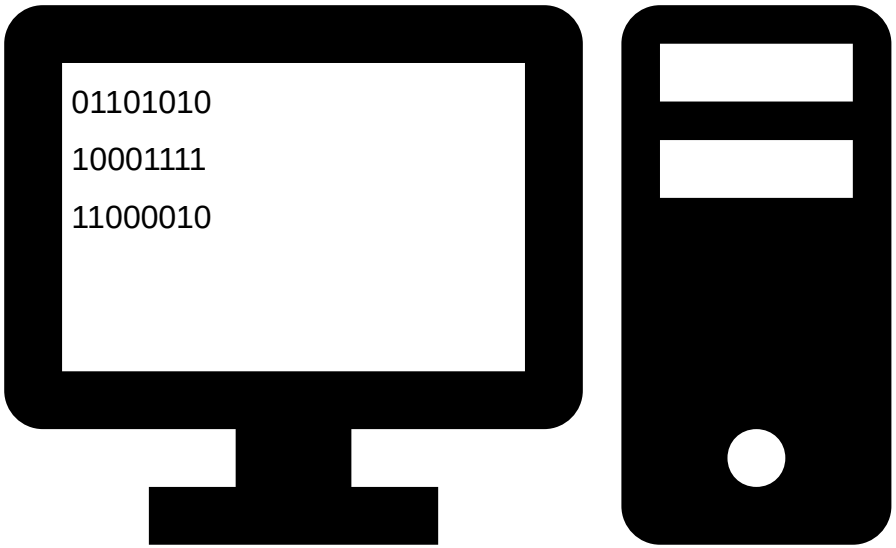

ELSE

MSGBOX «Perfekt»

END IF

# Interpreter

- Übersetzt einzelne Anweisungen in Maschinencode während Ausführung
- Durchführung läuft „zeilenweise“ ab

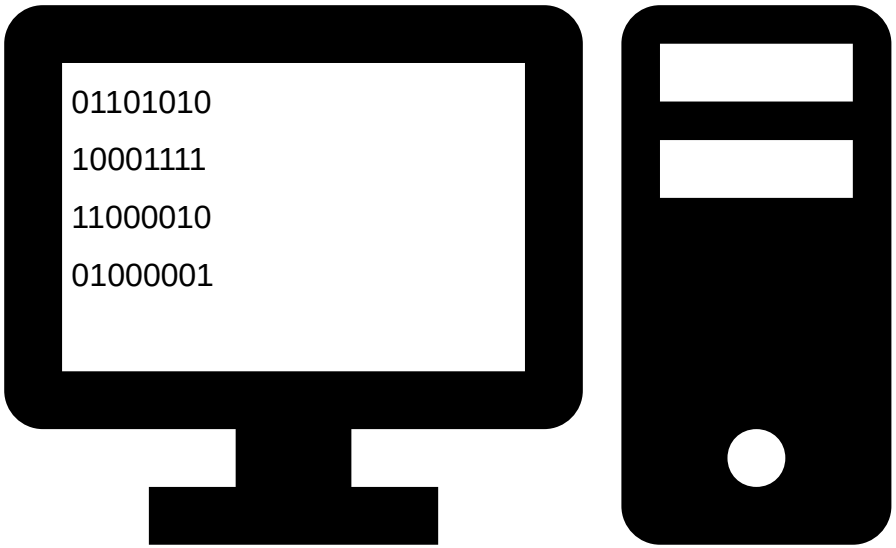



MSGBOX «Perfekt»

