

Aufgabe 2: Stringsuche mit Brute-Force und Boyer-Moore

Implementieren Sie die Stringsuche-Algorithmen nach Brute-Force und Boyer-Moore.

```
TextSearcher s = new BruteForceSearcher();
Out.println(s.countOccurrences("wettergott", "gott", false)); // Ausgabe 1
Out.println(s.countOccurrences("wettergott", "tt", false)); // Ausgabe 2
Out.println(s.countOccurrences("wettergott", "t", false)); // Ausgabe 4
Out.println(s.countOccurrences("er ist kaiser", "er", false)); // Ausgabe 2
Out.println(s.countOccurrences("er ist kaiser", "er", true)); // Ausgabe 1
```

Implementieren Sie die Klassen *BruteForceSearcher* und *BoyerMooreSearcher* im Paket *at.jku.students* mit folgenden Schnittstellen (alles *public*):

```
class BruteForceSearcher extends TextSearcher {
    int countOccurrences(String text, String pattern, boolean wholeWordOnly) { ... }
}
class BoyerMooreSearcher extends TextSearcher {
    int countOccurrences(String text, String pattern, boolean wholeWordOnly) { ... }
}
abstract class TextSearcher {
    abstract int countOccurrences(String text, String pattern, boolean wholeWordOnly);
}
```

Die *countOccurrences*-Methode durchsucht den Text und zählt wie oft das Muster vorkommt. Abhängig vom Parameter *wholeWordOnly* zählt ein Muster nur dann als gefunden wenn es als eigenes Wort vorkommt, oder aber auch wenn es innerhalb eines Wortes steht. Die Groß-/Kleinschreibung soll ignoriert werden.

Testen Sie Ihre Implementierung mit dem vorgegebenen Programm *TextSearcherTest.java*. Dieses Programm durchsucht den Text aus der Datei *alice.txt* nach den Mustern in der Datei *patterns.txt*.

alice.txt

```
Alice's Adventures in Wonderland
Lewis Carroll
CHAPTER I
Down the Rabbit-Hole
Alice was beginning to get very tired of sitting by
her sister on the bank, and of having nothing to do:
once or twice she had peeped into the book her sister
was reading, but it had no pictures...
```

patterns.txt

```
a
affectionately
alarm
alarmed
alas
alice
alive
all...
```

Das Testprogramm sucht jedes Muster mit Brute-Force und Boyer-Moore und misst die benötigte Zeit. Um die Messgenauigkeit zu erhöhen, wird jede Suche 100-mal wiederholt. Folgende Tabelle zeigt das Messergebnis.

Pattern	#	Brute	#	Boyer

a	634	116	634	115
affectionately	1	85	1	15
alarm	2	86	2	23
alarmed	1	82	1	22
alas	4	88	4	41
alice	417	86	417	37
alive	3	83	3	33
all	182	83	182	37
allow	3	83	3	22
almost	6	82	6	29
alone	4	82	4	39
along	6	82	6	25
aloud	5	82	5	23
already	2	81	2	18
also	2	88	2	32
altered	1	82	1	20
alternately	1	83	1	14
altogether	5	83	5	22
...				

	6193	3381	6193	1342

Abzugeben ist: Java-Programm, Testergebnisse