

Projektblatt P1 (20 P)

Abgabe: Montag 19. September 2022, 10:00h

Entpacken Sie zunächst die Archiv-Datei `vorgaben-P1.zip`, in der sich neben mehreren Rahmendateien für die zu lösenden Aufgaben auch die Hilfsklasse `IOTools.java` befindet. Ergänzen Sie alle Dateien mit Ausnahme von `IOTools.java` zunächst durch einen Kommentar, der Ihren Namen beinhaltet. Ergänzen Sie die Dateien dann durch Ihre Lösungen gemäß der Aufgabenstellung unten. Der abgeänderte bzw. hinzugefügte Java-Sourcecode in den Dateien `Aufgabe_1_1.java`, `Aufgabe_1_2.java`, `Aufgabe_1_3.java` und `Aufgabe_1_4.java` sollte syntaktisch richtig und vollständig formatiert sein. Antworten in Textform müssen stets in Form von (Block-)Kommentaren erfolgen. Alle Dateien mit der Endung `.java` sollten am Ende fehlerfrei übersetzt werden können.

Verpacken Sie die Dateien für Ihre Abgabe in einem ZIP-Archiv mit dem Namen `IhrNachname.IhrVorname.P1.zip`, die Sie auf Ilias hochladen.

Führen Sie dazu in dem Verzeichnis, in dem Sie die Dateien bearbeitet haben, folgenden Befehl auf der Kommandozeile aus:

```
zip IhrNachname.IhrVorname.P1.zip Aufgabe_1_*.java
```

Aufgabe 1: Fehler finden

5P

Die folgende `main`-Methode enthält zehn Zeilen mit je (mindestens) einem syntaktischen Fehler. Diese fehlerhaften Zeilen sollen Sie nun finden, die Fehler darin beschreiben und eine mögliche Korrektur angeben.

Ergänzen Sie die Datei `Aufgabe_1_1.java` dazu wie folgt:

Fügen Sie nach jeder Zeile, in der Sie einen Fehler entdecken, einen (mehrzeiligen) Kommentar ein, indem Sie kurz beschreiben, worin der bzw. die Fehler bestehen. Kommentieren Sie dann die fehlerhafte Zeile aus und fügen Sie eine neue (nicht auskommentierte) Zeile hinzu, in der Sie den bzw. die Fehler korrigieren. Korrigieren Sie den Code so, dass die entsprechende Codezeile keine Fehler mehr enthält und dokumentieren Sie Ihre Änderung. Die Korrektur sollte, gemessen an geänderten Zeichen in der Zeile, möglichst geringfügig ausfallen (allerdings dürfen Sie die jeweilige Zeile nicht einfach auskommentieren).

Beispiel:

```
// int x = 1.0;
int x = 1;
// 1.0 ist eine Gleitkommazahl und kann daher nicht einer
// int Variablen zugewiesen werden.
// 1.0 zu int Wert 1 geändert
```

```
1    public static void main(String[] args){
2        System.out.println("Übung " + 1 + ':' );
3        int a1=1,a2=a1;a3=a1+a2;
4        boolean b = (3 < 2);
5        short s1 = 33333;
6        char c1 = 'ü';
7        char c2 = '1', c3 = c2 + 1;
8        float f = 1.0;
9        double d = 1E2E;
10       int i = 'uOOOO';
11       long l = 0xABC;
12       char c = '\';
13       short s2 = 10, s3 = 3, s4 = (short) s2 % (short) s3;
14       int bb = (0x1 < 0x2) ? (0xA / 0x5) : 0x;
15   }
```

Hinweis: Verwenden Sie den Compiler, um die Fehler zu finden. Übersetzen und verändern Sie dazu das (inhaltlich völlig sinnlose) Programm in der Datei `Aufgabe_1_1.java` schrittweise, indem Sie jeweils die Kommentarzeichen vor einer der Zeilen entfernen und dann das Programm übersetzen. Analysieren Sie den (bzw. die) angezeigten Fehler und ergänzen Sie anschließend die Datei wie oben beschrieben.

Aufgabe 2: Operationen/Ausdrücke nachvollziehen 5P

Gegeben seien die ganzzahligen (Typ `int`) Variablen

$a = 1, b = 2, c = 3, d = 4$.

