



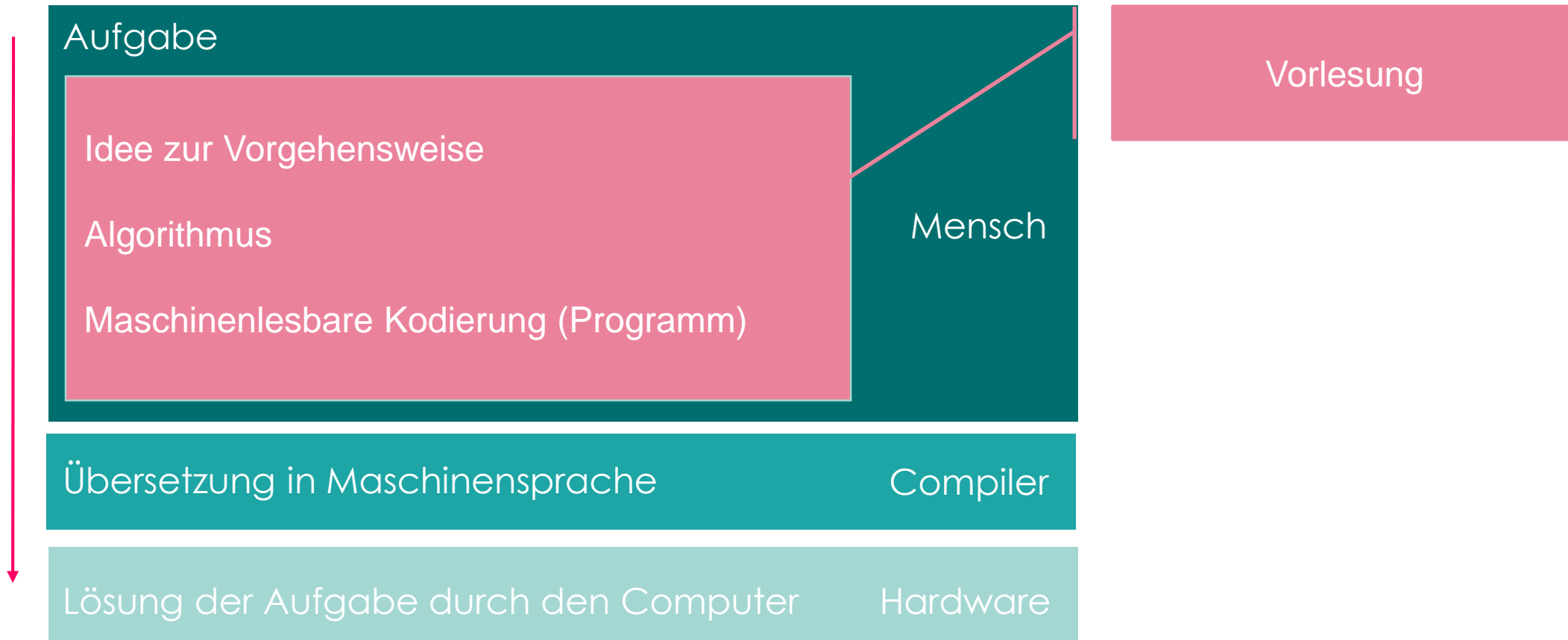
University  
of Basel

# Einige Grundlagen der Informatik

M. Lüthi – Universität Basel

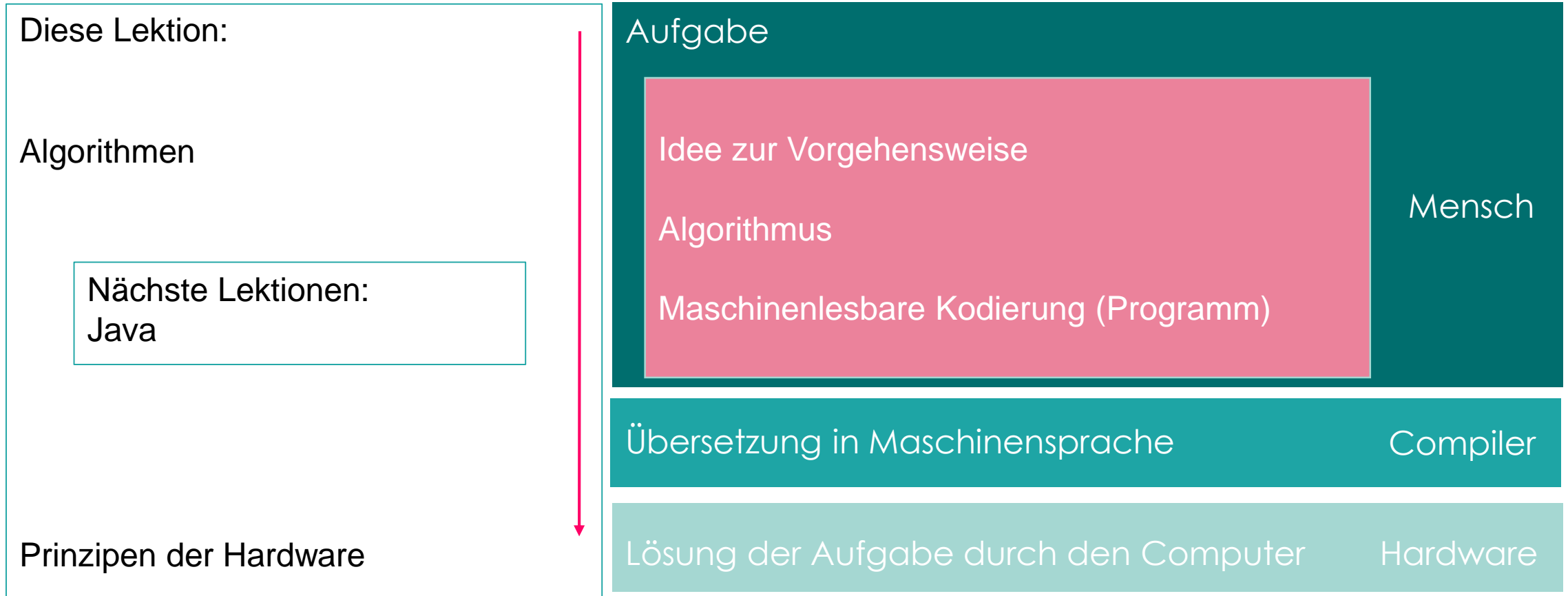
# Was heisst Programmieren

Exaktes Instruieren eines Computers, eine bestimmte Aufgabe zu lösen.



# Agenda

Trennung Algorithmus und Hardware in Java nicht ganz perfekt.

