Einheit 1 Grundlegenbegriffe

Objektorientierte Programmierung

Lukas Wais

CODERS.BAY

Version: 12. September 2023

Inhaltsverzeichnis I

Inhaltsverzeichnis II

Einführung in die Computerarchitektur

CPU

Abbildung: AMD Ryzen 7 CPU in den Sockel eingebaut. Foto von Olivier Collet auf Unsplash.

Verwendet Register, sehr schnell, aber klein und flüchtig

Eine alte und einfache CPU-Architektur

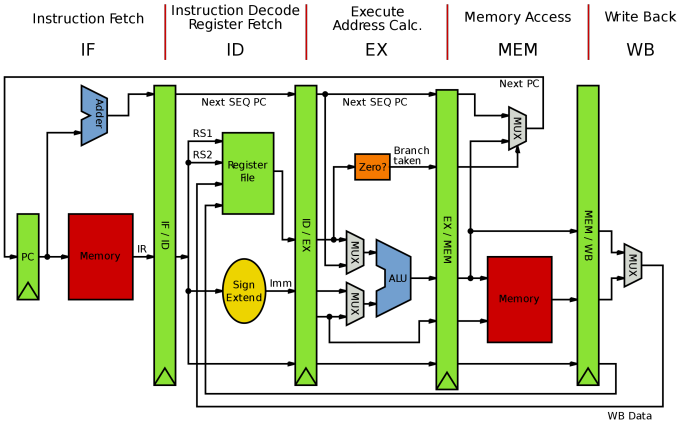


Abbildung: Pipelined MIPS Architecture

RAM

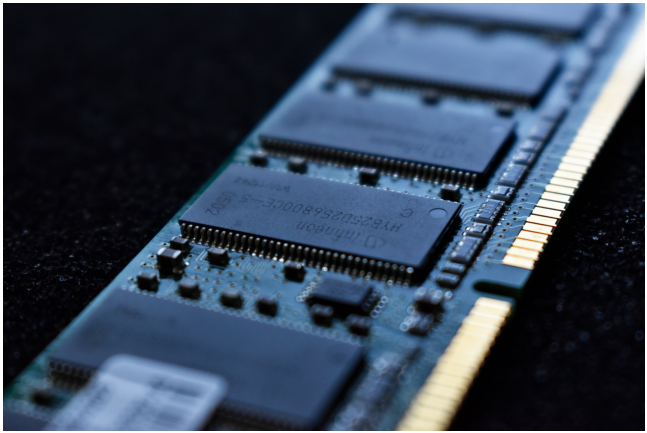


Abbildung: Nahaufnahme eines RAM-Sticks. Foto von Liam Briese auf Unsplash.

Flüchtig, aber schnell. Im Vergleich zu Registern langsam.

Wie der RAM Daten speichert

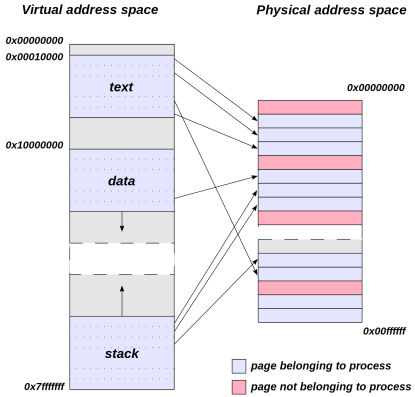


Abbildung: Abstraktion und Beispiel für die virtuelle Adressierung

HDD

Abbildung: Nahaufnahme einer Festplatte. Foto von Denny Müller auf Unsplash.

Nicht flüchtig, aber sehr langsam. Heutzutage werden in den meisten Fällen SSDs (Solid State Disks) verwendet.

PCB

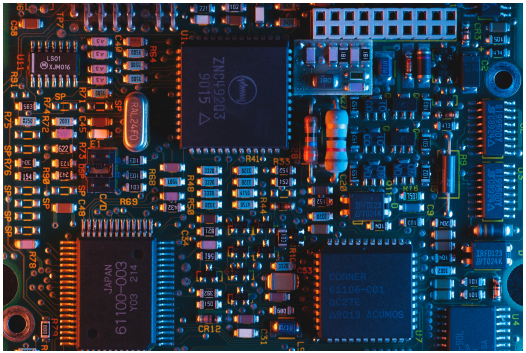


Abbildung: Ausschnitt einer Leiterplatte. Foto von Umberto auf Unsplash.

Alle Komponenten sind auf einer Platine zusammengelötet. Oft sehr spezialisiert.

Kompilierung

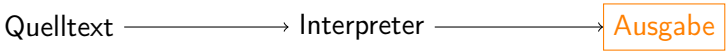
Was sind Programmiersprachen?

Was sind Programmiersprachen?

