

Pflichtaufgaben:

Aufgabe 1: Lokale Arrays

- Erstellen Sie eine Funktion, die ein `int[10]` erstellt, mit den Zahlen 1-10 befüllt und anschließend zurückgibt. Was müssen Sie bei der Rückgabe beachten?
- Führen Sie die Funktion in `main` aus und prüfen Sie die Array-Elemente auf Korrektheit. Wie erklären Sie sich das Ergebnis? Wie müssten Sie die Funktion verändern, damit das Problem nicht auftritt?

Aufgabe 2: `int` to `char*`

Erstellen Sie eine Funktion `char* toBinaryString(int)`, die einen Integer binär in eine Zeichenkette schreibt und zurückgibt. Führende nullen sollen nicht beachtet werden.

Bsp.: `toBinaryString(5) = "101"`

Aufgabe 3: Matrizenmultiplikation

Erstellen Sie eine Funktion `multiply(int* A, int* B, int* C, int n)`, die zwei quadratische Matrizen *A* und *B* der Länge *n* miteinander multipliziert und das Ergebnis in *C* speichert.

Erinnerung:
$$c_{ik} = \sum_{j=1}^m a_{ij} \cdot b_{jk}$$

Empfohlene Aufgabe:

Aufgabe 4: Buchstaben zählen

Schreiben Sie ein Programm, das die Buchstaben eines beliebig langen Texts zählt und ausgibt. Der Text soll beim Aufruf des Programms, ohne Anführungszeichen, als Parameter angegeben werden (`char **argv`). Bedenken Sie, dass der Text durch die fehlenden Anführungszeichen möglicherweise über mehrere Parameter verteilt ist.

Es sollen nur Kleinbuchstaben und Zahlen berücksichtigt werden. Alle anderen Zeichen sollen als "Unbekannt" gezählt werden.

Bsp.:

```
>charcounter.exe Hello World!
```

d: 1

e: 1

l: 3

o: 2

r: 1

Unbekannt: 3