

# Das Halteproblem

IMPOSSIBLE

```
int i;  
i = Integer.parseInt(eingabe);  
while(i!=10)  
{  
    System.out.println(i);  
    i= i + 2;  
}
```

- Hält das gegebene Programm für jede mögliche Benutzereingabe? Teste Werte wie 0,1,2,... Begründe deine Aussage und gebe an, für welche Werte das Programm anhält und für welche nicht.

## Halteproblem

Kann man einen Algorithmus definieren, der entscheidet, ob ein gegebenes Programm auf einer gegebenen Eingabe nach endlich vielen Schritten hält oder nicht?

## Der Beweis

Wir schauen uns im Folgenden eine Spezialisierung des Halteproblems an. Es handelt sich um das Selbstanwendungsproblem, woraus die Unlösbarkeit des Halteproblems folgt.

$H = \{x \mid x \text{ ist ein Programm, das, wenn man es auf die Eingabe } x \text{ anwendet, also auf sich selbst, nach endlich vielen Schritten hält}\}$

Angenommen das Halteproblem wäre entscheidbar, dann gibt es eine Black Box, die Folgendes berechnet:

$x \longrightarrow$  [Black Box]  $\longrightarrow$  1, falls  $x$  bei Eingabe  $x$  stoppt  
 $\longrightarrow$  0, falls  $x$  bei Eingabe  $x$  nicht stoppt

Diese bauen wir um, so dass sie stoppt, falls die Black Box 0 ausgibt. Falls die Black Box 1 ausgibt, geht die neue Box in eine Endlosschleife.

$x \longrightarrow$  [Black Box]  $\longrightarrow$  Endlosschleife  
 $\longrightarrow$  (stoppt)

Es sei  $z$  der Programmtext der neuen Box. Wenn  $z$  auf  $z$  stoppt, dann heißt das, dass die BlackBox für  $z$  den Wert 0 ausgibt. Da die Black Box aber gerade ein Entscheider für das Halteproblem ist, bedeutet das, dass  $z$  auf  $z$  in einer Endlosschleife ist, ein Widerspruch.

## Alan Turing und das Halteproblem

Die Frage, ob ein Programm auf einer beliebigen Eingabe hält oder nicht, ist sowohl von theoretischer als auch von praktischer Seite bedeutend. Aus praktischer Sicht ist es natürlich wünschenswert, wenn ein Programm nie in eine Endlosschleife läuft. Auch bei rechenintensiven Programmen, die über Wochen Berechnungen durchführen, wäre es interessant zu wissen, ob sie noch ein Ergebnis liefern oder schon in der Endlosschleife sind. Alan Turing bewies als erster, dass es keinen Algorithmus gibt, der diese Frage für alle möglichen Algorithmen und beliebige Eingaben beantwortet. Diesen Beweis vollzog er mithilfe der von ihm definierten Turingmaschine.



- Alan Turing hat die Unlösbarkeit des Halteproblems für die Turing Maschine gezeigt. Wieso gilt dies auch für Personal Computer, Handys und Tablets?



- Schreibe ein Programm *perfect\_number*, das für eine gegebene Zahl ausgibt, ob sie eine vollkommene Zahl ist.
- Verändere das Programm so, das es nun sucht, ob es eine ungerade vollkommene Zahl gibt.
- Was würde es für *perfect\_number* bedeuten, wenn das Halteprogramm lösbar wäre?

### Vollkommene Zahlen

Eine natürliche Zahl  $n$  heißt vollkommene Zahl, wenn sie gleich der Summe all ihrer positiven Teiler außer sich selbst ist.

**Beispiel:** Die Zahl 6 hat die Teiler 3,2,1. Es gilt  $3+2+1=6$ . Also ist 6 eine vollkommene Zahl.

**Offenes Problem:** Es ist ein offenes mathematisches Problem, ob es eine ungerade vollkommene Zahl gibt.