

Übungen zu Algorithmen

Wintersemester 2003/2004

Blatt 4

Aufgabe 4.1 (30 Punkte)

Schreiben Sie ein Programm, das eine Zeichenkette einliest und dann feststellt, ob die Eingabe ein Palindrom war.

Def.: Ein Palindrom ist eine Zeichenkette, die von vorne und von hinten gelesen gleich lautet.

Bsp.: Otto, MAOAM, einnegermitgazellezagtimmerennie

Dabei kommt es nicht auf die Groß- und Kleinschreibung an. (MAOam **ist** ein Palindrom). Ihr Programm soll eine entsprechende Meldung ausgeben, falls kein Zeichen eingegeben worden ist.

Aufgabe 4.2 (35 Punkte)

Der Märchenkönig Dodon hat einen Kerker mit 100 Zellen und je einem Gefangenen darin in seiner Burg. Alle Zellen lassen sich mit demselben Schlüssel öffnen. Wird der Schlüssel einmal im Schloß herumgedreht, ist die Tür offen. Dreht man ihn nocheinmal herum ist die Tür wieder verschlossen usw.

Am Abend vor seinem 50. Geburtstag läßt Dodon von einem Boten alle Zellentüren öffnen, denn er will den Gefangenen die Freiheit schenken. Die Wachen sorgen dafür, daß kein Gefangener vor 0:00 Uhr seine Zelle verläßt.

Dodon entscheidet sich anders und entsendet einen zweiten Boten, der den Schlüssel in jeder zweiten Zellentür – beginnend bei der zweiten – nocheinmal herumdreht. Danach schickt er einen dritten Boten, der den Schlüssel in jedem dritten Schloß – beginnend beim dritten – umdreht, ... , dann einen k -ten, der den Schlüssel in jedem k -ten Schloß – beginnend beim k -ten – herumdreht usw. Um 0:00 Uhr dreht der 100. Bote den Schlüssel in nur einer Tür – der 100. – herum und alle Gefangenen, deren Zellentüren unverschlossen sind, dürfen gehen.

Welche Gefangenen sind es und warum bleiben gerade diese Zellen offen?

Schreiben Sie ein Java-Programm, das für eine variable Anzahl $n > 0$ von Zellen das Schlüsseldrehen der n Boten durchführt und die Nummern der am Ende unverschlossenen Zellen ausgibt. Verwenden Sie ein Array von `boolean`-Werten.

Aufgabe 4.3 (35 Punkte)

Ein magisches Quadrat ist eine quadratische Anordnung natürlicher Zahlen, wobei die Summe der Zahlen in jeder Zeile und in jeder Spalte gleich ist.

Beispiel für $n = 3$:

6	1	8
2	9	4
7	5	3

Füllen Sie das Quadrat, indem Sie die Zahl 1 in die Mitte der obersten Reihe setzen. Setzen Sie die Zahl $i + 1$ eine Zeile über die Zahl i , wenn i durch n teilbar ist. Sonst setzen Sie $i + 1$ eine Zeile unter i in die Spalte links von i . Falls Sie dabei das Quadrat verlassen, betreten Sie es auf der gegenüberliegenden Seite wieder (s. Beispiel).

Schreiben Sie ein Java-Programm, das eine ungerade ganze positive Zahl $n \leq 19$ einliest und dann ein magisches Quadrat der Größe n berechnet und ausgibt, das mit den positiven ganzen Zahlen von 1 bis n^2 gefüllt ist.

Aufgabe 4.4 (0 Punkte)

Machen Sie sich mit folgenden UNIX-Kommandos vertraut:

Kommando	Erklärung
<code>java Aufg4.1 > dat.txt</code>	Umlenken der Ausgabe des Programms <code>Aufg4.1</code> in die <code>dat.txt</code> .
<code>dos2unix p.java</code>	Überführt die Textdatei <code>p.java</code> vom DOS- ins UNIX-Format.
<code>unix2dos q.java</code>	Überführt die Textdatei <code>q.java</code> vom UNIX- ins DOS-Format.
<code>du</code>	Gibt den Speicherplatz für den aktuellen Katalog in KByte aus.
<code>rm -r Blatt2</code>	Löscht das Verzeichnis <code>Blatt2</code> samt Inhalt.

Kombinierte vi-Kommandos

Durch Voranstellen eines `d` (*delete*) oder `c` (*change*) vor ein Positionierungskommando ergeben sich viele verschiedene Kommandos zum Löschen und Verändern von Text im *vi*. Außerdem können viele *vi*-Kommandos durch Eingabe einer Zahl direkt vor dem Kommando mehrfach ausgeführt werden.

Mode	Kommando	Erklärung
K	<code>3x</code>	Löscht drei Zeichen ab der aktuellen Position.
K	<code>12~</code>	Verwandelt 12 Klein- in Großbuchstaben und umgekehrt.
K	<code>5dd</code>	Löscht fünf Zeilen ab der aktuellen.
K	<code>21>></code>	21 Zeilen ab der aktuellen um ein <i>tab</i> nach rechts shiften.
K	<code>4rx</code>	Vier Zeichen ab der aktuellen Position durch vier <i>x</i> ersetzen.
K	<code>4s</code>	Vier Zeichen ab der aktuellen Position ersetzen (\rightarrow I-Mode).
K	<code>8Y</code>	Speichert acht Zeilen ab der aktuellen im Puffer.
K	<code>1G</code>	Setzt den Cursor an den Anfang von Zeile 1.
K	<code>3J</code>	3 Zeilen vereinen.
K	<code>5itextESC</code>	Fügt 5-mal <i>text</i> ein.
K	<code>d\$</code>	Von der aktuellen Position bis zum Zeilenende alles löschen.
K	<code>dG</code>	Von der aktuellen Position bis zum Dateiende alles löschen.
K	<code>dfx</code>	Von der aktuellen Position bis zum nächsten <i>x</i> rechts inkl. alles löschen.
K	<code>4dFx</code>	Von der aktuellen Position bis zum 4. <i>x</i> links inkl. alles löschen.
K	<code>3dw</code>	Von der aktuellen Position bis zum Ende des 3. Wortes alles löschen.
K	<code>cc</code>	Ersetzt die ganze Zeile (\rightarrow I-Mode).
K	<code>cw</code>	Von der aktuellen Position bis zum Wortende alles ersetzen (\rightarrow I-Mode).
K	<code>cb</code>	Vom Wortanfang bis zur aktuellen Position alles ersetzen (\rightarrow I-Mode).
K	<code>c\$</code>	Von der aktuellen Position bis zum Zeilenende alles ersetzen (\rightarrow I-Mode).
K	<code>c0</code>	Vom Zeilenanfang bis zur aktuellen Position alles ersetzen (\rightarrow I-Mode).
K	<code>ctx</code>	Von der akt. Pos. bis vor das nächste <i>x</i> rechts alles ersetzen (\rightarrow I-Mode).