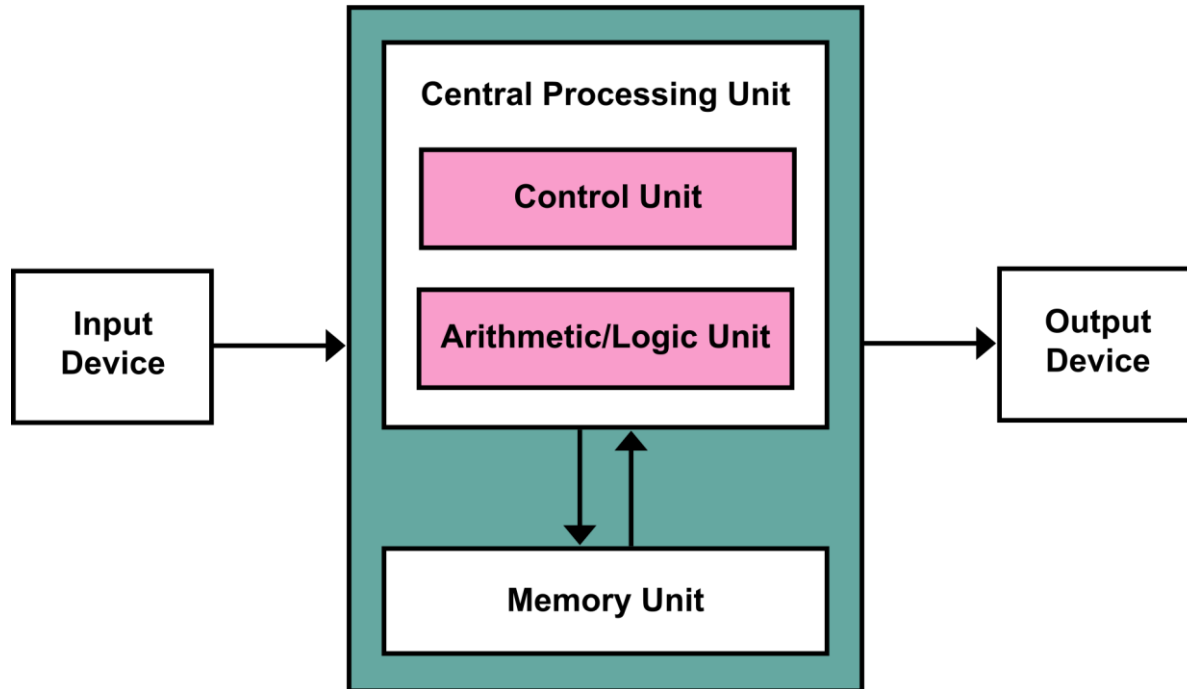




University  
of Basel

# Hardware

# Von-Neumann Architektur



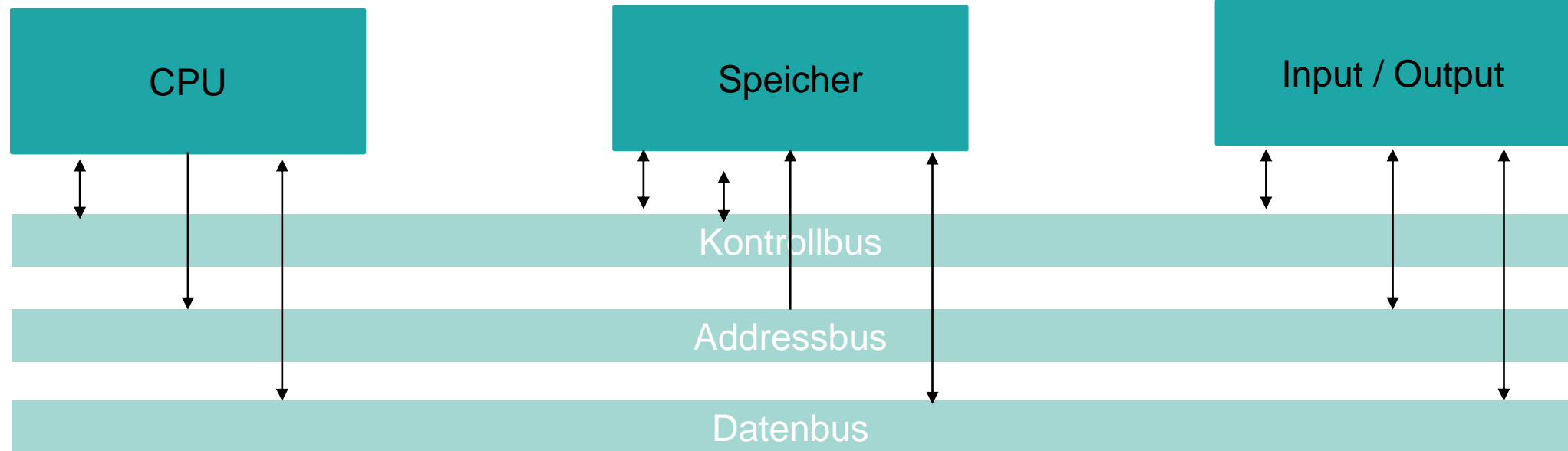
By Kapoht - Own work, CC BY-SA 3.0,  
<https://commons.wikimedia.org/w/index.php?curid=25789639a>

Grundsätzlicher Aufbau moderner Computer

---

# Speicherzugriff

Adresse	Inhalt
1	7
2	«Hallo»
3	😊



# Speicher

Kleinste Speichereinheit hat 2 Zustände

- 1 Bit
- Zustände werden i.A. mit 0 und 1 bezeichnet

Mit 2 Speichereinheiten  $2^2=4$  Zustände darstellbar

Mit 8 Bit  $2^8=256$  Zustände darstellbar

8 Bit = 1 Byte

Heutzutage sind Bytes die kleinsten adressierbaren Speichereinheiten  
Kleinere Einheiten müssen aus einem Byte extrahiert werden

Bit 0	Bit 1	Zustand
0	0	0
0	1	1
1	0	2
1	1	3

# Intermezzo: Zahlensystem

Ein **Zahlensystem** (number system) besteht aus

- endlich vielen Ziffern (digits) und
- einer Vorschrift  
wie werden Zeichenreihen als Zahl interpretiert?

