

## 11. Übung

# Algorithmen & Programmierung

J. Akhundov

J. Pönisch

M. Reißner

M. Richter

### Aufgabe 1

Einem Programm werden nacheinander  $n$  Zahlen aus  $\{1, 2, 3, \dots, 2n\}$  als Eingabe präsentiert. Die Eingabe wird durch 0 abgeschlossen. Implementieren Sie einen Algorithmus für dieses Programm, der alle Zahlen ausgibt, die nicht eingegeben wurden. Sie können im Rahmen der Implementierung davon ausgehen, dass  $n < 100$  gilt. Zum Einlesen der Zahlen können Sie wieder die Funktion `zahl()` aus den vergangenen Übungen verwenden. Zur Ausgabe ist `printf()` zulässig.

### Aufgabe 2

Eine korrekte Raumbezeichnung in der TU Chemnitz folgt folgenden Regeln

- Die Bezeichnung beginnt mit einer Ziffer zwischen 1 und 4.
- Dann folgt ein Schrägstrich (/).
- Nach dem Schrägstrich kann (muss aber nicht) der Gebäudeteil durch einen Großbuchstaben angegeben werden. Ein Gebäudeteil kann zusätzlich durch einen anschließenden Kleinbuchstaben ergänzt werden.
- Anschließend folgt die maximal dreistellige Raumnummer. Hierbei sind Vornulln zulässig.
- Raumteile werden ggf. durch einen angehängten Kleinbuchstaben gekennzeichnet.
- Anstatt eines Gebäudeteils mit Raumnummer kann auch ein einzelnes Wort als Raumangabe verwendet werden. Das Wort beginnt mit einem Großbuchstaben und es folgen mindestens zwei Kleinbuchstaben.

**Korrekte** Raumbezeichnungen sind beispielsweise:

1/201, 2/N115, 2/N012a, 1/B208, 1/B006, 2/Eb4, 1/336b, 3/Aula, 2/Senatssaal, 1/Heizhaus

Inkorrekte Raumbezeichnungen sind z.B.:

201	↪ kein Uniteil angegeben
2/n115	↪ Kleinbuchstabe für Gebäudeteil
3/aula	↪ Raum beginnt mit Kleinbuchstaben
1/208A	↪ Großbuchstabe als Raumteil
1/0316	↪ mehr als drei Ziffern

- a) Zeichnen Sie einen Automaten, der als Eingabe eine Zeichenkette mit dem abschließenden Symbol EOT erhält, und der zwei Terminalzustände hat: einen, den der Automat erreicht, wenn die Zeichenkette eine **korrekte** Raumnummer darstellt, und den einen, der im anderen Fall erreicht wird.
- b) Geben Sie zusätzlich das entsprechende Syntaxdiagramm an.

### Aufgabe 3

In einem zweidimensionalen Feld sei eine  $m \times n$ -Matrix gespeichert. Die Elemente der Matrix haben folgende Eigenschaft

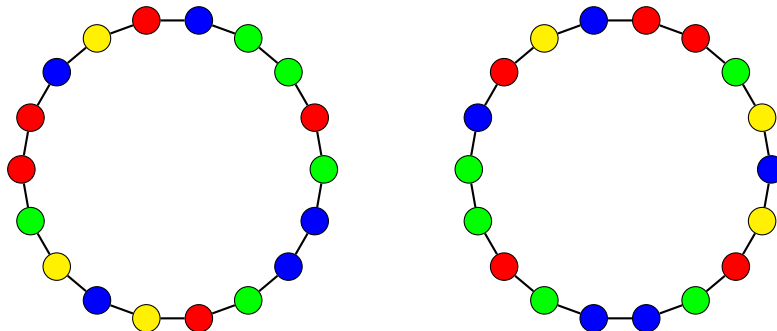
$$\begin{aligned}
 a_{i,j} &\leq a_{i,j+1} & \forall i = 1, \dots, n & \quad \forall j = 1, \dots, m-1 \\
 a_{i,j} &\leq a_{i+1,j} & \forall i = 1, \dots, n-1 & \quad \forall j = 1, \dots, m
 \end{aligned}$$

M.a.W.: die Zeilen und Spalten sind aufsteigend sortiert.

Entwerfen Sie einen Algorithmus zur Bestimmung der Position eines gesuchten Elements  $x$ . Sie können davon ausgehen, dass das Element  $x$  in der Matrix enthalten ist. Der Algorithmus soll bezüglich der Anzahl der Vergleiche möglichst effizient sein. Nutzen Sie dazu die Eigenschaft der Matrix aus. Geben Sie für Ihren Algorithmus an, wie viele Vergleiche im worst case notwendig sind. Wo steht dann das gesuchte Element? Formulieren Sie ihren Algorithmus als Funktion.

### Aufgabe 4

Wir betrachten Ketten, die aus verschiedenfarbigen Perlen bestehen. Folgende Perlenketten sind z.B. identisch.



- a) Geben Sie eine Datenstruktur zur Beschreibung einer Perlenkette an. Zeichnen Sie das Speicherlayout ihrer Datenstruktur auf.

- b) Implementieren Sie einen Algorithmus, der zwei Perlenketten  $K_1$  und  $K_2$  auf Gleichheit prüft, als Funktion in Python (Wirtschaftsinformatiker) bzw. C (alle anderen). Die Funktion soll beide Ketten als Parameter erhalten und einen Wahrheitswert („wahr“ bei Gleichheit, sonst „unwahr“) zurückgeben.

(Aufgabe aus dem Bundeswettbewerb Informatik 2007/08, 1. Runde)