

Javakurs SS03 - Übung 3

1 Quadrat Zahlen

Schreibe ein Programm, das die ersten 100 Quadratzahlen (1, 4, 9, 16, ...) ausgibt.

2 Minimum

Schreibe ein Programm, das eine Liste von ganzen Zahlen als Kommandozeilenparameter nimmt, diese in ein `int`-Array schreibt, das Array nach dem kleinsten Wert durchsucht und es ausgibt.

3 Zufällige Karte geben

Aufrufe von `Math.random()` liefern (Pseudo-)Zufallszahlen von `0.0` bis `0.999...` als `double`. Verwende dies um eine `int`-Zufallszahl von `0` bis `3` (für eine zufällige Spielkartenfarbe) und eine zweite von `0` bis `12` (Spielkartenwert) zu erzeugen. Gebe dann textuell Farbe (Karo, Herz, Pik oder Kreuz) und Wert (As, 2 bis 10, Bube, Dame, König) aus. Verwende für einen Ausgabeteil eine `switch`-Anwendung und für den anderen die Zufallszahl als Index eines Stringarrays.

4 1D Zelluläre Automaten und ASCII-Graphik

Stellt euch eine Reihe von Zellen vor, die in einer Linie nebeneinander liegen. Jede Zelle hat einen Zustand, tot oder lebendig. Nun sollen sich die Zustände der Zellen in der Zeit schrittweise verändern, und zwar ist der Zustand abhängig von dem alten Zustand und den Zustand der beiden benachbarten Zellen.

Den Zustand von je drei Zellen kann man als eine Zahl von `0` bis `7` auffassen: die Zustände als `0/1`-Ziffern der Binärcodierung. Zum Beispiel Tot-Lebendig-Tod entspricht $010_2 = 2$.

Eine Regel für den zellulären Automaten definiert nun für jeden Zellenzustand unter Berücksichtigung ihrer Nachbarn, d.h. es ergibt sich eine Tabelle mit acht Einträgen.

Bsp.:

Nachbarschaft	Nachfolger
LLL (7)	L (1)
LLT (6)	T (0)
LTL (5)	T (0)
LTT (4)	L (1)
TLL (3)	T (0)
TLT (2)	L (1)
TTL (1)	L (1)
TTT (0)	T (0)

Damit kann man dann die Regel einfach als Folge von acht Bits auffassen, also als Zahl von 0 bis 255 (im Beispiel $10010110_2 = 150$).

Schreibe ein Programm, das eine Zahl 0 bis 255 als Kommandoparameter einliest, und die Entwicklung der Zellen in der Zeit berechnet. Gib in jeden Zeitschritt (*Generation*) die Zellen in einer Zeile aus, jede Zelle je nach Zustand als anderen Buchstaben (z.B. mit `System.out.print('')` und `System.out.print('*')`), am Ende der *Population* für einen Zeitpunkt Zeilenumbruch mit dem Aufruf `System.out.println()`. Für die Zellen am Rand ist es am einfachsten, wenn Du die linkeste und rechteste Zelle als benachbart ansiehst („wraparound“).

Am interessantesten ist es, wenn Du die Anfangspopulation zufällig belegst. Dazu kannst Du die den Aufruf `Math.random()` verwenden, der (Pseudo-)Zufallszahlen von 0.0 bis (ausschließlich) 1.0 als `double` liefert.

Für einige Regeln sehen die Ergebnisse den Mustern auf Muscheln recht ähnlich. Siehe auch <http://www.inf.fu-berlin.de/lehre/SS03/javakurs/util/cell1d.html>.

5 Rätsel lösen

In dem folgenden Rätsel sind Ziffern durch Buchstaben ersetzt:

```

OMA
+OPA
----
PAAR

```

Schreibe ein Programm, das alle Lösungen ausgibt.

Wenn Du noch Lust hast, kannst Du auch noch die Lösung von

SEND
+MORE

MONEY

suchen. (Wenn Du davon ausgehst, das die drei Zahlen nicht mit Null beginnen, so gibt es für dieses zweite Rätsel eine eindeutige Lösung.)