

Programmieren C: Mehr Schleifen, Restrechnung; Primfaktorenzerlegung

Klaus Kusche

Schreib ein Programm, das mit einer Zahl n auf der Befehlszeile aufgerufen wird und deren Primfaktoren ausgibt, d.h. jene Zahlen, die n glatt teilen, selbst nicht teilbar sind, und zusammenmultipliziert n ergeben.

Ignoriere das Vorzeichen der Eingabe und gib bei 0 und 1 die Zahl selbst aus.

Für dieses Problem gibt es viele verschiedene richtige Lösungsansätze, u.a. folgende:

Variante 1:

- Teste der Reihe nach alle Zahlen ab 2, solange n noch größer als 1 ist.
 - Solange die aktuelle Zahl n teilt, gib sie aus und dividiere sie aus n weg.

Für diese Idee brauchst du offensichtlich zwei Schleifen (aber kein if). Denk nach: Wie werden diese beiden Schleifen richtig angeordnet? Nacheinander oder “ineinander”?

Variante 2:

- Beginne mit 2 als zu probierende Zahl und wiederhole Folgendes so lange, bis n gleich 1 ist:
 - Wenn die aktuelle Zahl n teilt, gib sie aus, dividiere sie aus n weg, und probiere es im nächsten Umlauf mit derselben Zahl noch einmal.
 - Sonst gib nichts aus und probiere es im nächsten Umlauf mit der nächsten Zahl.

Für diese Idee sollte eine Schleife und ein **if** reichen!

Zusatzaufgabe:

- Kannst du das Ergebnis mit Potenzen ausgeben, also z.B. bei 24 statt “2 2 2 3” schön “ $2^3 3$ ”? (“ 1 ” solltest du weglassen!)

Dafür darfst du die Zahl nicht gleich ausgeben, wenn sie n teilt, sondern musst zuerst einmal nur mitzählen, wie oft du sie aus n wegdividieren kannst.

Erst wenn die Zahl n nicht mehr teilt, kommt die Ausgabe:

- Hat sie null Mal geteilt, wird nichts ausgegeben.
- Hat sie ein Mal geteilt, wird nur die Zahl ausgegeben.
- Hat sie öfter geteilt, wird die Zahl, das Hochzeichen und die Anzahl ausgegeben.

Die Zusatzaufgabe geht mit Lösungsvariante 1 viel leichter, sie bei Variante 2 dazuzubauen, ist ziemlich kompliziert!