Arabische Zahlensysteme zur Basis  $\beta$

- Natürliche Zahl  $z$  wird geschrieben als Polynom

$$z = \sum_{i=0}^{n-1} z_i \beta^i, \quad 0 \leq z_i \leq \beta$$



# Intermezzo: Binärsystem

Basis  $\beta = 2$

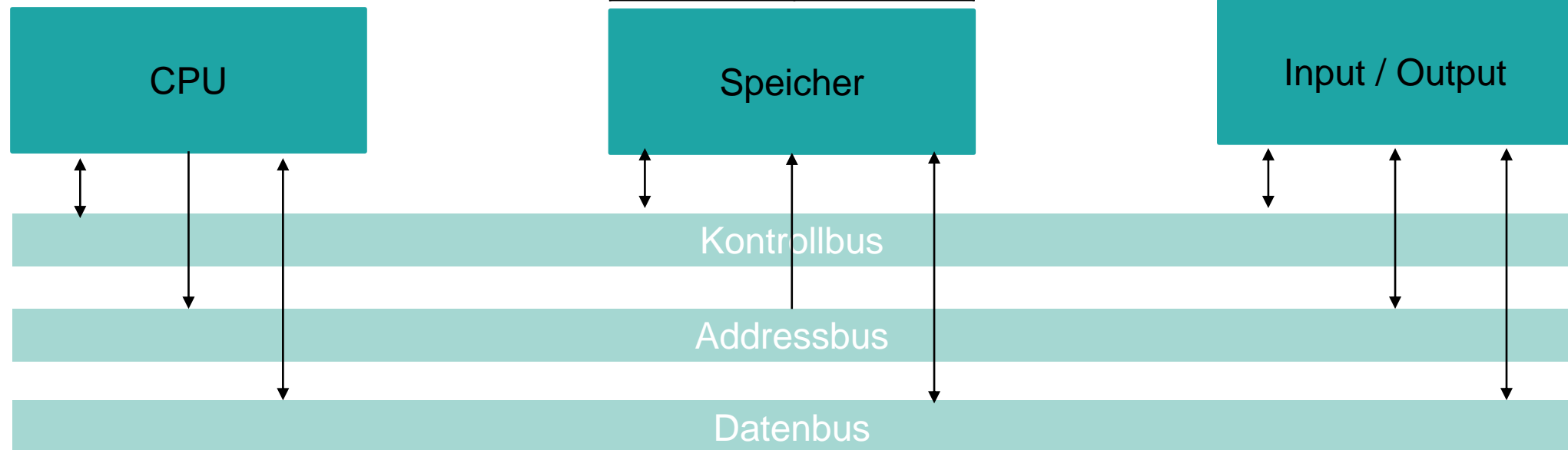
$$z = \sum_{i=0}^{n-1} z_i 2^i, \quad 0 \leq z_i \leq 1$$

Beispiele:

- $1_d = 1 \cdot 2^0 = 1_b$
  - $7_d = 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 111_b$
  - $9_d = 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 1001_b$
-

# Speicher & Adressierung

Adresse	Inhalt
00000001	01011011
00000010	11100001
00000011	10010111
00000100	00011011
00000101	11000111



# Quiz: Speicher

Wodurch wird der maximal mögliche Speicher in einem Computer begrenzt?

Wie viele Bits können auf einmal aus dem Speicher gelesen werden oder in den Speicher geschrieben werden?

Welcher Dezimalzahl entspricht die Zahl 1100111?

Welcher Binärzahl entspricht die Zahl 12?

Welcher Zahl im 16-er System (Hexadezimalsystem) entspricht die Dezimalzahl 65?

---

# Variablen

Variablen sind Namen für Speicheradressen

- Einfacher zu merken als Zahl

Variable hat einen Wert

- Inhalt des Speichers an dieser Adresse

Variable hat einen Typ

- Wie gross ist Speicherstelle?
- Wie muss Bitmuster interpretiert werden?

Kleine Zahl  $x = 11$

Grosse Zahl  $y = 257$

Zeichen  $z = 'a'$

Adresse	Inhalt
00000000	00001011
00000001	00000001
00000010	00000001
00000011	10010001
00000100	11000111

# Elementare Datentypen

<b>byte</b>	8 Bit Zahl	$-2^7 \dots 2^7 - 1$	( -128, .... , 127 )
<b>short</b>	16 Bit-Zahl	$-2^{15} \dots 2^{15} - 1$	( -32768, .... , 32767 )
<b>int</b>	32 Bit-Zahl	$-2^{31} \dots 2^{31} - 1$	( -2 147 483 648, .... , 2 147 483 647 )
<b>long</b>	64 Bit-Zahl	$-2^{63} \dots 2^{63} - 1$	
<b>float</b>	32 Bit IEEE-754-1985 Gleitkommazahl		
<b>double</b>	64 Bit IEEE-754-1985 Gleitkommazahl		
<b>char</b>	16 Bit Unicode		
<b>boolean</b>	Wahrheitswert, <i>false</i> oder <i>true</i>		

