

Übungsblatt 8

Abgabe: 5.1.2009

Aufgabe 1 Beweise, wir brauchen Beweise..... (50%)

Aufgabe 1.1 ggT reloaded

Die folgende JAVA Methode soll den *grössten gemeinsamen Teiler* (ggT) zweier positiver ganzer Zahlen ermittelt.

```
public static int ggt(int wert1, int wert2)
{
    while (wert1 != wert2)
    {
        if (wert1 > wert2)
            wert1 = wert1 - wert2;
        else
            wert2 = wert2 - wert1;
    }
    return wert1;
}
```

Zeigt mit Hilfe der Regeln der operationellen Semantik, dass diese Methode ihren Zweck erfüllt. Die bekannten Axiome:

1. $ggT(x, x) = x$.
2. $ggT(x, y) = ggT(y, x)$.
3. $ggT(x, y) = ggT(x, y - x)$ für $x < y$.

könnten dabei helfen.....

Aufgabe 1.2 Ein paar Punkte vom Weihnachtsjan (+20%)

Zeigt, dass die Methode terminiert, falls `wert1` und `wert2` positive ganze Zahlen sind!

Aufgabe 2 Weihnachtliche Märchenstunde: Die Bremer Stadtmusikanten (50%)

Unsere vier bekanntesten Bremer (nein, nicht Thomas Schaaf, James Last, Thorsten Frings und Willi Lemke) stehen vor dem Problem, einen Bach überqueren zu müssen: Es gibt auf beiden Seiten einen kleinen Steg und in der Mitte einen Stein, leider kann

jeweils nur ein Musikant auf jeder Stelle stehen - und unsere vier Freunde mögen sich auch nicht soweit voneinander entfernen. Wer bei den Gebrüdern Grimm¹ nachliest, erfährt, wie wichtig Zusammenhalt für unser Quartett ist.

Ein Teil des Problems – der fehlende Platz auf den Stegen und dem Stein – lässt sich einfach lösen: Jeder der schon mal am Bremer Rathaus war, weiss, das man die Bremer Stadtmusikanten stapeln kann. Damit sie jetzt aber über den Bach kommen können, sind einige Aktionen nötig, schliesslich kann ja jedes Tier nur ein kleineres huckepack nehmen.

Also stehen Esel, Hund, Katze und Hahn (übereinander) auf Steg A. Der Hahn springt auf den Stein B, dann springt die Katze auf den anderen Steg C. Nun hüpfte der Hahn ebenfalls rüber nach C, landet also auf dem Rücken der Katze, und macht so Platz für den Hund auf dem Stein B. Danach flattert der Hahn zurück nach A auf den Esel, und die Katze begibt sich auf den Hund auf dem Stein. Nachdem der Hahn jetzt auf die Katze springt, kann der Esel den Bach überqueren und springt von Steg A zum Steg C.

Nun müssen nur noch Hund, Katze und Hahn vom Stein in der richtigen Reihenfolge auf den Rücken des Esels kommen: Also geht der Hahn auf den Esel, die Katze auf den leeren Steg A, dann der Hahn auf den Rücken der Katze und der Hund auf den Rücken des Esels. Nun springt der Hahn auf den Stein B, die Katze auf den Hund (der auf dem Esel auf Steg C ist), und zum Schluss springt der Hahn auf den Rücken der Katze – geschafft!

Dieses Problem, wie ein Stapel Musikanten von A nach C kommt, und dabei nur einen Hilfsplatz B hat, sollt Ihr nun mit einem Java-Programm lösen, allerdings nicht nur für die vier Stadtmusikanten, sondern für eine beliebige Zahl n an Musikanten (Falls Stadt und Land Bremen jemals zu Geld kommen sollten, genügt wohl ein Quartett nicht mehr den künstlerischen Ansprüchen, dann muss es ein richtiges Stadtmusikantenorchester sein...).

Wenn Ihr Euch den Ablauf oben genau anschaut, erkennt Ihr eine rekursive Vorschrift, welche Aktionen systematisch abzulaufen haben.

Benutzt bei Eurer Implementierung für A, B und C 3 `int`-Arrays. Die Grösse der Musikanten (sie sind alle unterschiedlich gross) wird als eine `int`-Zahl angegeben. Der grösste Musikant wird auf den Wert k =Länge des Arrays abgebildet, der nächstkleinere Musikant auf den Wert $k-1$ abgebildet, usw.

Die Spitze eines Musikantenstapels befindet sich jeweils an der Arrayposition 0, auf dem Anfangsturm muß also dort der kleinste Musikant, abgebildet auf die Zahl 1, liegen. Alle Positionen der anderen zwei Stapel sollen mit dem Wert 0 initialisiert werden, der bedeutet, daß sich dort kein Musikant befindet.

¹Grimm, Brüder. Kinder- und Hausmärchen. Vollständige Ausgabe. Mit 184 Illustrationen zeitgenössischer Künstler und einem Nachwort von Heinz Rölleke. S. 180-189. Düsseldorf und Zürich, 19. Auflage 1999.