

Praktikum Programmieren C/C++ Blatt 2

Aufgabe 1: Arithmetisches Mittel

Schreiben Sie unter Verwendung einer `while`-Schleife ein Programm, das Zahlen von der Tastatur mit Hilfe der Anweisung

```
scanf("%d", &eingabe);
```

von der Tastatur einliest und in der `int` Variable `eingabe` speichert. In der `while`-Schleife sollen die Zahlenwerte aufsummiert werden. Die Schleife soll mit der Eingabe von 0 beendet werden. Dann soll die Summe und der Mittelwert der eingegebenen Zahlen auf dem Bildschirm ausgegeben werden.

Aufgabe 2: Datentyp `bool`

In C gibt es keinen fest eingebauten Datentyp für Boolesche Werte. Definieren Sie über `#define` Anweisungen einen Datentyp `bool` und die Konstanten `true` und `false`. Verwenden Sie für alle weiteren Aufgaben diesen Datentyp.

Aufgabe 3: Logik

- a) Implementieren Sie die Boolesche Funktion „Implikation“ unter Verwendung der vorhandenen Booleschen Funktionen „oder“ und „nicht“. Wenden Sie die Funktion im Hauptprogramm auf zwei über `scanf` eingelesene Werte an und geben Sie das Ergebnis aus.
- b) Implementieren Sie basierend auf der Funktion Implikation die Funktion „Äquivalenz“ (beidseitige Implikation, Biimplikation). Wenden Sie die Funktion im Hauptprogramm auf zwei über `scanf` eingelesene Werte an und geben Sie das Ergebnis aus.
- c) Geben Sie eine vollständige Wahrheitstabelle für beide Funktionen aus.

Aufgabe 4: Schaltjahr

Schaltjahre sind die Jahre, die durch 4 teilbar sind, außer den Jahren die durch 100 aber nicht durch 400 teilbar sind. Es sei n eine natürliche Zahl (eine Jahreszahl) und A,B,C,S die Aussagen:

- A: n ist durch 4 teilbar.
- B: n ist durch 100 teilbar.

C: n ist durch 400 teilbar.
Dann kann man auf zwei Weisen bestimmen ob n ein Schaltjahr ist:

1. $A \ \&\& \ !(B \ \&\& \ !C)$
2. $(C \ || \ !B) \ \&\& \ A$

Verwenden Sie die beiden Varianten um zweimal die Funktion

```
bool istSchaltjahr(int n);
```

zu implementieren (nennen Sie die Funktion einmal `istSchaltjahr1` und einmal `istSchaltjahr2`) und vergleichen Sie die Laufzeit der beiden Funktionen miteinander, indem Sie ausrechnen, wie viele Schaltjahre es zwischen den Jahren 1 und 1000000000 gibt.

Zum Abfrage der Systemzeit benötigt man eine Bibliotheksfunktion, die in `time.h` deklariert ist (verwenden Sie also `#include <time.h>`). Definieren Sie sich für die Zeitmessung folgende Variablen:

```
time_t startzeit, endzeit;
```

Mit den Anweisungen

```
time(&startzeit);  
time(&endzeit);
```

wird die aktuelle Zeit in die Variable `startzeit` bzw. `endzeit` geschrieben. Der Ausdruck `difftime(endzeit, startzeit)` berechnet die Differenz der beiden Zeiten und liefert einen Wert vom Typ `double` zurück, der die verstrichene Zeit in Sekunden angibt. Ihr Programm sollte sowohl die Anzahl der Schaltjahre ausgeben als auch die verstrichene Zeit.

Aufgabe 5: Fakultät

In Wikipedia ist die Fakultätsfunktion wie folgt definiert:

$$n! := \begin{cases} n \cdot (n-1)! & n > 0 \\ 1 & n = 0 \end{cases}.$$

Beispiel:

$$1! = 1$$

$$2! = 1 \cdot 2 = 2$$

$$3! = 1 \cdot 2 \cdot 3 = 6$$

$$4! = 1 \cdot 2 \cdot 3 \cdot 4 = 24$$

$$5! = 1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 = 120$$

Implementieren Sie eine **rekursive** Funktion Fakultät:

```
int fak(int x)
```

Implementieren Sie zwei Varianten dieser Funktion, indem Sie einmal einen **bedingten Ausdruck** und einmal eine **bedingte Anweisung** verwenden.

Schreiben Sie ein Hauptprogramm, welches mit `scanf` eine positive ganze Zahl `n` einliest und den Wert von Fakultät(`i`), $0 \leq i \leq n$ berechnet und auf der Konsole ausgibt.