---

# Gleitkommazahlen

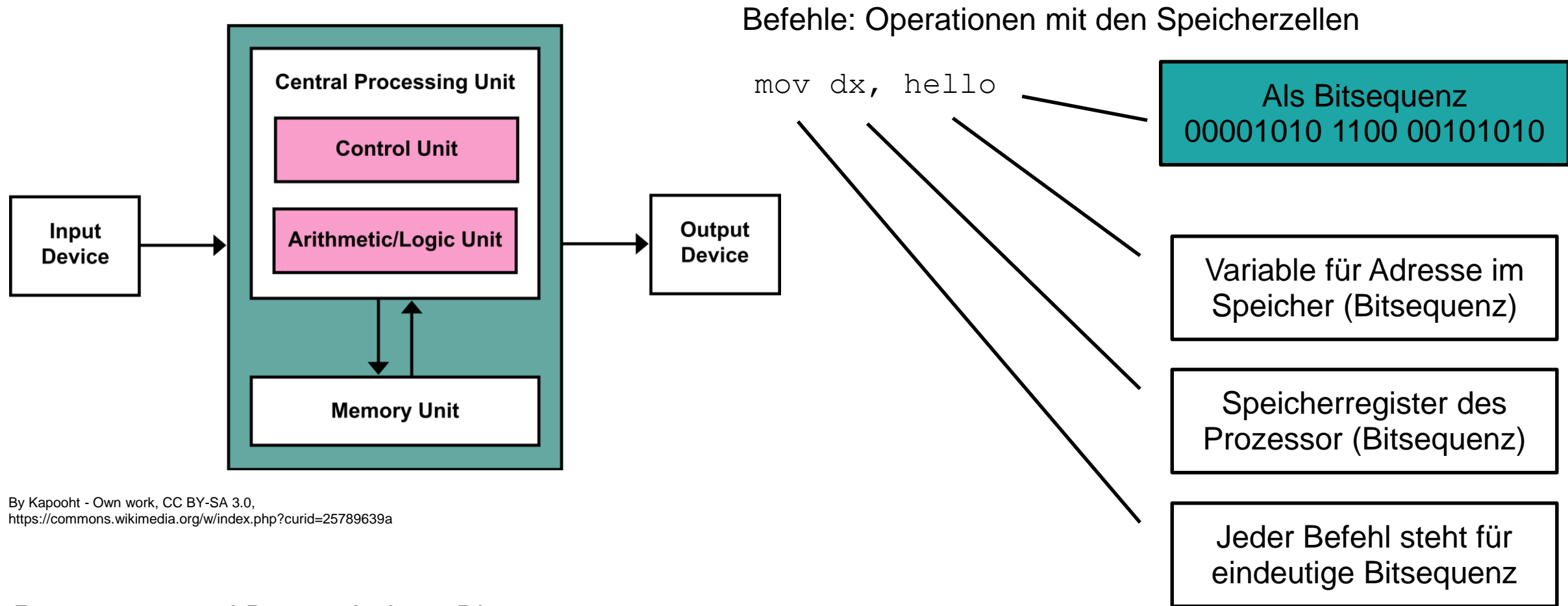
Darstellung einer Floating-Point-Zahl (IEEE 754-1985)

$$z = (-1)^v \cdot \text{Mantisse} \cdot 2^{\text{Exponent}}$$

<b>float</b>	v	Exponent	Mantisse
	1 Bit	8 Bit	23 Bit
<b>double</b>	v	Exponent	Mantisse
	1 Bit	11 Bit	52 Bit

---

# Von-Neumann Architektur : Befehle



By Kapoht - Own work, CC BY-SA 3.0,  
<https://commons.wikimedia.org/w/index.php?curid=25789639a>

*Programme und Daten sind nur Bitsequenzen*

- *Sind im selben Speicher gespeichert (Charakteristik von Neumann architektur)*

# Diskutieren Sie

Jedes Programm, das wir schreiben, wird irgendwann in Maschinensprache übersetzt.

- Weshalb programmiert man typischerweise nicht direkt in Maschinensprache?
  - Was könnten Vorteile sein, direkt in Maschinensprache zu programmieren?
  - Gibt es Programme, die in Maschinensprache geschrieben werden müssen?
-





