

**Aufgabe 1.** Das folgende Programm sollte die Fakultätsfunktion implementieren und  $5! + 17!$  auf der Konsole ausgeben. Zufälligerweise haben wir 6 Fehler dabei gemacht. Schnapp sie dir alle!

```
1  /* Fakultaetstest
2   * (c) 2015 Clelia und Johannes */
3
4  #include <studio.h>
5
6  int fakultaet (n) {
7      int ergebnis = 0;      /* speichert die Fakultaet */
8
9      while (n > 0)          /* verkleinere n, bis es */
10         ergebnis *= n;     /* null ist und multi- */
11         n--;                /* pliziere mit ergebnis */
12     return ergebnis;
13 }
14
15 int main () {
16     int add2fak;
17
18     add2fak = fakultaet (5) + fakulataet (17);
19     printf ("5! + 17! = %i\n", add2fak);
20     return 0,
21 }
```

**Aufgabe 2.** Implementiere die Signumsfunktion  $\text{sgn}(x)$ , den Absolutbetrag  $\text{betrag}(x)$ ,  $\cos(x)$  und die Wurzelfunktion  $\text{wurzel}(x)$  (mit dem Heron-Verfahren vom ersten Zettel) als Funktionen und lagere sie in ein eigenes Modul aus.

**Aufgabe 3.** a) Implementiere für  $x \in \mathbb{R}$  und  $n \in \mathbb{N}$  eine Potenzfunktion  $x^n = \text{power}(x, n)$  mit der Double-and-Add-Methode:

$$\text{power}(x, n) = \begin{cases} 1 & \text{wenn } n = 0 \\ x \cdot \text{power}\left(x^2, \frac{n-1}{2}\right) & \text{wenn } n \text{ ungerade} \\ \text{power}\left(x^2, \frac{n}{2}\right) & \text{wenn } n \text{ gerade} \end{cases}$$

zuerst mal rekursiv.

---

- b) Implementiere eine Potenzfunktion `naiv_power(x, n)`, indem du eine Schleife von 1 bis  $n$  laufen lässt und bei jedem Durchlauf eine mit 1 Initialisierte Variable mit  $x$  multipliziert. Berechne  $0,999999999^{2000000000}$  einmal mit `power(x, n)` von oben und einmal mit `naiv_power(x, n)` (es sollte ca. 0,818731 raus kommen).
- c) \* Implementiere die Double-and-Add-Methode mit einer Schleife, also ohne rekursiven Aufruf.
- d) \* Frage einen Tutor wie man Zeit messen kann und vergleiche die Laufzeiten der 3 Funktionen.

**Aufgabe 4.** Diese Aufgabe wird auf eine `power(x, y)`-Funktion führen, die für beliebige  $x \in \mathbb{R}^+$  und  $y \in \mathbb{R}$  den Wert von  $x^y$  berechnet.

- Implementiere die Exponential-Funktion `expo(x)`, die  $e^x$  mithilfe folgender Reihendarstellung:

$$e^x = \sum_{k=0}^{\infty} \frac{x^k}{k!}$$

- Implementiere eine Logarithmus-Funktion `logarithm(x)`, die  $\ln(x)$  mithilfe folgender Reihendarstellung berechnet:

$$\ln(x) = 2 \cdot \sum_{k=0}^{\infty} \left( \frac{x-1}{x+1} \right)^{2k+1} \frac{1}{2k+1}$$

- Verwende die Formel

$$x^y = e^{y \cdot \ln(x)}$$

um `power(x, y)` zu bestimmen.

**Aufgabe 5.** Implementiere die Riemann'sche Zeta-Funktion für  $s \in \mathbb{R}$ :

$$\zeta(s) := \sum_{k=1}^{\infty} \frac{1}{k^s}$$

Teste die Funktion für einige Werte  $s \in ]1, 3[$ . Für Werte  $s \leq 1$  gilt  $\zeta(s) = \infty$ .

---