Prof. Dr. Stefan Brass
Dipl.-Inform. Vaibhav Kasturia

Martin-Luther-Universität Halle-Wittenberg
Institut für Informatik
Wintersemester 2018/19

Objektorientierte Programmierung: Hausaufgabenblatt 8

Abgabe: 17.12.2018, 11:00

Mit diesem Übungsblatt soll die Verwendung von Zeichenketten, Arrays und statischen Methoden geübt werden.

Hausaufgabe 1: (7 Punkte)

Bitte beachten Sie: die Einsendung der Aufgabe erfolgt nur im YAPEX.
Open exercises via code → Freigabecode: `5cook9ikecnm-6131` (Buchstaben zählen)

Schreiben Sie ein Programm, welches die Anzahl Vorkommen jedes Buchstabens in einer Eingabezeile ausgibt. Das Programm soll die eingelesene Zeichenkette Zeichen für Zeichen durchgehen. Alle Zeichen, die nicht ein Großbuchstabe `A` bis `Z` oder Kleinbuchstabe `a` bis `z` sind, sollen ignoriert werden. Auch Umlaute wie `Ä` sollen ignoriert werden. Für die 26 Buchstaben benötigt man jeweils einen Zähler (natürlich nicht einzelne Variablen, sondern ein Array). Der Zähler soll bei jedem Vorkommen des entsprechenden Buchstabens (z.B. `a` und `A`) um 1 erhöht werden.
Das Programm soll in einer Klasse "`Buchstaben`" stehen. Lesen Sie mit der Methode [`nextLine()`] der Scanner-Klasse eine Textzeile ein. **Um YAPEX-Timeout-Probleme zu vermeiden, soll Ihr Programm auf eine Eingabeaufforderung verzichten.** Geben Sie für alle Zeichen, welche mindestens einmal im Text vorkommen, eine Tabelle der Form `Zeichen: Anzahl (Prozent)` aus (alphabetisch sortiert). Das Zeichen soll dabei der Großbuchstabe sein, die Anzahl aber auch alle Vorkommen als Kleinbuchstabe mit enthalten. Die Prozentzahl soll sich auf alle Buchstaben im Text beziehen (ignorierte Zeichen sollen nicht mitgezählt werden) und ggf. abgerundet sein (wie es sich bei der Integer-Division automatisch ergibt). Ein Beispiel (mit unterstrichener Eingabe des Nutzers) ist:

```
Mississippi!                      // Eingabe
I: 4 (36%)                        // Ausgabe
M: 1 (9%)
P: 2 (18%)
S: 4 (36%)
```

Hinweise: Da Java auf Unicode basiert, und Unicode aufwärtskompatibel zu ASCII ist, kann man sich darauf verlassen, dass die 26 Großbuchstaben Codes haben, die direkt aufeinander folgen. Gleiches gilt auch für die Kleinbuchstaben. Z.B. könnte man mit `c-'A'` einen Wert zwischen 0 und 25 berechnen, wenn man sicher ist, dass in der Variablen `c` vom Typ `char` einer der Großbuchstaben steht. Das könnte man vorher durch Vergleiche wie `c >= 'A'` sicherstellen. (Bei Bedarf schaue man sich die Methode [`toUpperCase()`] der `String`-Klasse oder die statische Methode [`Character.toUpperCase(char c)`] in der Dokumentation an.)

Hausaufgabe 2: (7 Punkte)

Bitte beachten Sie: die Einsendung der Aufgabe erfolgt nur im YAPEX.
Open exercises via code → Freigabecode: `5cook9j4c11a-2266` (BT)

Schreiben Sie eine statische Methode `binomFormel(int n)`, die die bionomische Formel für $(a+b)^n$ für beliebiges n ausgibt. Die Klasse soll `BF` heissen, das Hauptprogramm ist folgendermaßen vorgegeben:

```
// Hauptprogramm:
public static void main(String args[]) {

        // Wert fuer Parameter n beschaffen (Argument oder Eingabe):
        int n;
        if(args.length > 0)
                n = Integer.parseInt(args[0]);
        else {
                Scanner scanner = new Scanner(System.in);
                System.out.print("n = ");
                n = scanner.nextInt();
        }
        if( n <= 0) {
                System.out.println("Eingabe muss > 0 sein!");
                return;
        }

        // Ihre Methode macht die eigentliche Arbeit:
        binomFormel(n);
}
```

Für $n = 2$ soll die Methode folgende Ausgabe drucken ($a^2 + 2ab + b^2$):

```
a^2 + 2ab + b^2
```

Für $n = 3$ soll Folgendes ausgegeben werden (es steht für $a^3 + 3a^2b + 3ab^2 + b^3$):

```
a^3 + 3a^2b + 3ab^2 + b^3
```

Allgemein ist die [binomische Formel] (siehe auch [Binonischer Lehrsatz]) bekanntlich

$$(a+b)^n = \sum_{k=0}^{n} \binom{n}{k} a^{n-k}\, b^k.$$

Dabei ist der [Binomialkoeffizient]:

$$\binom{n}{k} = \frac{n!}{k!\,(n-k)!} = \frac{n*(n-1)*\cdots*(n-k+1)}{k*(k-1)*\cdots*1}$$

In der letzten Darstellung haben Zähler und Nenner jeweils k Faktoren. Sie dürfen gerne weitere statische Methoden schreiben, z.B. eine für den Binomialkoeffizienten, und ggf. eine für die Fakultät $n! = 1*2*\cdots*n$. Beachten Sie, dass Binomialkoeffizient und Exponent jeweils nur gedruckt werden sollen, wenn sie nicht 1 sind, und auch a^0 bzw. b^0 nicht ausgegeben werden sollen (wie in den obigen Beispielen).