University  
of Basel

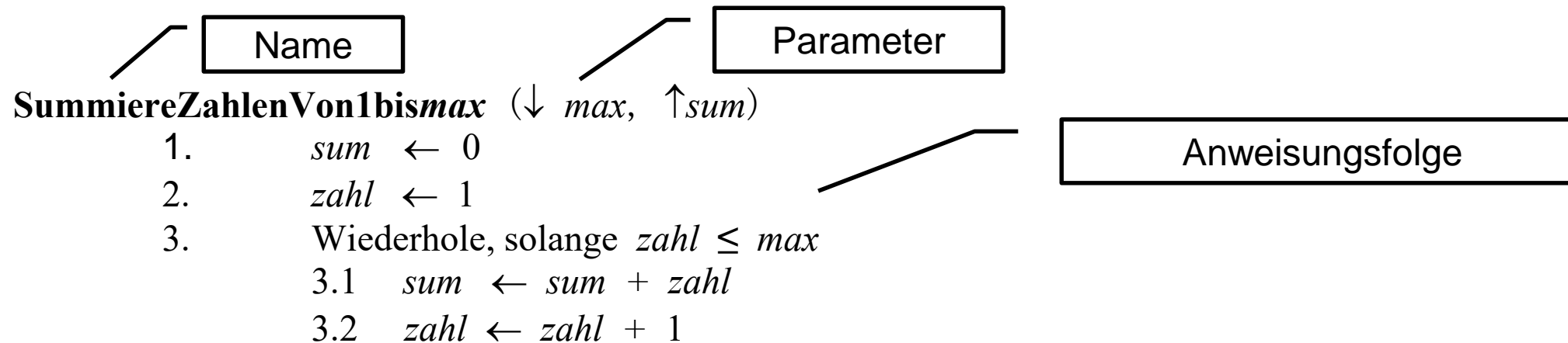
# Algorithmen

# Algorithmus

## Algorithmus:

Schrittweises, präzises Verfahren zur Lösung eines Problems

Problem: Summiere die Zahlen von 1 bis  $max$ :  $sum = \sum_{i=1}^{max} i$



## Programm

Beschreibung eines Algorithmus in einer Programmiersprache

# Variablen

Variablen: Behälter für Werte



Variablen können Wert ändern  
 $x \leftarrow x + 1$

x
100

Variablen haben Datentyp (Menge erlaubter Werte)

Variablentyp

Werte

*Zahl*

...

- in eine Zahlenvariable passen nur Zahlen

*Zeichen*

...

- in eine Zeichenvariable passen nur Zeichen

---

# Strukturelemente

Strukturelemente um einen Algorithmus zu formulieren