Geben Sie für jeden der folgenden Ausdrücke an, welchen Typ und Wert das Ergebnis hat. Begründen Sie Ihre Antwort, indem Sie die bei der Auswertung des Ausdrucks nacheinander ausgeführten Operationen und die dazugehörigen Zwischenergebnisse angeben. Falls eine Variable in dem jeweiligen Ausdruck verändert wird (z.B. durch einen Inkrement/Dekrement-Operator), schreiben Sie zusätzlich die Wirkung dieses Operators auf die Variable dazu. Sie dürfen davon ausgehen, dass bei jeder Teilaufgabe zu Beginn die vier Ausdrücke $(a == 1)$, $(b == 2)$, $(c == 3)$ und $(d == 4)$ zu `true` ausgewertet würden.

Ergänzen Sie hierzu die Datei `Aufgabe_1_2.java` wie im Folgenden beschrieben.

- Ergänzen Sie die Erklärungen zur Auswertung der Ausdrücke und welche Werte der Ausdruck sowie die Variablen a, b, c und d am Ende haben, in den Blockkommentaren zu den Aufgabenteilen (a) - (f).
- Entfernen Sie die Kommentarzeichen vor den Ausdrücken, deklarieren Sie eine Variable mit dem für das jeweilige Ergebnis passenden Typ und weisen Sie den Ausdruckswert dieser Variablen zu.
- Ergänzen Sie die Ausgabeanweisungen so, dass der Ausdruckswert ausgegeben wird.

Beispiel: Für den Ausdruck $c++ * (d - b)$ ergibt sich:

```
/* Aufgabenteil (x)
Ergebnistyp: int
Ergebniswert: 6
c++ -> 3 und c = 4,
```

```

d - b -> 2
3 * 2 -> 6
*/
int e = c++ * (d - b);
System.out.println("e = " + e);

```

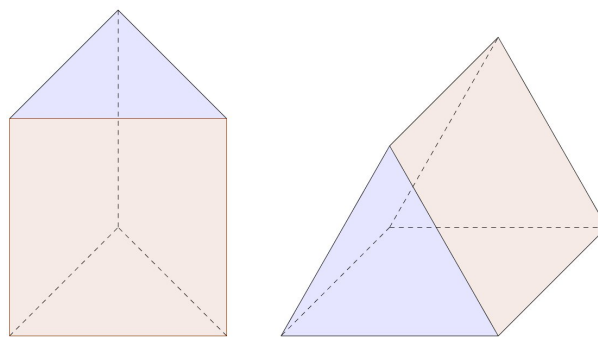
Anmerkung: Die Codestücke enthalten zum Teil stilistisch sehr schlechten Code und dienen nur dazu, zu testen, inwieweit Sie mit komplexen Ausdrücken zurechtkommen.

- (a) $a + b / c - d;$
- (b) $a + ++a + a++;$
- (c) $c *= d /= ++a;$
- (d) $c != b + a || a + c > d;$
- (e) $(++a + ++b == d-- + --d) ? a : c;$

Aufgabe 3: Arithmetische Ausdrücke

3P

*"Ein Prisma ... ist ein geometrischer Körper, der durch Parallelverschiebung eines ebenen Vielecks entlang einer nicht in dieser Ebene liegenden Geraden im Raum entsteht"*¹ Das folgende Bild zeigt ein Prisma, bei dem das Vieleck, welches die Grundfläche bildet, ein gleichseitiges Dreieck ist.



¹Quelle: Wikipedia

Sei nun a die Kantenlänge dieses Dreiecks und h die Höhe des Prismas. Dann lassen sich die Grundfläche G , die Mantelfläche M , die Oberfläche O und das Volumen V dieses Prismas wie folgt berechnen:

$$G = \frac{a^2}{4} \cdot \sqrt{3}$$

$$M = 3 \cdot a \cdot h$$

$$O = 2 \cdot G + M$$

$$V = G \cdot h$$

Vervollständigen Sie die Klasse in der Datei `Aufgabe_1_3` zu einem Programm, welches zunächst die ganzzahlige Kantenlänge und die Höhe eines solchen Prismas einliest und dann die oben genannten Größen G , M , O und V berechnet und gerundet ausgibt. Ergänzen Sie dazu die `main`-Methode der Klasse `Aufgabe_1_3` wie folgt:

