



**Aufgabe 1.** Installiere einen Compiler auf deinem Computer und kompiliere ein Hallo-Welt-Programm. Informationen dazu und einen Download-Link für Cygwin findest du im Skript auf der Webseite des Kurses:

<http://www.ah-effect.net/>

Versucht, die folgenden Aufgaben zu lösen, indem ihr Euch die fehlenden Sprachkonstrukte aus dem Skript aneignet und die Tutoren gegebenenfalls um Hilfe bittet. Wenn Ihr die Aufgaben nicht bewältigt, macht Euch keine Sorgen: Wir erklären die fehlenden Sprachkonstrukte auch in der morgigen Vorlesung. Hauptsache ist, dass ihr es versucht.

**Aufgabe 2.** Der folgende Code ist *Pseudocode*, d.h. eine etwas formaliere als umgangssprachliche Beschreibung eines Algorithmus. Es ist jedoch *kein* C-Code. Schreibe zu jedem der folgenden Algorithmen ein C-Programm, welches die gleichen Schritte durchführt.

Finde währenddessen oder mit Hilfe deines Programms heraus, was die Algorithmen berechnen.

---

### Algorithmus 1

---

**Input:** Ganze Zahl  $c \in \mathbb{N}$

**Output:** Entweder Ja oder Nein.

```
1: set  $n := 2$ .
2: if  $n > \sqrt{c}$  then
3:   return Ja
4: end if
5: if  $n$  teilt  $c$  then
6:   return Nein
7: end if
8: set  $n := n + 1$ 
9: goto 2
```

---



---

**Algorithmus 2**

---

**Input:** Ganze Zahlen  $a, b \in \mathbb{N}$ **Output:** Eine ganze Zahl  $k \in \mathbb{N}$ 

```
1: if  $a = 0$  then
2:   return  $b$ 
3: end if
4: if  $b = 0$  then
5:   return  $a$ 
6: end if
7: if  $a > b$  then
8:   set  $a = a - b$ 
9: else
10:  set  $b = b - a$ 
11: end if
12: goto 4
```

---

---

**Algorithmus 3**

---

**Input:** Reelle Zahl  $a \in \mathbb{R}_{\geq 0}$ **Output:** Reine reelle Zahl  $x \in \mathbb{R}$ 

```
1: set  $x := 2$  und  $y := 1$ .
2: if  $|x - y| \leq 10^{-10}$  then
3:   return  $x$ 
4: end if
5: set  $x := y$ 
6: set  $y := \frac{1}{2} \cdot \left(x + \frac{a}{x}\right)$ 
7: goto 2
```

---