- Anweisungen und Anweisungsfolgen
- Verzweigungen
- Wiederholungen (Schleifen)

*Grundelemente der Strukturierten Programmierung!*

---

# Anweisungen

Beispiel: Wertzuweisung

$$x \leftarrow x + 1$$


Variable

Ausdruck

1. Werte Ausdruck aus
2. Weise seinen Wert der Variablen zu

*Beliebige, klar definierte Anweisungen sind möglich*

- *Keine Einschränkung durch Programmiersprache oder Hardware*
-

# Anweisungsfolge (Sequenz)

calculation ( $\downarrow x$ ,  $\downarrow y$ ,  $\uparrow z$ )

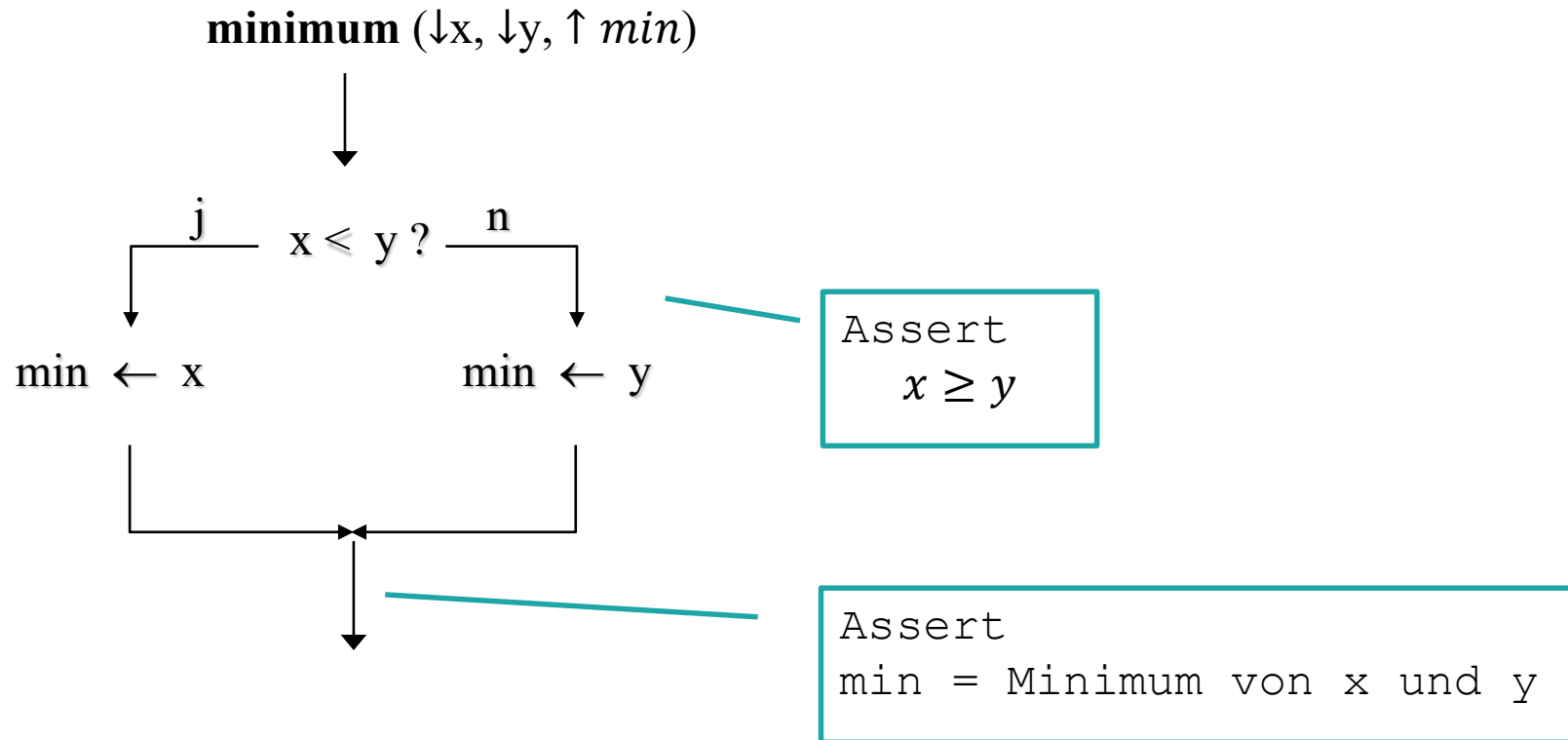
$\downarrow$   
 $x \leftarrow 2$   
 $y \leftarrow 4$   
 $x \leftarrow x + 1$   
 $z \leftarrow x + y$   
 $\downarrow$

