

Strings

1 Passwörter

- Schreiben Sie ein Programm, mit dem zufällige Passwörter generiert werden können. Welche Zeichenklassen (Großbuchstaben, Ziffern, Sonderzeichen, ...) zugelassen sind sollte konfigurierbar sein! Implementieren Sie auch eine Funktion mit der Sie beliebig viele Passwörter beliebiger Länge generieren können.
- Schreiben Sie eine Funktion, die überprüft ob ein Passwort gültig und sicher ist. Wir definieren ein gültiges/sicheres Passwort wie folgt: das erste Zeichen ist ein Kleinbuchstabe; das Passwort enthält mindestens einen Großbuchstabe, eine Ziffer und ein Sonderzeichen; das Passwort hat mindestens 7 Zeichen. Erzeugen Sie mit der zuvor implementierten Funktion 100 gültige Passwörter. Berechnen Sie dabei auch das Verhältnis von gültigen zu ungültigen Passwörtern.

2 Textanalyse

- Schreiben Sie ein Programm, um Texte zu analysieren. Normalisieren Sie den Text zuerst, indem Sie Sonderzeichen eliminieren und den Text in Kleinbuchstaben umwandeln. Zählen Sie wie oft jedes Wort im Text vorkommt. Implementieren Sie auch Funktionen um alle Wörter nach Häufigkeit sortiert auszugeben, und um die n häufigsten Wörter auszugeben.
- Testen Sie ihre Programm mit dem Buch „Pride and Prejudice“ von Jane Austen. Download: <http://www.gutenberg.org/cache/epub/1342/pg1342.txt>
- Erweitern Sie ihr Programm, sodass es *auch* n-Gramme unterstützt. Ein n-Gramm ist ein Wortfragment der Länge n, Beispiel: die 3-Gramme des Strings „Das ist Text“ sind „Das“, „ist“, „Tex“ und „ext“. Es gibt keine 3-Gramme des Wortes „er“!
- Erweitern Sie ihr Programm, sodass auch die Position von Wörtern/n-Grammen gespeichert wird. Beispiel: bei „Test, das ist ein Test“ wären die Positionen: Test:0 und 4; das:1; ist:2; ein:3
- Erweitern Sie ihre Programm, sodass Sie nicht nur einen, sondern mehrere Texte lesen können. Beispiel: Sie lesen 2 Dokumente a und b ein; natürlich möchten Sie für a und b getrennte Statistiken erstellen!