

Freiwilliges Aufgabenblatt #1

Lösen Sie die hier gestellten Aufgaben und kommen Sie mit diesen Lösungen in die Übungsgruppe, in die Sie sich angemeldet haben. Bereits **am Tag zuvor** müssen Sie allerdings bereits in TUWEL angegeben haben, welche Aufgaben Sie tatsächlich gelöst haben. Geben Sie nur die Beispiele (=Aufgaben) als gelöst an, welche vollständig und wie verlangt durchgeführt wurden.

Dieses erste Aufgabenblatt ist freiwillig. Damit ist gemeint, dass es keinen Prüfungscharakter hat. Es wird allerdings unbedingt empfohlen die Aufgaben durchführen und zu dem Termin zu kommen.

Nachdem in der ersten Übungseinheit das Kennenlernen der Kollegen im Vordergrund stand, geht es in dieser Übung um das gemeinsame Bearbeiten der Aufgabe und spielerische Kennenlernen von Programmieren, ohne tatsächlich eine Aufgabe am Computer lösen zu müssen. Die Aufgaben können auf Papier gelöst werden. Wenn Sie wollen dürfen Sie die Aufgaben jedoch gerne in einem Java-System ausprobieren.

Worte, die in einem Programm vorkommen können sind *in typewriter geschrieben*. Hinweise und Verweise sind auf der Seite geschrieben. Worte *in kursiv geschrieben* führen einen neuen Begriff ein.

Wenn Sie es nicht bereits gemacht haben, lesen Sie bitte die relevanten Teile des Skriptums und erarbeiten Sie gemeinsam die bereits durchgegangenen Fragen am Ende des Kapitels.

Unbedingt auch immer Hinweise lesen!

Siehe Skriptum Seite 77.

FizzBuzz

Aufgabe 1: Bis 100 zählen

FizzBuzz ist ein sehr einfaches Spiel welches Kindern Division spielerisch beibringt. Bei diesem Spiel zählt man von 1 bis 100. Die erste Aufgabe ist es, das Programm auf Seite 13 und 17 im Skriptum so zu ändern, dass es von 1 bis 100 zählt und in jeder Zeile eine Zahl ausgibt.

Statt `UnbekannteZahl` dürfen Sie natürlich auch gerne eine Klasse mit besser passendem Namen wählen. Die Methoden `gleich` und `kleiner` sind nicht mehr notwendig. Allerdings wird eine neue Methode `ausgeben` benötigt welche die tatsächliche Arbeit (Schleife und Zahlen ausgeben) durchführen soll.

Gehen Sie so an diese Aufgabe heran, dass jedes Gruppenmitglied einen Lösungsvorschlag entwirft. Dann vergleichen Sie Ihre Vorschläge, wählen den besser geeigneten aus und überarbeitet diesen Vorschlag gemeinsam.

Aufgabe 2: Rechnen

Bei *FizzBuzz* muss man aber nicht nur einfach bis 100 zählen:

1. Zahlen welche ganzzahlig durch **drei** dividierbar sind, werden durch das Wort „Fizz“,
2. Zahlen welche ganzzahlig durch **fünf** dividierbar sind, werden durch das Wort „Buzz“ und
3. Zahlen welche ganzzahlig durch **beides** dividierbar sind, werden durch das Wort „FizzBuzz“ ersetzt.

Hinweis: Gehen Sie Schritt für Schritt vor.

Auch dieses Programm bietet keine sonderliche Schwierigkeiten bei der Implementierung. Die Aufgabe wurde aber berühmt, da bei Bewerbungsgesprächen ein Großteil von Informatikabsolventen nicht in der Lage war diese Aufgabe zu lösen.

Ihre Aufgabe ist es, sich zu zweit zu überlegen wie man mittels mehrerer **if** und Modulo-Operatoren „%“ anstatt der Zahlen das richtige Wort in einer Zeile ausgeben kann.

Verändern Sie die Methode **ausgeben** Ihrer Klasse, damit sie sich wie oben beschrieben verhält. Gehen Sie dazu wie in Aufgabe 1 vor.

Dieser Operator wurde bereits auf Seite 15 im Skriptum vorgestellt.

Binärzahlen

Aufgabe 3: Matrikelnummer in Binär

Berechnen sie die Binärdarstellung Ihrer Matrikelnummer, jedes Gruppenmitglied für sich, und negieren Sie die entstandenen Binärzahlen (im Zweierkomplement). Kontrollieren Sie gegenseitig, ob die Ergebnisse richtig sind.

Aufgabe 4: Binäre Verknüpfungen

Berechnen Sie die AND-, OR- und XOR-Verknüpfungen der in Aufgabe 3 ermittelten negierten Binärzahlen (jedes Gruppenmitglied) und kontrollieren Sie gegenseitig, ob die Ergebnisse übereinstimmen.

Aufgabe 5: Umwandeln zu Dezimal

Wandeln Sie die in Aufgabe 4 ermittelten Binärzahlen in die Dezimaldarstellung um (jedes Gruppenmitglied) und kontrollieren Sie gegenseitig, ob die Ergebnisse übereinstimmen.

Theoriefragen

Beantworten Sie die Fragen 6, 7, 9 und 10, indem sich jedes Gruppenmitglied eine Antwort überlegt. Anschließend vergleichen Sie Ihre Antworten, wählen eine davon aus und überarbeiten Sie diese gemeinsam.

Aufgabe 6: Architektur üblicher Computer

Erarbeiten Sie wie ein üblicher Computer funktioniert. Was macht die ALU, die Control-Unit und der Hauptspeicher? Schreiben sie ein ganz simples Java Programm welches die Ausführungsschritte eines Computers simuliert.

Vergleiche Seite 32 im Skriptum ganz unten.

Aufgabe 7: Variablen

In welcher Komponente einer üblichen Computer-Architektur liegen die Werte der verwendeten Variablen? Welche Komponente führt die Modulo-Operation von FizzBuzz aus? Überlegen Sie gemeinsam wie oft die Control-Unit und der Bus auf die Daten zugreifen müssen.

Aufgabe 8: Deklaration und Definition

Erarbeiten Sie alleine drei Beispiele von Deklarationen und Definitionen. Danach soll Ihnen Ihr Kollege erklären um was es sich handelt (Deklaration oder Definition) und weshalb das so ist.

Aufgabe 9: Assembler-Sprachen

Was ist eine Assembler-Sprache? Welche Vor- und Nachteile haben Assembler-Sprachen gegenüber höheren Programmiersprachen und reinen Maschinensprachen?

Aufgabe 10: Kommentare

Woran erkennt und wozu braucht man Kommentare? Nennen Sie zumindest zwei Gründe, warum man Kommentare in das Programm schreibt, obwohl der Compiler diesen Text sowieso verwirft. Gehen Sie Ihre Kommentare in FizzBuzz durch: Genügen Sie den Kriterien?