END IF

# Interpreter

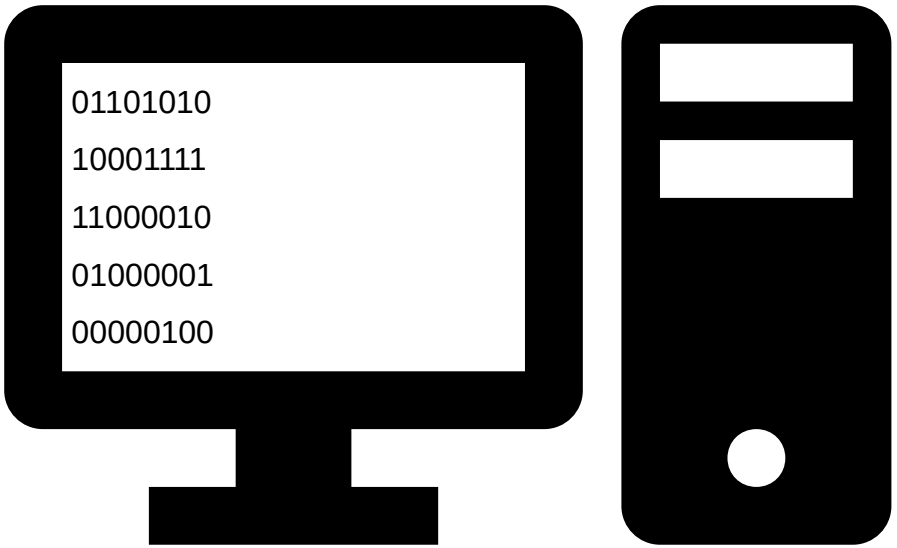
- Übersetzt einzelne Anweisungen in Maschinencode während Ausführung
- Durchführung läuft „zeilenweise“ ab



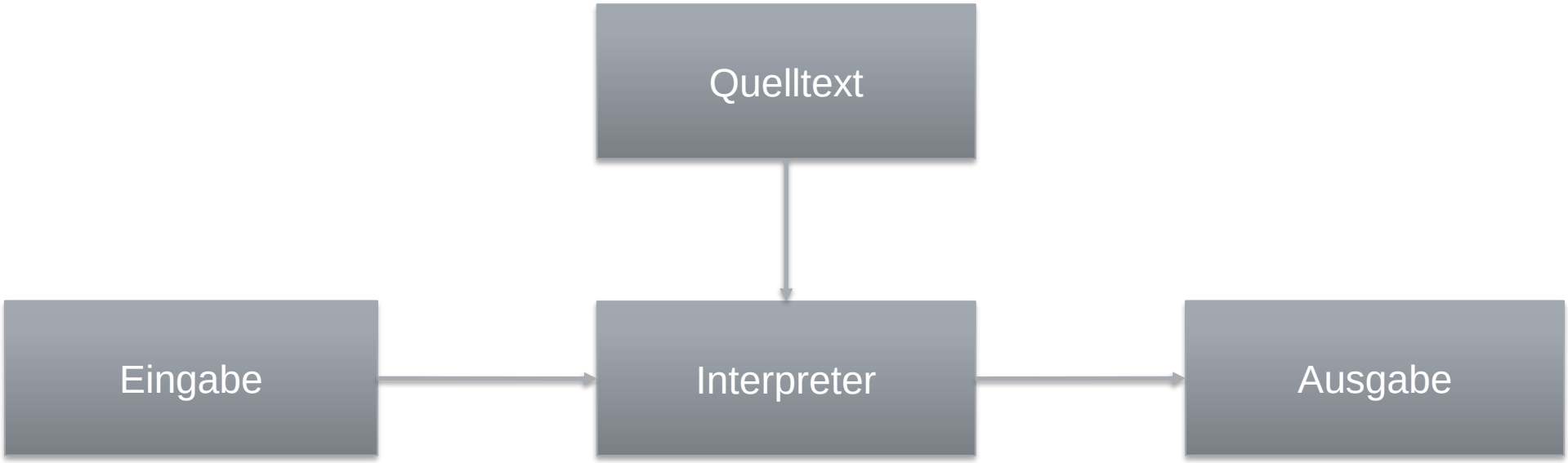

END IF

# Interpreter

- Übersetzt einzelne Anweisungen in Maschinencode während Ausführung
- Durchführung läuft „zeilenweise“ ab



