2. Aufgabenblatt

(Besprechung in den Tutorien 31.10.–04.11.2022)

Aufgabe 1. Berechenbar oder nicht?

Die unbewiesene Goldbachsche Vermutung lautet: „Jede gerade Zahl größer 2 lässt sich als Summe zweier Primzahlen darstellen."

1. Sei $f : \mathbb{N} \to \mathbb{N}$ eine Funktion, die bei allen Eingaben genau dann 1 ausgibt, wenn die Goldbachsche Vermutung gilt, und sonst 0. Existiert ein Algorithmus, der bei Eingabe $n \in \mathbb{N}$ nach endlicher Zeit $f(n)$ ausgibt?
2. Existiert ein Algorithmus, der bei Eingabe einer binär kodierten natürlichen Zahl n genau dann 1 ausgibt, wenn n eine gerade Zahl größer 2 ist und sich als Summe zweier Primzahlen darstellen lässt, und sonst 0?
3. Beschreiben Sie den sich ergebenden Unterschied, wenn folgende, gegenüber 2. modifizierte Aufgabe betrachtet wird: Bei Eingabe n ist genau dann 1 auszugeben, wenn n eine gerade Zahl größer 2 ist und sich als Differenz zweier Primzahlen darstellen lässt, und 0 sonst.

Aufgabe 2. Berechenbar oder nicht?

Geben Sie an, ob folgende Funktionen berechenbar sind. Begründen Sie Ihre Antworten.

1. $f(n) = \begin{cases} 1, & \text{falls } n \text{ Studierende unter den dann geltenden Abstandsregeln} \\ & \text{die Klausur im Audimax schreiben können.} \\ 0, & \text{sonst.} \end{cases}$

2. $g(n) = \begin{cases} 1, & \text{falls } n \text{ Tage nach dem 24.12.2022 die Sonne nicht scheint} \\ & \text{oder schönes Wetter ist.} \\ 0, & \text{sonst.} \end{cases}$

   (Anmerkung: Sonnenschein impliziert schönes Wetter.)

Aufgabe 3. Berechenbar oder nicht?

Im Folgenden sei $B \in \mathbb{N}$ „Die kleinste Zahl, die sich nicht mit weniger als zwanzig deutschsprachigen Wörtern definieren lässt." Desweiteren sei die Funktion $f : \mathbb{N} \to \mathbb{N}$ definiert als

$$f(n) := \begin{cases} 1, & n \leq B, \\ 0, & \text{sonst.} \end{cases}$$

Ist die Funktion f berechenbar oder nicht?

Aufgabe 4. Berechenbarkeit

Sei $f : \mathbb{N} \to \mathbb{N}$ die Funktion aus der Vorlesung (Beispiel 3, Folie 21) mit

$$f(n) := \begin{cases} 1, & \begin{array}{l}\text{falls die Dezimaldarstellung von } n \text{ in der Dezimalbruchentwicklung} \\ \text{von } \pi \text{ vorkommt}\end{array} \\ 0, & \text{sonst} \end{cases}.$$

Desweiteren definieren wir für $x \in \mathbb{N}$ die Funktion $f_x : \mathbb{N} \to \mathbb{N}$ wie folgt

$$f_x(n) := \begin{cases} 1, & \text{falls die Dezimalbruchentwicklung von } \pi \ n \text{ aufeinanderfolgende} \\ & \text{Konkatenationen der Dezimaldarstellung von } x \text{ enthält} \\ 0, & \text{sonst} \end{cases} .$$

Zum Beispiel gilt $f_{141}(1) = 1$. Die Funktion f_1 entspricht also der Funktion aus Beispiel 4 aus der Vorlesung (Folie 21).

Worin besteht das Problem in folgendem vermeintlichen „Beweis“ der Berechenbarkeit von f?

„Für jedes $x \in \mathbb{N}$ ist f_x berechenbar (analog zum Beweis der Berechenbarkeit von f_1 aus der Vorlesung). Um nun die Funktion $f(n)$ zu berechnen, kann ein Algorithmus also einfach den Wert von $f_n(1)$ berechnen und ausgeben. Also ist auch f berechenbar.“