3 Prinzipielle Grenzen der Berechenbarkeit

Ulam-Funktion

$$ulam(n) = \begin{cases} 1, & \text{falls } n = 1 \\ ulam\left(\frac{n}{2}\right), & \text{falls } n \text{ gerade } (n > 1) \\ ulam(3 \cdot n + 1), & \text{falls } n \text{ ungerade } (n > 1) \end{cases}$$

Man sagt, ein Algorithmus **terminiert** für die Eingabe n, wenn er für diese Eingabe nach endlich vielen Arbeitsschritten zu einem Ende kommt, so dass die Berechnung in endlicher Zeit abgeschlossen wird.

Terminiert die Ulam-Funktion für jede beliebige Eingabe?

Unbekannt!

Einschränkungen für die Berechenbarkeit:

- Ist die Fragestellung formal nicht beschreibbar, kann keine Lösung berechnet werden.
- Es gibt Aufgabenstellungen, deren Berechnung mehr Zeit kostet, als aufgewendet werden kann.

Frage:

Gibt es eine Methode, die überprüfen kann, ob eine bestimmte Methode für eine bestimmte Eingabe terminiert oder nicht?

Pseudocode:

```
HALT (methode, eingabe)
        wenn methode(eingabe) terminiert
                gib true zurück
        sonst
                gib false zurück
```

Widerspruchsbeweis:

Betrachte folgende Methode:

```
TEST(methode)
        solange HALT(methode, methode)
                //keine Aktion
        endesolange
```

TEST terminiert genau dann ….

Test terminiert genau dann, wenn die methode mit sich selbst als Eingabe nicht terminiert.

```
TEST(methode)
        solange HALT(methode, methode)
                //keine Aktion
        endesolange
```

Rufe nun TEST(TEST) auf.

- HALT(TEST, TEST) liefert true, d.h. Methode TEST terminiert mit sich selbst als Eingabe, dann Endlosschleife
- HALT(TEST, TEST) liefert false, d.h. keine Terminierung, dann terminiert TEST(TEST)

Widerspruch: Es gibt keine solche Methode HALT

Einschränkungen für die Berechenbarkeit:

- Ist die Fragestellung formal nicht beschreibbar, kann keine Lösung berechnet werden.
- Es gibt Aufgabenstellungen, deren Berechnung mehr Zeit kostet, als aufgewendet werden kann.
- Es gibt formal beschreibbare Probleme, die prinzipiell nicht berechnet werden können. Berechenbar heißt, dass es einen Algorithmus gibt, der diese Funktion berechnet.