

Writeup Excercise 2

Arianova Mendioroz (s0518061)
Bernhard Johannes Eiling (s0535003)

Pre-Lab:

1. Diese Funktionalität kann über eine HashMap realisiert werden. Zuerst erstellt man eine HashMap mit dem Indeztypen „Character“ (der hier gewählte Typ „Integer“ als Inhalt der Hashmap ist optional):

```
HashMap<Character, Integer> numChar = new HashMap<Character, Integer>();
```

Danach kann die leere Hasmap mit einem Index versehen werden. Dies lässt sich am einfachsten durch eine for-Schleife realisieren, in der der Index i in einen Character gecastet wird:

```
for (int i = 97; i < 123; i++) {  
    numChar.put((char)i, 0);  
}
```

Hier wird vom ASCII Index 97 bis 122 gelaufen, die genau das Alphabet der kleinen Buchstaben repräsentieren. Als default Wert wird die „0“ eingetragen.

Nun kann die HashMap über Buchstaben angesprochen werden und ein beliebiger Integer an dieser Position eingetragen werden.

2. Um einen Character zu normalisieren kann man dessen Integer Repräsentation nutzen. In der folgenden Methode wird überprüft ob der Character im Intervall der Groß- oder Kleinbuchstaben liegt und im Falle eines Großbuchstaben in einen Kleinbuchstaben umgewandelt. Dies geschieht durch Addition von 32 zu der Integerrepräsentation des Buchstaben.

```
static char normalizeChar(char chara) {  
    if ((int)chara > 64 && (int)chara < 91) {  
        int ascii = (int)chara + 32;  
        return (char)ascii;  
    } else {  
        return chara;  
    }  
}
```

Eine Umwandlung von Klein- zu Großbuchstabe könnte man mittels Subtraktion von 32 erreichen.

3. Ein „carriage return“ ist ein Kommande mit dem ein Programm zurück zum Beginn einer Zeile springen kann.

Das Wort leitet sich von der Schreibmaschine ab. Hier musste der Wagen um wieder zum Beginn der gerade geschriebenen Zeile zu springen auf die Anfangsposition fahren. Daher der Name „Wagenrücklauf“ = „carriage return“.

Lab Exercise:

1. Um einzelne Zeichen einzulesen verwenden wir einen `BufferedReader` dem wir die einzulesende Datei übergeben. Mit der Methode `read()` werden nun einzeln alle Zeichen der Datei eingelesen. Um nur Buchstaben zu erhalten wird überprüft ob der ASCII-Wert des eingelesenen Zeichens sich im Intervall der Buchstabenwerte befindet.

Wenn kein Zeichen mehr zum einlesen vorhanden ist gibt die `read()` Methode -1 zurück. Dadurch wird die `while`-Schleife abgebrochen.

2. Um eine Datei mit beliebigem Inhalt zu schreiben verwenden wir einen `BufferedWriter`. Alle Zeichen die in die Datei geschrieben werden sollen werden mit der Methode `write()` dem `BufferedWriter` übergeben. Die eigentliche Datei wird mittels einer `FileWriter` erzeugt.

5. Die Komplexität des Algorithmus beträgt $O(n)$.