

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

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

```
TextSearcher s = new BruteForceSearcher();
Out.println(s.countOccurrences("wettergott", "Gott", false, false)); // Ausgabe 1
Out.println(s.countOccurrences("wettergott", "Gott", true, false)); // Ausgabe 0
Out.println(s.countOccurrences("wettergott", "tt", false, false)); // Ausgabe 2
Out.println(s.countOccurrences("wettergott", "t", false, false)); // Ausgabe 4
Out.println(s.countOccurrences("Er ist Kaiser", "er", false, false)); // Ausgabe 2
Out.println(s.countOccurrences("Er ist Kaiser", "er", false, true)); // Ausgabe 1
Out.println(s.countOccurrences("Er ist Kaiser", "er", true, true)); // Ausgabe 0
```

Implementieren Sie die Klassen *BruteForceSearcher* und *BoyerMooreSearcher* im Paket *at.jku.ssw.pi.search.ue10* mit folgenden Schnittstellen (alles *public*):

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

Die *countOccurrences*-Methode durchsucht den Text und zählt, wie oft das Muster vorkommt. Wenn der Parameter *matchCase* den Wert *true* hat, dann soll die Groß-/Kleinschreibung berücksichtigt werden. Wenn der Parameter *wholeWordOnly* den Wert *true* hat, zählt ein Muster nur dann als gefunden, wenn es als eigenes Wort vorkommt.

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	166 634 102
affectionately	1	144 1 13
alarm	2	139 2 20
alarmed	1	140 1 22
alas	4	137 4 26
alice	417	137 417 26
alive	3	140 3 24
all	182	137 182 31
...		
	6786	6154 6786 1000

Implementierungshinweise:

- Verwenden Sie die Methode *isLetterOrDigit(char c)* der Klasse *Character* um zu prüfen, ob ein Zeichen ein Buchstabe oder ein Leerzeichen ist.
- Verwenden Sie die Methoden *toLowerCase(char c)* oder *toUpperCase(char c)* der Klasse *Character* um einen Buchstaben in einen Kleinbuchstaben oder Großbuchstaben zu verwandeln.