

Programmiermethoden in der Mathematik WS 05/06
Woche 5 – Tutorium

Schreibe zu einer/einigen der folgenden Aufgaben

- zuerst einen Algorithmus auf (als Pseudocode, Struktogramm oder Flussdiagramm)
- und implementiere diesen dann in C++:

1. Analysis 1 Übungsblatt 1 Aufgabe 6: Binomialkoeffizient

Eingabe: $n, k \in \mathbb{N}, k \leq n$. Ausgabe: $\binom{n}{k}$. Benutze möglichst wenig Rechenoperationen.

2. Ü5 A20: Fibonacci-Zahlen: $a_1 = a_2 = 1, a_{n+1} = a_n + a_{n-1}, n \geq 2$.

Eingabe: $N \in \mathbb{N}, N \geq 1$.

Ausgabe: a_1 bis a_N , untereinander (oder besser in Reihen mit je fünf Werten) auf dem Bildschirm.

3. Ü5 A20: Seien a_n die Fibonacci-Zahlen und $b_n = \frac{a_{n+1}}{a_n}$.

Eingabe: $\varepsilon \in \mathbb{R}^+$.

Ausgabe: b_n und n , so dass $|b_n - b_{n-1}| < \varepsilon$.

4. Ü5 A22 (Algorithmus war in der VL): Sei $A_{MN} := \left\{ \frac{1}{2^m} + \frac{1}{n}, m \leq M, n \leq N \right\}$.

Eingabe: $M, N \in \mathbb{N}, M, N \geq 1$.

Ausgabe: $\min A_{MN}$ und $\max A_{MN}$.

5. Ü6 A23 (ähnlicher Algorithmus war in der VL): Die Reihe $\sum_{n=1}^{\infty} \frac{1}{n(n+1)(n+2)}$ konvergiert.

Eingabe: $\varepsilon \in \mathbb{R}^+$.

Ausgabe: Partialsumme $S_N = \sum_{n=1}^N \frac{1}{n(n+1)(n+2)}$ und N , so dass $|S_N - S_{N-1}| < \varepsilon$.

Modifikation: Versuche, die Werte $n, n+1, n+2$ nicht alle in jedem Schritt neu zu berechnen, sondern die entsprechenden vom letzten Schleifendurchlauf zu verwenden.

6. Zur Berechnung von π gibt es z.B. folgende Möglichkeiten:

$$\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \dots + \dots \quad (\text{Gregory 1638-1675})$$

$$\frac{\pi^2}{6} = \frac{1}{1^2} + \frac{1}{2^2} + \frac{1}{3^2} + \dots \quad (\text{Euler 1707-1855})$$

Berechne π mit einer Darstellung so genau wie möglich. Zum Vergleich s.

<http://pi314.at/math/100000digits.html>.

Mit der Anweisung `cout.precision(10);` vor einer `cout`-Anweisung erreicht man, dass insgesamt z.B. 10 Stellen (Vor- und Nachkommastellen ohne Zählen des Dezimalpunkts) ausgegeben werden.