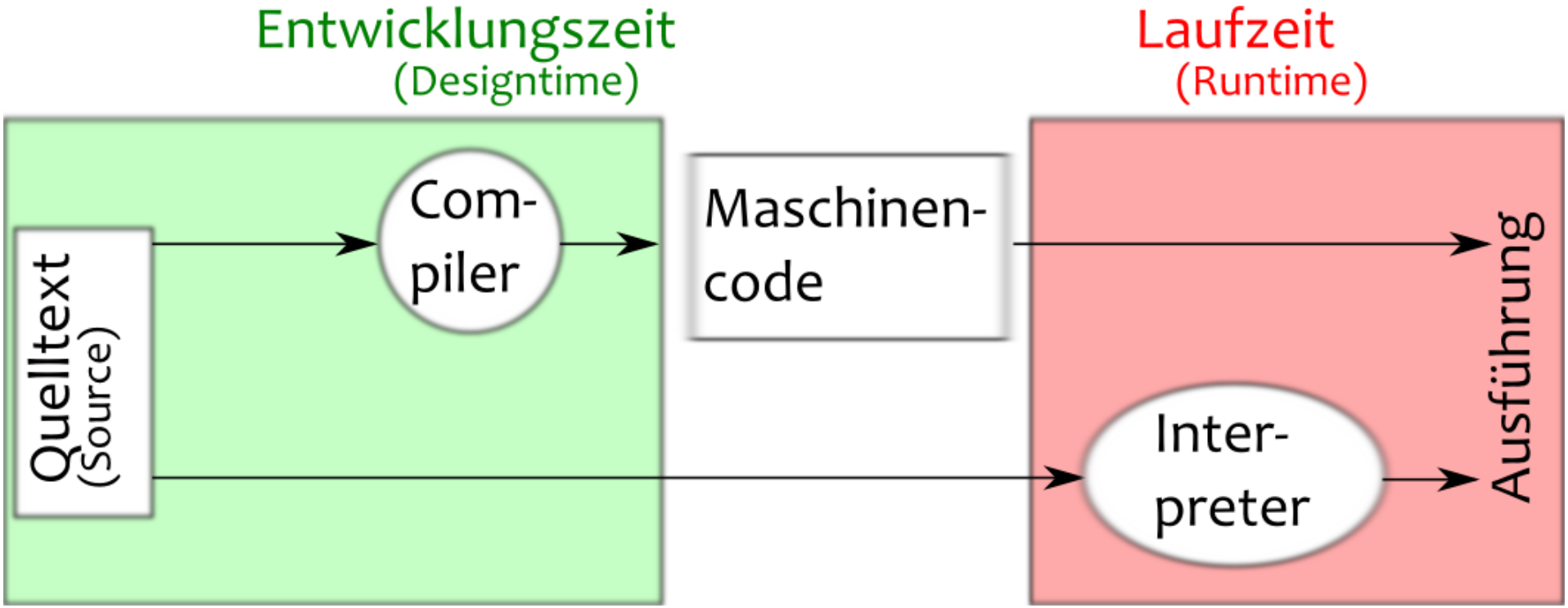
# Interpreter



# Interpreter

Quelle: https://www.reddit.com/r/ProgrammerHumor/comments/3fizjs/jit/
[Stand: 09.09.2022]

# Compiler & Interpreter




Quelle: http://www.roro-seiten.de/uhu/index.php?page=3972 [Stand: 09.09.2022]

# Compiler & Interpreter

## Vor- und Nachteile

| | Vorteile | Nachteile |
|---|---|---|
| Compiler | • Schnell bei Ausführung<br>• Syntaktische Fehler werden beim Kompilieren gefunden<br>• Compiler kann Code optimieren | • Kompilieren benötigt Zeit und Ressourcen<br>• Änderungen im Quelltext müssen erneut kompiliert werden<br>• Auf andere Plattform muss erneut kompiliert werden |
| Interpreter | • Kann sofort getestet werden<br>• Ausführbarer Code erst bei Laufzeit | • Ausführung langsamer und ineffizienter<br>• Gleiche Programmteile müssen immer wieder übersetzt werden<br>• Fehler tritt bei Laufzeit auf. Programm lief bis dahin |