Eine Programmiersprache ist eine formale Sprache zur Formulierung von Datenstrukturen und Algorithmen, d. h. von Rechenvorschriften, die von einem Computer ausgeführt werden können. Sie setzen sich üblicherweise aus **schrittweisen Anweisungen** aus erlaubten (Text-)Mustern zusammen, der sogenannten Syntax.

Source: Wikipedia

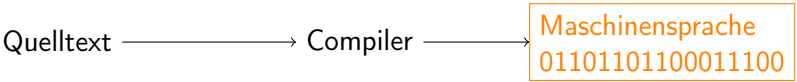
Skript-Sprachen



Skript-Sprachen - Beispiele

- Shell Script
- Batch Files
- PHP
- JavaScript

Kompilierte Sprachen



Kompilierte Sprachen - Beispiele

- C
- C++
- **Java und C#** [1]
- Go

[1] Sind JIT Kompiliert → Mischung aus beiden Konzepten.
Mehr dazu auf: RedHat Developers

Was ist Java?

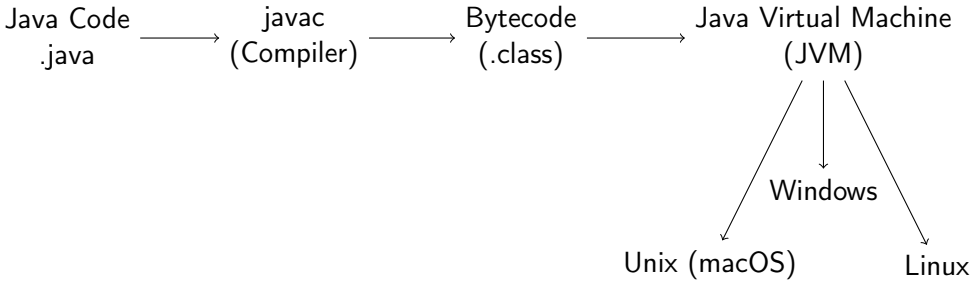
Java ist eine Insel

Abbildung: Mount Bromo, Indonesia. Foto von Thomas Ciszewski auf Unsplash.

Das ist auch Java

```java
@Override
public boolean equals(Object obj) {
    if (obj instanceof Person) {
        Person other = (Person) obj;
        return this.lastname.equals(other.lastname)
            && this.firstname.equals(other.firstname);
    }
    return false;
}
```

Java Kompilierung



Java Versionen

- JDK ... Java Development Kit $\rightarrow$ für die Entwicklung von Java-Anwendungen
- JRE ... Java Runtime Environment $\rightarrow$ für die Ausführung von Java-Anwendungen
- JDK > 8

Wir verwenden Java 17 ... den aktuellen LTS Release

Hello World in Java

```
System.out.println("Hello World");
```

C# Kompilierung

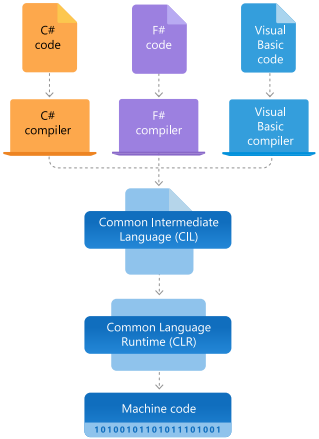


Abbildung: Architektur vom . NET Framework. Source:
https://dotnet.microsoft.com/en-us/learn/dotnet/what-is-dotnet-framework

Features von C#

Sehr ähnlich zu Java.
70% Java, 10% C++, 5% Visual Basic, 15% neu.

Wie in Java

- Object-orientation (single inheritance)
- Interfaces
- Generics (mächtiger als in Java)
- Exceptions
- Threads
- Namespaces (ähnlich Java packages)
- Attributes (ähnlich Java annotations)
- Lambda expressions
- Garbage collection
- Reflection
- Dynamic loading of code
- ...

Wie in C++

- Struct types
- Operator overloading
- Pointer arithmetic in unsafe code
- Some syntactic details

Wirklich neue Features

im Vergleich zu Java

- Reference parameters
- Stack-allocated objects (structs)
- Block matrices
- goto statement
- Delegates
- System-level programming
- Versioning
- Partial Types
- Anonymous types
- Nullable types
- Tuple types
- Nested methods
- Ranges

Syntactic Sugar

- Properties
- Indexers
- Events
- Iterators
- Extension methods
- SQL-like query expressions
- Asynchronous programming
- Pattern matching
- ...

