

Praktikum 5

Ziele

1. Spielsimulation
2. Sortieralgorithmen
3. Dateien lesen und schreiben

Aufgaben

Aufgabe 1

Erstellen Sie eine Software, die eine Million Mal das Würfelspiel Craps simuliert. Es müssen keine Einsätze oder Wetten berücksichtigt werden. Für die Spielregeln können Sie sich an die Beschreibung des deutschen Wikipedia-Artikels¹ halten:

[...] der Shooter [wirft] gleichzeitig zwei Würfel.

Wirft der Shooter im ersten Wurf:

- die Augensumme 7 oder 11, so hat er ein Natural und gewinnt sofort.
- die Augensumme 2, 3 oder 12, so ist dies ein Crap und er verliert sofort.
- die Augensumme 4, 5, 6, 8, 9 oder 10, so ist die geworfene Augensumme sein Point, und der Shooter würfelt ein weiteres Mal.

Ab dem zweiten Wurf gilt:

- Wirft der Shooter dieselbe Augensumme wie im ersten Wurf, also seinen Point, so gewinnt er.
- Wirft der Shooter die Augensumme 7, so verliert er („Sieben – aus dem Spiel“).
- Wirft er irgendeine andere Augensumme, so würfelt er ein weiteres Mal.

¹ <https://de.wikipedia.org/wiki/Craps>

Mit Hilfe Ihres Programms sollen Sie folgende Fragen beantworten können:

- Wie viele Spiele werden gewonnen beim ersten Wurf, beim zweiten Wurf... usw. bis zum 20. Wurf. Wie viele Spiele werden erst nach dem 20. Wurf gewonnen?
- Wie viele Spiele werden verloren beim ersten Wurf, beim zweiten Wurf... usw. bis zum 20. Wurf. Wie viele Spiele werden erst nach dem 20. Wurf verloren?
- Wie groß sind die Chancen Craps zu gewinnen? (Sie sollten herausfinden, dass Craps eines der fairsten in Spielkasinos gespielten Spiele ist. Was bedeutet diese Aussage?)
- Wie lang dauert (d.h. aus wie vielen Würfeln besteht) ein Craps-Spiel im Durchschnitt?
- Erhöhen sich die Gewinnchancen mit der Dauer eines Craps-Spiels?

Aufgabe 2

Nehmen Sie eine Zahl per Tastatur entgegen (denken Sie an die Fehlerbehandlung!) und dementsprechend viele Namen. Implementieren Sie zwei verschiedene Sortieralgorithmen (es bieten sich an: Selection Sort, Insertionsort, Bubblesort - ganz Mutige versuchen sich an einem rekursiven Quicksort). Sortieren Sie die Namen wahlweise mit dem einen oder anderen Algorithmus und schreiben Sie das Ergebnis in eine Datei.

Aufgabe 3

Lesen Sie die beigefügte Datei `biotechnology.txt` aus und analysieren Sie die absoluten Häufigkeiten der einzelnen Wörter mit Hilfe eines Hash. Für die Zerlegung des Textes `$text` in einzelne Wörter hilft Ihnen folgender Code-Snipsel:

```
$text =~ s/[\-\. \? \, \! \; ]//g; # Löschung der häufigsten Satzzeichen
@words = split(/\s+/, $text); # Zerlegung an Leerzeichen
```

Geben Sie die zehn (hundert) häufigsten Wörter mit ihrem Vorkommen aus.