

Programmierung 1

Übungsblatt Woche 6 - 01. & 02. Dezember 2022

Portfoliorelevant!

1. Spannung, Strom und Widerstand

Implementieren Sie mit Hilfe von Funktionen das in den Folien vorgestellte C-Programm zur Berechnung der Spannung. Verwenden Sie den Code von Folie 9 und 13.

Erweitern Sie ihr Programm, d.h. die main-Funktion, um eine Schleife, in der der Benutzer gefragt wird, ob er Spannung U , Strom I oder Widerstand R berechnen möchte. Davon abhängig soll entweder U , I oder R mit Hilfe geeigneter Funktionen (wie im ersten Teil bereits für die Spannung umgesetzt) berechnet werden.

2. Gewinn

Schreiben Sie eine Funktion `int gewinn (int a, int b, int c)`. Wenn alle Argumente jeweils voneinander verschieden sind, soll 0 zurückgegeben werden. Wenn alle Zahlen gleich sind, soll 20 zurückgegeben werden. Wenn nur zwei Zahlen gleich sind, soll 10 zurückgegeben werden.

3. Runden

Schreiben Sie eine Funktion `int rundeSumme (int a, int b, int c)`, die die Summe der auf Zehner gerundeten Werte von a , b und c zurückgibt. Schreiben Sie hierzu eine Hilfsfunktion `int runde10 (int n)`, die den formalen Parameter auf den nächsten Zehner aufrundet (ab Endziffer 5) oder abrundet (bis Endziffer 4), sodass Code nicht unnötig wiederholt wird.

4. Teiler

Implementieren Sie eine Funktion `int ggT (int a, int b)` zur Bestimmung des größten gemeinsamen Teilers (ggT) mit Hilfe des euklidischen Algorithmus. Dieser funktioniert wie folgt: Gegeben seien zwei natürliche Zahlen a und b . Man zieht nun solange die kleinere von der größeren Zahl ab, bis beide Zahlen gleich sind. Das Resultat $ggT(a, b)$ ist der größte gemeinsame Teiler von a und b .

5. Vielfaches

Gegeben seien wieder zwei natürliche Zahlen a und b . Gesucht ist diesmal das kleinste gemeinsame Vielfache (kgV), also die kleinste Zahl, die sowohl a als auch b als Teiler hat. Im Gegensatz zum ggT gibt es für den kgV leider keine so einfache Berechnungsvorschrift wie den euklidischen Algorithmus. Da wir seit Aufgabe 4) jedoch schon den größten gemeinsamen Teiler $ggT(a, b)$ ausrechnen können, können wir aber zur Berechnung des kgV die folgende Beziehung nutzen: $a \cdot b = ggT(a, b) \cdot kgV(a, b)$.

Zu implementieren ist nun also die Funktion `int kgV (int a, int b)`, welche zur Bestimmung des kgV zweier Zahlen a und b deren ggT verwendet.

6. Mauer

Implementieren Sie eine Funktion `bool mauern (int klein, int gross, int ziel)`. Wir wollen eine Mauer der Länge `ziel` mauern. Dazu haben wir Anzahl `klein` Steine der Länge 1 und `gross` Steine der Länge 5. Die Funktion soll `true` zurückliefern, wenn das Ziel erreicht werden kann, andernfalls `false`.

Beispiele:

- `mauern(3, 1, 8) → true`
- `mauern(3, 1, 9) → false`
- `mauern(3, 2, 10) → true`