# Mischform

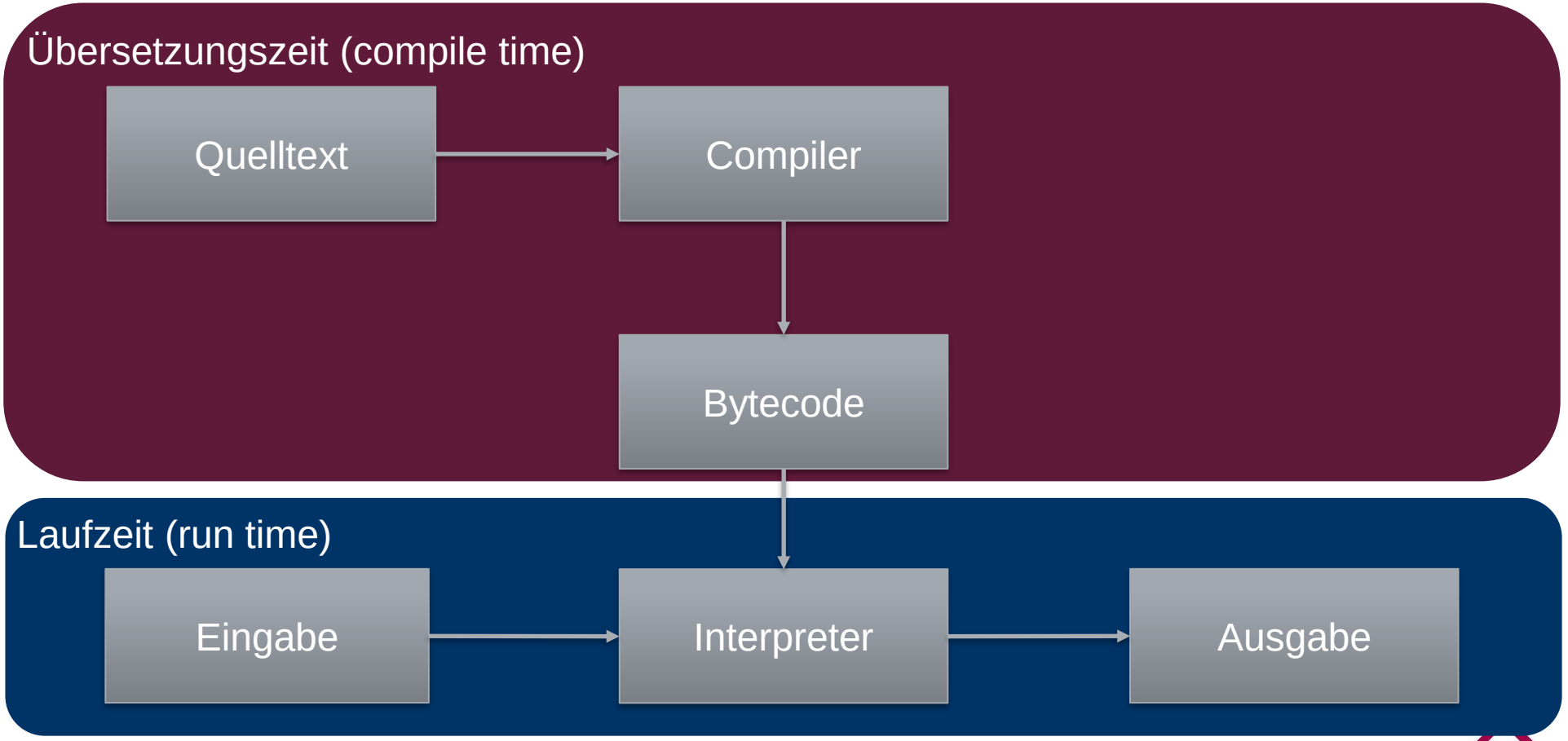
- Kombiniert Compiler mit Interpreter
  1. Quellcode wird in Bytecode kompiliert
  2. Bytecode wird zur Laufzeit interpretiert

