

Algorithmische Mathematik I

3. Übung

1. Ersetzen Sie bei der Berechnung der Collatz-Folge die Anweisung $n = 3 * n + 1$; durch
 - (a) $n = n + 1$; . 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) $n = n + 3$; . Für welche Startwerte terminiert das Verfahren dann? Beweisen Sie die Korrektheit Ihrer Antwort. (3+2 Punkte)

2. Es sei z die 2-Komplementdarstellung einer negativen Zahl mit l Bits.
 - (a) Wie sieht die Darstellung derselben Zahl aus, wenn $2l$ Stellen für die Komplementdarstellung zur Verfügung stehen?
 - (b) Welche Zahl entsteht, wenn man in z jede 0 durch eine 1 und jede 1 durch eine 0 ersetzt? (1+2 Punkte)

Bemerkung zu Teil (a): Eine solche Umwandlung kann zum Beispiel notwendig werden, wenn eine Variable vom Typ `short` in eine Variable vom Typ `long` umgewandelt wird.

3. Es sei $(a_{l-1}a_{l-2} \dots a_0)_{-10} := \sum_{i=0}^{l-1} a_i(-10)^i$, wobei $a_i \in \{0, \dots, 9\}$ sei für $i \in \{0, \dots, l-1\}$. Man nennt $a_{l-1} \dots a_0$ Darstellung von $\sum_{i=0}^{l-1} a_i(-10)^i$ zur Basis -10 , falls $a_{l-1} \neq 0$ gilt oder $l = 1$ und $a_0 = 0$ gelten.
 - (a) Schreiben Sie $(19375573910)_{-10}$ als Dezimalzahl.
 - (b) Geben Sie eine Darstellung von $(9230753)_{10}$ zur Basis -10 an.
 - (c) Zeigen Sie, dass es für jede ganze Zahl x eine Darstellung zur Basis -10 gibt.
 - (d) Ist die Darstellung aus Aufgabenteil (c) immer eindeutig? (1+1+2+3 Punkte)

4. Schreiben Sie ein C++-Programm, das für jede Zahl $n \in \{1, \dots, 1000000\}$ die Länge der in n beginnenden Collatz-Folge berechnet und anschließend ein $n^* \in \{1, \dots, 1000000\}$ ausgibt, für das die in n^* beginnende Collatz-Folge am längsten ist. Geben Sie auch die Länge der in n^* beginnenden Collatz-Folge an. Achten Sie darauf, mögliche Überläufe durch geeignete Abfragen abzufangen. (5 Punkte)

Abgabe: Montag, den 29.10.2018, bis 10:12 Uhr.

Abgabe der Programmierübungen:

Per E-Mail an `alma_prog_gr_XX@dm.uni-bonn.de`, wobei XX durch die Nummer Ihrer Übungsgruppe zu ersetzen ist, also z.B. `alma_prog_gr_07@dm.uni-bonn.de`, wenn Sie in Gruppe 7 sind, oder `alma_prog_gr_12@dm.uni-bonn.de`, wenn Sie in Gruppe 12 sind. Wenn Sie Ihre Übungsgruppe nicht kennen, schreiben Sie an `alma_prog_gr_unbekannt@dm.uni-bonn.de`.

Öffnungszeiten des Help Desks:

Montags, 16 – 19 Uhr und freitags, 12 – 15 Uhr, jeweils in Raum N1.002, Endenicher Allee 60, Nebengebäude.

www.mathematics.uni-bonn.de/files/bachelor/help-desk.pdf

Genauere Zeiten der Programmierübungen:

Gruppe 1:	Mo	14:30 - 16:00	Wegelerstr. 6
Gruppe 2:	Mo	12:30 - 14:00	Endenicher Allee 60
Gruppe 4:	Di	10:30 - 12:00	Endenicher Allee 60
Gruppe 5:	Di	12:30 - 14:00	Wegelerstr. 6
Gruppe 6:	Di	14:00 - 15:30	Wegelerstr. 6
Gruppe 7:	Di	16:00 - 17:30	Endenicher Allee 60
Gruppe 8:	Do	14:00 - 15:30	Wegelerstr. 6
Gruppe 9:	Do	16:00 - 17:30	Endenicher Allee 60
Gruppe 10:	Mi	12:30 - 14:00	Endenicher Allee 60
Gruppe 11:	Mi	14:30 - 16:00	Wegelerstr. 6
Gruppe 12:	Do	10:30 - 12:00	Endenicher Allee 60
Gruppe 13:	Do	12:30 - 14:00	Wegelerstr. 6
Gruppe 14:	Mi	18:00 - 19:30	Endenicher Allee 60

Die Programmierübungen fangen also nicht wie sonst üblich “c.t.”, also eine Viertelstunde nach der vollen Stunde, sondern entweder genau zur vollen Stunden oder eine halbe Stunde danach an. Die Anfangszeiten der Theorieübungen sind stets “c.t.” gemeint.