

Aufgabe 1. Das folgende Programm sollte die Summe der ersten n Zahlen berechnen. Allerdings enthält es 4 Fehler. Finden Sie sie alle!

```
1  /* Summe der ersten n Zahlen. */
2
3  #include <iostream>
4
5  int main ()
6  {
7      int n = 20;      /* Addiere bis zu dieser Zahl */
8      int i;
9      int summe;      /* speichert Zwischenergebnis */
10
11     i = 0;
12
13     while (i < n)
14     {
15         summe += i    /* addiere ite Zahl auf summe */
16     }
17     std::cout << "Das Ergebnis ist " << summe << std::endl;
18     return 0;
19 }
```

Was fällt Ihnen auf, wenn Sie das Programm nach Korrektur ausführen? Könnte man diese Aufgabe nun also effizienter implementieren?

Aufgabe 2.

- a) Implementieren Sie den Primzahltest (Algorithmus 1) von gestern.
- b) Schreiben Sie 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} (warum?). Implementieren Sie damit einen Algorithmus, um die Quadratwurzel einer positiven Zahl auszurechnen (siehe Algorithmus 2 von gestern).

Aufgabe 4. Implementieren Sie den Algorithmus 3 von gestern, der den größten gemeinsamen Teiler zweier Zahlen berechnet.

Aufgabe 5. Schreiben Sie ein Programm, um den Wert der Reihe

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

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