- Interpretiertes Programm läuft auf einer virtuellen Maschine


Quelle: https://imgflip.com/memegenerator/167618150/Family-guy-Noahs-ark [Stand: 15.02.2022]. Bearbeitet

# Mischform

## Compiler + Interpreter

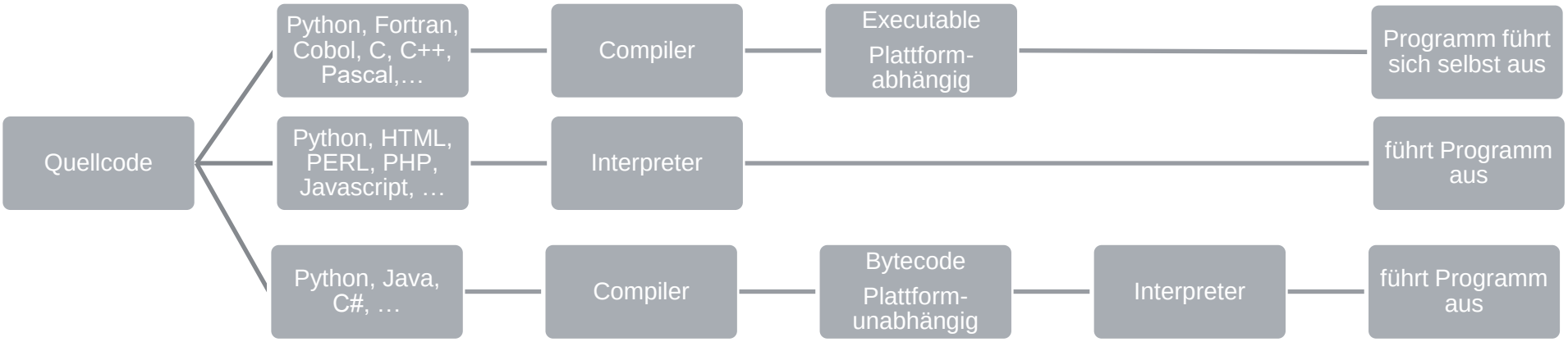


# Compiler & Interpreter

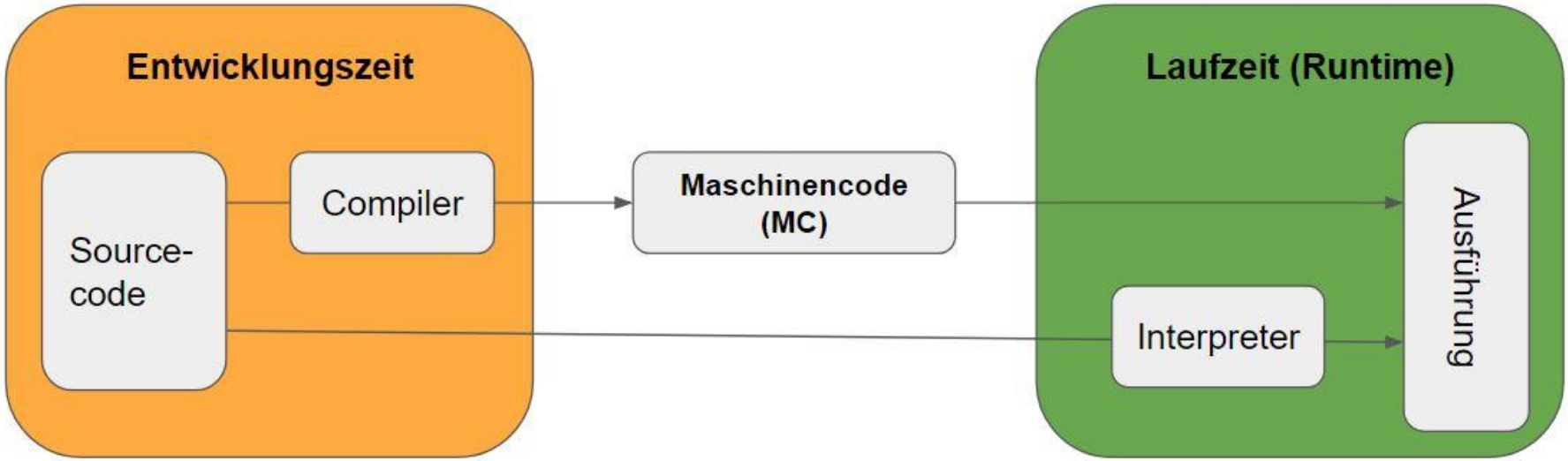
## Vor- und Nachteile

| | Vorteile | Nachteile |
|---|---|---|
| Compiler | • Schnell bei Ausführung<br>• Syntaktische Fehler werden beim Kompilieren gefunden<br>• Compiler kann Code optimieren | • Kompilieren benötigt Zeit und Ressourcen<br>• Änderungen im Quelltext müssen erneut kompiliert werden<br>• Auf andere Plattform muss erneut kompiliert werden |
