

Aufgabe 1. Berechne die Summe der ersten n ungeraden Zahlen mit einer `while`-Schleife. Wie ist die Ausgabe für $n = 1, \dots, 15$. Was fällt dir auf? Könnte man diese Aufgabe nun also effizienter programmieren?

Aufgabe 2.

- Implementiere den Primzahltest (Algorithmus 1) von gestern.
- Schreibe ein Programm, das jeweils die nächste Primzahl nach 20000, 30000 und 40000 findet.

Aufgabe 3. Für $a \in \mathbb{R}^+$ konvergiert die Folge (a_n) mit $a_0 = a$ und

$$a_{n+1} = \frac{1}{2} \left(a_n + \frac{a}{a_n} \right)$$

gegen \sqrt{a} . Implementiere damit einen Wurzellalgorithmus.

Aufgabe 4. Implementiere den Algorithmus 2 von gestern, welcher den größten gemeinsamen Teiler zweier Zahlen berechnet.

Aufgabe 5. Implementiere den Cosinus über seine Reihendarstellung mit einer Schleife. Du kannst die Formel bei Wikipedia nachschlagen, sie selbst entwickeln oder diese hier verwenden:

$$\cos(x) = \sum_{k=0}^{\infty} (-1)^k \cdot \frac{x^{2k}}{(2k)!}$$

Aufgabe 6. Schreibe ein Programm, um den Wert der Reihe

$$\sum_{k=1}^{\infty} \frac{1}{k^2}$$

zu berechnen. Er sollte $\frac{\pi^2}{6}$ sein. Wichtig ist, sich ein geeignetes Abbruchkriterium zu überlegen.