



Passwörter

Du brauchst ein neues Passwort?

Bitte sehr:

Dfg4T&o%kkM

Gängige Passwort-Generatoren geben gerne derartige Zeichenfolgen aus. Die Zeichen werden zufällig gewählt, so dass es schwierig ist, das gesamte Passwort automatisiert oder manuell zu erraten.

Leider haben solche Passwörter den Nachteil, dass es auch schwierig ist, sie sich zu merken, und man dazu neigt, sie sich doch irgendwo aufzuschreiben.

Leichter zu merken sind Passwörter, die man einigermaßen gut aussprechen kann. Auch solche Passwörter können mit Hilfe geeigneter Regeln automatisch erstellt werden. Zum Beispiel sollten nicht zu viele Ziffern hintereinander enthalten sein, denn lange Zahlen sind schwierig zu merken. Außerdem sollte ein Buchstabe nicht mehrfach hintereinander vorkommen, denn Doppel-Buchstaben sind meist schwer auszusprechen.

Junioraufgabe 1

Schreibe einen eigenen Passwort-Generator. Dein Programm soll zufällige und sinnlose Zeichenfolgen ausgeben, die schwer zu erraten, aber einigermaßen gut auszusprechen und deshalb leichter zu merken sind.

Überlege dir und implementiere mindestens drei eigene Regeln, nach denen dein Programm arbeiten soll. Begründe zu jeder Regel, warum sie die Aussprechbarkeit bzw. Merkbarkeit von Passwörtern verbessert.

Es könnte sein, dass dein Programm auch so ein Passwort ausgibt:

schriopmeta9bunzquaEke

Es ist doppelt so lang wie das Passwort oben. Trotzdem kann man es sich besser merken.

MississippiBlues39

Baulwürfe

Ein Forschungsteam will das Verhalten der Maulwürfe untersuchen, die offenkundig, aber unterirdisch ein großes Feld bevölkern und dort ihre Hügel aufwerfen. Das Team hat das Feld gesichtet und dabei eine sogenannte Hügelkarte erstellt. Eine Hügelkarte ist in kleine Planquadrate eingeteilt; jedes Planquadrat, auf dem sich ein Hügel befindet, ist mit X markiert. Hier ist ein Ausschnitt dieser Karte:

```

XXX XXXXXXXXXXXX XXXXXXX
X X X XXX XXX XXXXXXX
X X XXXXXXXXXX XXXXXXX
XXX XXXXXXXXXXXX XXXXXXX
XXXX XXXXXXXXXX XXXXXXX
XXXX XXXXXXXXXX XXXXXXX

```

Beim Betrachten der Karte macht das Forschungsteam eine Entdeckung: Es kann nicht sein, dass dort nur gewöhnliche Maulwürfe leben. Ein

Planquadrat mit einem Maulwurfshügel ist nämlich immer rundum von Quadraten ohne Hügel umgeben. Die Markierungen bilden aber auch genau dieses Muster:

```

XXX
X X
X X
XXX

```

Diese grenzen gelegentlich aneinander, überlappen sich aber nicht.

Im Forschungsteam herrscht große Aufregung: Es muss sich um eine neue Maulwurfart handeln, die ihre Hügel zu solch rechteckigen Bauen formiert. Die Forscher taufen sie „Baulwürfe“. Nun wollen sie anhand der Hügelkarte die auf dem Feld befindlichen Baulwurfbaue zählen.

Junioraufgabe 2

Hilf dem Forschungsteam und schreibe ein Programm. Es soll die Hügelkarte einlesen und die Anzahl der Baulwurfbaue ausgeben.

Wende dein Programm auf die Beispiele an, die du auf den [BWINE-Webseiten](#) findest, und dokumentiere die Ergebnisse.

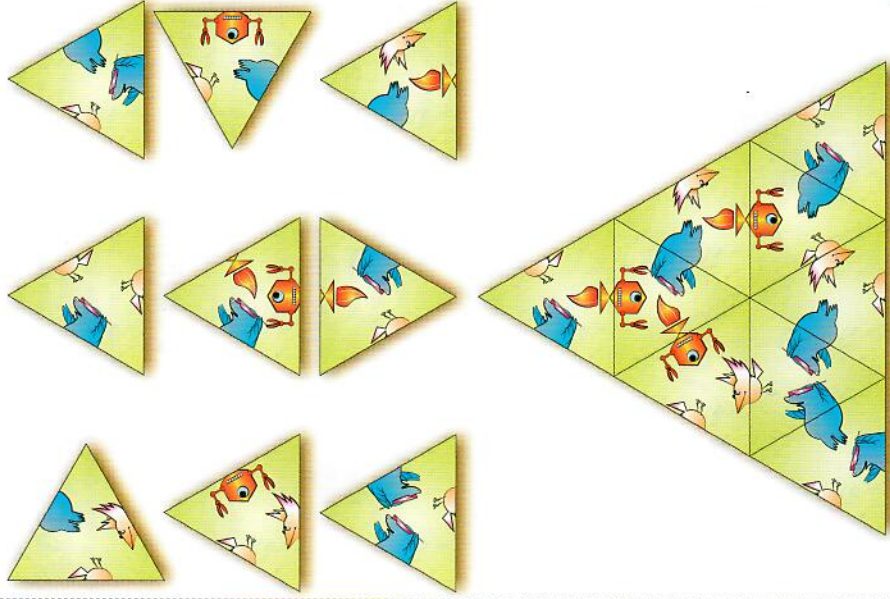
Dreieckspuzzle

Lizzy hat ein Dreieckspuzzle geschenkt bekommen. Es besteht aus neun dreieckigen Teilen. Auf den einzelnen Teilen sind Figuren abgebildet, aber immer nur zur Hälfte.

Aufgabe 2

Die Teile sollen zu einem großen Dreieck zusammengesetzt werden, und zwar so, dass die Figurenhälften zueinander passen. Nach einigem Probieren hat Lizzy eine Lösung gefunden – siehe unten.

Lizzy lernt gerade programmieren und denkt sich: „Bestimmt kann das Puzzle auch mit einem Computerprogramm gelöst werden. Und das Programm schreibe ich gleich so, dass es auch andere Dreieckspuzzles mit neun Teilen lösen kann:



Opä Jürgen blättert in einer Zeitschrift aus der Apotheke und findet ein Rätsel. Es ist eine Liste von Wörtern gegeben, die in die richtige Reihenfolge gebracht werden sollen, so dass sie eine lustige Geschichte ergeben. Leerzeichen und Satzzeichen sowie einige Buchstaben sind schon vorgegeben.

„Oh je, was für eine Arbeit!“ sagt Opä Jürgen. „Kein Problem!“ erwidert seine Enkelin Lotta. „Der Computer kann es doch für uns machen.“

Aufgabe 1

Hilf den beiden und schreibe ein Programm, das einen Lückentext und eine Liste von Wörtern einliest und anschließend den vervollständigten Text ausgibt. Du kannst davon ausgehen, dass es nur eine richtige Lösung gibt.

Wende dein Programm auf die Beispiele an, die du auf den [BWINE-Webseiten](#) findest, und dokumentiere die Ergebnisse.