Assert  
 $x = 3, y = 4, z = 7$

*Assertion (Zusicherung)*  
Aussage über den Zustand des Algorithmus  
an einer bestimmten Stelle

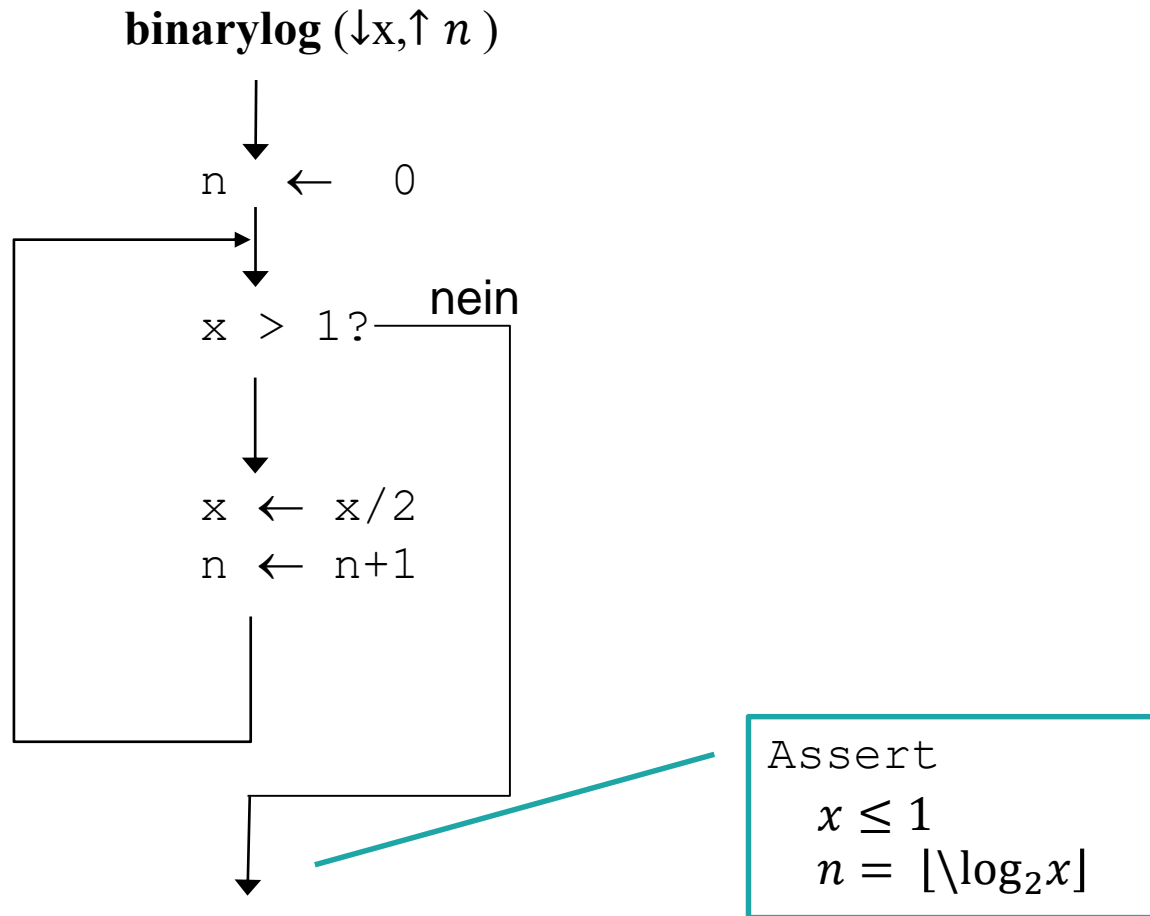
# Auswahl (Verzweigung)

