

Algorithmische Mathematik I

3. Übung

1. (a) Ersetzen Sie im Programm 1.25 (Collatz-Folge) aus der Vorlesung in Zeile 26 die Anweisung „ $n = 3 * n + 1;$ “ durch „ $n = n + 1;$ “.
Zeigen Sie, dass das Programm dann stets terminiert, und geben Sie (mit Hilfe der O -Notation) eine möglichst gute Schranke für die Zahl der Rechenschritte an.
- (b) Was passiert, wenn Sie die obige Anweisung durch „ $n = n + 3;$ “ ersetzen? Für welche Startwerte terminiert das Verfahren dann? Beweisen Sie die Korrektheit Ihrer Antwort. (3+2 Punkte)
2. Die Existenz im Beweis von Satz 2.1 kann auch aus Lemma 1.19 gefolgert werden. Wie? (5 Punkte)
3. Geben Sie einen Algorithmus in Pseudocode an, der zu gegebenen natürlichen Zahlen a und b die Zahl a^b berechnet, und der nur $O(\log_2 b)$ elementare Rechenoperationen ($+$, $-$, $*$, $/$, $\%$) durchführt. Beweisen Sie, dass Ihr Algorithmus tatsächlich diese Eigenschaften hat. (6 Punkte)
4. In dieser Aufgabe betrachten wir Komplementdarstellungen zur Basis 2.
 - (a) Schreiben Sie die Zahl -25 in Komplementdarstellung mit 8 Bits und mit 16 Bits.
 - (b) Sei z die Komplementdarstellung einer negativen Zahl mit l Bits. Welche Zahl entsteht, wenn man in z jede 0 durch eine 1 und jede 1 durch eine 0 ersetzt?
 - (c) Für welche negativen Zahlen x ist die Komplementdarstellung mit l Bits bis auf die Vorzeichenstelle identisch mit der Komplementdarstellung von $-x$?
 - (d) Sei z die Komplementdarstellung einer negativen Zahl mit l Bits. Wie sieht die Darstellung derselben Zahl aus, wenn $2l$ Stellen für die Komplementdarstellung zur Verfügung stehen?
Bemerkung: Eine solche Umwandlung kann zum Beispiel notwendig werden, wenn eine Variable vom Typ `short` in eine Variable vom Typ `long` umgewandelt wird. (1+1+1+1 Punkte)

Öffnungszeiten des Help Desks: Dienstags, 13 – 16 Uhr und donnerstags, 10 – 13 Uhr, jeweils in Raum N1.002, Endenicher Allee 60, Nebengebäude.

Abgabe: Montag, den 7.11.2016, vor der Vorlesung.