C# Code

```
public override bool Equals(object obj)
{
  if ((obj == null) || !this.GetType().Equals(obj.GetType()))
  {
    return false;
  }
  else
  {
    Person p = (Person) obj;
    return (FirstName == p.FirstName) && (LastName == p.LastName);
  }
}
```

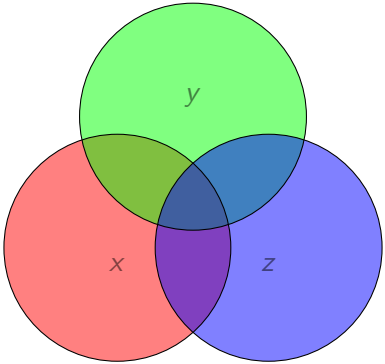
Hello World in C#

```
Console.WriteLine("Hello World");
```

Boolesche Algebra

Die Werte der Variablen sind die Wahrheitswerte *true* und *false*. Sie werden oft als 1 und 0 bezeichnet. Ein interessanter Artikel über die Mathematik der booleschen Algebra findet sich in der Stanford Encyclopedia of Philosophy.

Venn Diagram



AND

x	y	$x \wedge y$
0	0	0
0	1	0
1	0	0
1	1	1

OR

x	y	$x \vee y$
0	0	0
0	1	1
1	0	1
1	1	1

XOR

x	y	$x \veebar y$
0	0	0
0	1	1
1	0	1
1	1	0

NOT

x	$\neg x$
0	1
0	1
1	0
1	0

Einige Rechenvorschriften

- und bindet stärker als oder, vergleichbar mit Punkt vor Bindestrich Berechnung
- und kann mit dem Symbol $\cdot$ geschrieben werden
- oder kann mit dem Symbol $+$ geschrieben werden
- $\neg x$ kann als $\bar{x}$ geschrieben werden
- De Morgan Regel: $\bar{x} \cdot \bar{y} = \overline{x + y}$
- Gesetz der doppelten Verneinung: $\bar{\bar{x}} = x$
- Einige weitere Regeln und Schaltungen findet man hier

Was ist die Lösung?

x	y	z	$x \cdot y + z$
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Lösung

x	y	z	$x \cdot y + z$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Was ist die Lösung?

x	y	z	$\overline{x \cdot y \cdot z}$
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Lösung

x	y	z	$\overline{x \cdot y \cdot z}$
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Variablen Datentypen und Operatoren

Datentypen

Data Type	Size	Description
boolean	1 bit	wahr (true) oder falsch (false)
byte	1 byte	Zahlen von -128 bis 127
char	2 bytes	16-bit Unicode Character
short	2 bytes	Zahlen von -32.768 bis 32.767
int	4 bytes	Zahlen von -2^{31} bis $2^{31} - 1$
float	4 bytes	32-bit IEEE 754 Fließkommadarstellung
double	8 bytes	64-bit IEEE 754 Fließkommadarstellung
long	8 bytes	Zahlen von -2^{63} bis $2^{63} - 1$

Wie man eine Variable deklariert

```
int x = 3;
```

- Eine Variable ist ein Platzhalter für einen Wert mit einem bestimmten Typ
- Deklaration: `int x;`
- Initialisierung: `= 3;`

Einige Beispiele

In zwei Schritten und eine double Variable

```
int x;
x = 3;

double y = 4.267;
```

Special initializations

```
float myFloatingNumber = 123.456f;
char myCharacter = 's';
String name = "Lukas";
boolean javaIsFun = true;
```

Arithmetische Operatoren

Operator	Beschreibung
+	Addition und String-Verkettung
−	Subtraktion
*	Multiplikation
/	Division
%	Divisionsrest (modulo)

Unary Operators

Operator	Beschreibung
$+$	Unäres Plus, gibt positiven Wert an [2]
$-$	Unäres Minus, negiert den Ausdruck
$++$	Erhöht einen Wert um 1
$--$	Verringert einen Wert um 1
!	Logisches Komplement, invertiert den Wert eines booleschen Wertes

[2] Zahlen sind standardmäßig positiv

Gleichheits- und Beziehungsoperatoren

Operator	Beschreibung
==	gleich
!=	ungleich
>	größer als
>=	größer gleich
<	kleiner
<=	kleiner gleich

Bedingungsoperatoren

Operator	Beschreibung
&&	Bedingtes-AND
||	Bedingtes-OR
? :	Ternäroperator (kurz für if-then-else)

Bitweise und Bitverschiebungs-Operatoren

Operator	Beschreibung
~	Unäres bitweises Komplement
<<	Vorzeichenbehaftete Linksverschiebung
>>	Vorzeichenbehaftete Rechtsverschiebung
>>>	Vorzeichenlose Rechtsverschiebung
&	Bitweises AND
^	Bitweises exklusives OR
|	Bitweises inklusives OR

Typvergleichsoperator in Java

`instanceof`

Typvergleichsoperator in C#

```
is
```

Erste Klassen

Erste C# Klasse

```
public class Main
{
    public static void Main(string[] args)
    {
        Console.WriteLine("Hello C# World!");
    }
}
```

Erste Java Klasse

```
public class Main {
    public static void main(String[] args) {
        System.out.println("Hello Java World!")
    }
}
```