Beispiel: Suche das Minimum der zwei Zahlen  $x$  und  $y$



# Wiederholung (Schleife, Iteration)

Beispiel: Suche die grösste ganze Zahl  $n$  mit  $2^n$  kleiner oder gleich  $x$ .





# Beispiel: Vertausche zwei Variableninhalte

**swap** ( $\uparrow x$ ,  $\uparrow y$ )



$h \leftarrow x$

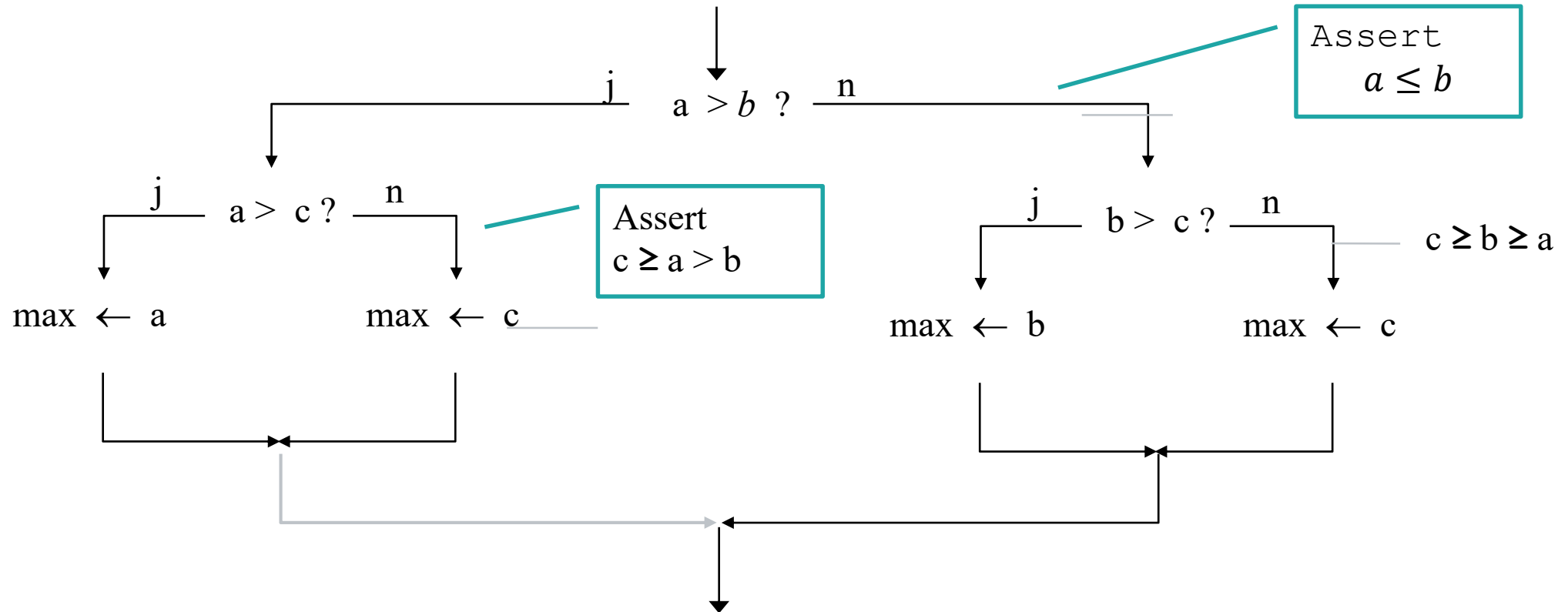
$x \leftarrow y$

$y \leftarrow h$



# Beispiel: Bestimme Maximum dreier Zahlen

Max ( $\downarrow a, \downarrow b, \downarrow c, \uparrow max$ )



# Beispiel: Euklidischer Algorithmus

Berechnet den größten gemeinsamen Teiler zweier Zahlen  $x$  und  $y$

