



Aufgaben zum Selbstüben zur 5. Übung

Algorithmen & Programmierung

J. Akhundov

J. Pönisch

M. Reißner

M. Richter

Für alle Aufgaben kann zur Ausgabe auf die Funktion `printf()` der Standardbibliothek zurückgegriffen werden. Im Rahmen der Aufgaben müssen Ganzzahlen von der Eingabe eingelesen werden. Wann immer Sie im Laufe eines Algorithmus eine Zahl einlesen müssen, können sie dazu die Funktion `int zahl()` verwenden. Wenn Sie die Algorithmen praktisch ausprobieren wollen, können Sie die besagte Funktion wie folgt definieren:

```
int zahl()
{
    int z;
    scanf("%d", &z);
    return z;
}
```

Randbedingung: Lösen Sie alle Aufgaben ohne die Benutzung von Feldern!

Aufgabe 1

Gegeben sei eine endliche Folge von n ganzen Zahlen ($n > 1$). Lesen Sie die Folge ein und zerlegen Sie die Folge in monotone Teilfolgen. Drucken Sie die Teilfolgen aus. Bei jeder Zustandsänderung der Folge (steigend \leftrightarrow fallend) ist eine neue Zeile zu beginnen.
Beispiel:

```
Eingabe  1 1 2 4 7 3 2 2 1 6 5 4 1 4 6
Ausgabe  1 1 2 4 7
          3 2 2 1
          6 5 4 1
          4 6
```

Aufgabe 2

Eine endliche Folge ganzer Zahlen, die durch Null abgeschlossen wird, soll eingelesen werden und in der Form wieder ausgegeben werden, dass eine Folge aufeinanderfolgender gleicher

Zahlen durch deren Anzahl, einen Multiplikationsstern und ihren Wert ersetzt wird. Es werden nur Lösungen als richtig gewertet, die ohne Felder auskommen.¹

Beispiel:

```
Eingabe  7 7 7 3 5 5 5 5 3 3 0
Ausgabe  3*7 3 4*5 2*3
```

¹Diese Eigenschaft wird auch als „konstante Speicherkomplexität“ bzw. „konstante Platzkomplexität“ bezeichnet.