

## Algorithmen und Programmierung

# Einfache Rekursionen (Trainingsaufgabe)

Rekursion stellt sich sehr oft als eine der größten Hürden heraus. Daher sollen Sie in dieser Trainingsaufgabe einige einfache rekursive Funktionen implementieren.

Die Grundidee ist dabei, ein Problem auf eine *einfachere Version von sich selbst* zurückzuführen. Was das genau heißt, unterscheidet jedoch sich von Problem zu Problem. Dieses Vorgehen soll solange wiederholt werden, bis ein meist trivialer Grundfall des Problems erreicht werden kann. Wie Sie wahrscheinlich wissen, haben wir bereits mit rekursiven Algorithmen gearbeitet, beispielsweise dem euklidischen Algorithmus zur Berechnung des größten gemeinsamen Teilers zweier Zahlen. Durch die Anwendung dieses Algorithmus haben sich beide Zahlen solange angenähert, bis sie schließlich gleich waren. „Einfacher“ heißt in diesem Sinn also nicht unbedingt einfacher zu lösen, aber *näher am Ziel*.

**Aufgaben:** Schreiben Sie eine Quelltextdatei `recursions.c`, welche die folgenden Funktionen rein rekursiv (ohne Verwendung von Schleifen) implementieren soll:

- Zählen von  $k$  bis 0  
`void countDown(unsigned int k);`
- Zählen von 0 bis  $k$   
`void countUp(unsigned int k)`
- Zählen von  $k$  bis 0 und wieder bis  $k$   
`void countDownUp(unsigned int k)`

*Beispiele:*

- `countDown(9)`  $\Rightarrow$  9876543210
- `countUp(3)`  $\Rightarrow$  0123
- `countDownUp(4)`  $\Rightarrow$  432101234

Diese Datei soll zur Abgabe **keine** `main`-Funktion beinhalten. Achten Sie also darauf, diese nach dem Testen auszukommentieren oder zu entfernen.

Sie können zur Umsetzung auch weitere Hilfsfunktionen nutzen, beispielsweise um weitere Funktionsparameter zum Zählen zu verwenden. Dies ist für diese Funktionen jedoch nicht notwendig.

## Hinweise zur Aufgabenstellung

Für die Lösung dieser Aufgabe benötigen Sie folgende Grundkenntnisse:

- Kontrollfluss in C (`if`)
- Rekursion
- Benutzung von `gcc`

## Hinweise zur Abgabe

- Erstellen Sie eine ZIP- bzw. TGZ-Archivdatei, welche die geforderten Dateien enthält.
- Fügen Sie dem Archiv keine weiteren Dateien oder Ordner hinzu.
- Reichen Sie Ihre Lösung unter <https://osg.informatik.tu-chemnitz.de/submit> ein.
- Bis zum Abgabeende (Deadline), sofern gegeben, können beliebig neue Lösungen eingereicht werden, die die jeweils älteren Versionen ersetzen.
- Ihr Programm muss auf der Testmaschine übersetzbar sein. Deren Details sind auf dem OpenSubmit-Dashboard verfügbar.
- Ihre Lösung wird automatisch validiert. Sie werden über den Abschluss der Validierung per eMail informiert.