| Interpreter | • Kann sofort getestet werden<br>• Ausführbarer Code erst bei Laufzeit | • Ausführung langsamer und ineffizienter<br>• Gleiche Programmteile müssen immer wieder übersetzt werden<br>• Fehler tritt bei Laufzeit auf. Programm lief bis dahin |
| Mischform | • Plattformunabhängig<br>• Es können Programmteile kompiliert werden | • Programm läuft in einer Virtuellen Maschine |

# Zusammenfassung

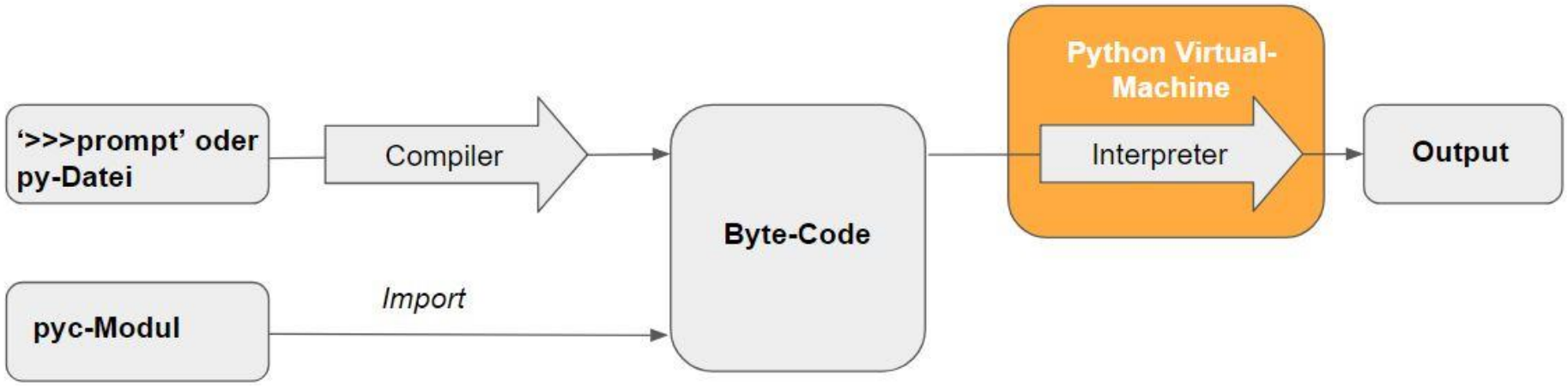


# Vom Source-Code zum “Ergebnis”




Achtung gilt nur für CPython!

# Vom Source-Code zum “Ergebnis”




Achtung gilt nur für CPython!

# Kompilieren & Interpretieren