- Fordern Sie den Benutzer nacheinander auf, einen ganzzahligen Wert für die Kantenlänge a und die Höhe h zu spezifizieren und lesen Sie diese dann mit Hilfe der `IOTools` ein. Speichern Sie die Werte in zwei passenden Variablen.
- Legen Sie Variablen für G , M , O und V an und berechnen Sie mit Hilfe der obigen Formeln die zugehörigen Größen. Sorgen Sie dafür, dass das Ergebnis zunächst in `double`-Genauigkeit berechnet wird. Verwenden Sie die Methode `Math.sqrt` für die Berechnung der Quadratwurzel einer Zahl.
- Geben Sie das Ergebnis mit Hilfe der Methode `Math.round` gerundet auf die nächste ganze Zahl in der Konsole aus.

Ein Beispiel für einen Programmablauf könnte wie folgt aussehen:

```
Kantenlaenge --> 5
Hoehe        --> 13
```

```
Grundflaeche: 11
```

Mantelflaeche: 195
Oberflaeche: 217
Volumen: 141

Aufgabe 4: Ganzzahldivision. Logische Ausdrücke 4P

Ergänzen Sie die `main`-Methode in der Klasse `Aufgabe_1_4` so, dass ein Programm entsteht, welches überprüft, ob die Ziffern einer vierstelligen Zahl von links nach rechts monoton steigen und das Ergebnis am Ende ausgibt. Die Ziffern $d_1d_2d_3d_4$ einer vierstelligen Zahl steigen genau dann monoton, wenn die folgende Bedingung erfüllt ist:

$$d_i \leq d_{i+1} \text{ für } i \in \{1, 2, 3\}$$

In der `main`-Methode in der Klasse `Aufgabe_1_4` ist bereits eine `int`-Variable vorgegeben, die mit Hilfe der ebenfalls vorgegebenen Methode `readInt4` mit einer vierstelligen ganzen Zahl initialisiert wird. Ergänzen Sie die `main`-Methode nun wie folgt:

- Legen Sie vier ganzzahlige Variablen an und initialisieren Sie diese mit den Werten der einzelnen Ziffern der eingelesenen Zahl.
- Legen Sie eine boole'sche Variable an und initialisieren Sie diese mit dem Ergebnis der oben genannten Bedingung.
- Geben Sie das Ergebnis wie in den folgenden beiden Beispielen in der Konsole aus. Hinweis: Nutzen Sie den ternären Operator innerhalb der Parameterklammern der Methode `System.out.println`, um verschiedene Ausgaben zu generieren.

Beispiel 1: Für die Zahl 1224 sollte das Ergebnis wie folgt aussehen:

Die Ziffern der Zahl 1224 sind monoton steigend.

Beispiel 2: Für die Zahl 1322 sollte das Ergebnis wie folgt aussehen:

Die Ziffern der Zahl 1322 sind nicht monoton steigend.

Aufgabe 5: Der Bedingungsoperator

3P

Ergänzen Sie die `main`-Methode in der Klasse `Aufgabe_1_5`, so dass ein Programm entsteht, welches zunächst drei ganzzahlige Werte einliest und dann die maximale Differenz zwischen zwei dieser Zahlen bestimmt und ausgibt. Beispiel: Werden die Zahlen -3, 11 und 7 eingegeben, sollte die Ausgabe wie folgt aussehen:

Maximale Differenz: 14

Gehen Sie wie folgt vor:

- Fordern Sie den Benutzer nacheinander auf, drei (ganzzahlige) Werte anzugeben, lesen Sie diese Werte ein und speichern Sie diese in drei Variablen vom Typ `int`.
- Legen Sie zwei weitere Variablen vom Typ `int` an und initialisieren Sie diese mit dem Minimum bzw. Maximum der drei Werte. Verwenden Sie hierbei einen geschachtelten, ternären Operator.
- Geben Sie als Ergebnis die Differenz zwischen dem Minimum und dem Maximum aus.