Prof. Dr. Stefan Brass
Dipl.-Inform. Annett Thüring
Dipl.-Inform. Steffen Schiele
Dipl.-Inform. Andreas Bienert

Martin-Luther-Universität Halle-Wittenberg
Institut für Informatik

Wintersemester 2014/15

Objektorientierte Programmierung: Hausaufgabenblatt 4

Abgabe: 20.11.2014, 12:00 Übung: 24./25.11.2014

Hausaufgabe 4: (4 Theoriepunkte)

Schreiben Sie ein Programm für ein einfaches, eindimensionales "Schiffe versenken" Spiel (für einen Spieler). In diesem Spiel besteht das Spielfeld aus einer Reihe von 25 Kästchen. Der Computer versteckt darin (zufällig) ein "Schlachtschiff" von 5 Kästchen Breite und einen "Zerstörer" von 3 Kästchen Breite. Die Plazierung der Schiffe haben wir schon für Sie programmiert, das Programm ist unten abgedruckt und kann unter folgender URI heruntergeladen werden:

`http://www.informatik.uni-halle.de/~brass/oop14/homework/Schiffe.java`

Das Programm füllt ein `int`-Array der Länge 25 mit "0", falls an der Position Wasser ist, und "1", falls an der Position ein Schiff ist.

Das Spiel besteht nun darin, dass der Spieler die Nummer eines Feldes (von 0 bis 24) eingibt, und das Spiel dann "Treffer" oder "Wasser" druckt (oder "überfluessig", falls das Feld bereits vorher beschossen wurde, und "Fehler", falls die Eingabe außerhalb des Bereiches 0–24 liegt). Ausserdem soll nach jedem Zug der aktuelle Spielstand (das Feld) ausgedruckt werden, natürlich nur soweit, wie es dem Spieler bekannt ist. In der Ausgabe verwenden Sie bitte das Zeichen

- "`.`" für unbekannt (noch nicht beschossen),
- "`-`" für Fehlschuss (Wasser),
- "`X`" für Treffer.

Dazu könnten Sie im `int`-Array außer den oben genannten Codes 0 und 1 noch folgende Codes verwenden:

- 2 für Positionen, an die der Spieler schon "geschossen" hat, die aber "Fehlschüsse" waren (Wasser),
- 3 für Treffer.

Ihr Programm soll erkennen, wenn beide Schiffe "versenkt" sind, d.h. 8 Treffer erzielt wurden. Dann sollen Sie ausgeben "`Gewonnen in n Zuegen.`", wobei das `n` durch die Anzahl Züge zu ersetzen ist (dieses Spiel endet immer damit, dass der Spieler gewinnt).


```
// Hausaufgabe 4: Eindimensionales "Schiffe versenken"-Spiel
// Name: _____

import java.util.Random;
import java.util.Scanner;

class Schiffe {
        public static void main(String[] args) {
                Scanner scan = new Scanner(System.in);
                Random rgen = new Random();
                // Random rgen = new Random(0);

                // Spielfeld: 25 Kaestchen
                final int groesse = 25;
                int feld[] = new int[groesse];
                        // Bedeutung der Werte im Spielfeld:
                        // 0: Wasser - noch nicht beschossen
                        // 1: Schiff - noch nicht beschossen
                        // 2: Wasser - Spieler weiss, das leer
                        // 3: Schiff - schon getroffen

                // Position des Schlachtschiffes (5 breit):
                int schlachtschiff = rgen.nextInt(groesse - 4);
                        // Startposition bei 0 bis 20
                feld[schlachtschiff+0] = 1;
                feld[schlachtschiff+1] = 1;
                feld[schlachtschiff+2] = 1;
                feld[schlachtschiff+3] = 1;
                feld[schlachtschiff+4] = 1;

                // Position des Zerstoerers (3 breit)
                int min = 0;
                int max = groesse - 3;
                if(schlachtschiff <= 3)
                        min = schlachtschiff + 6;
                        // Passt nicht mehr davor
                else if(schlachtschiff >= groesse - 8)
                        // Passt nicht mehr dahinter
                        max = schlachtschiff - 4;
                else if(rgen.nextInt(2) == 0)
                        // Zufallsentscheidung davor/dahinter
                        min = schlachtschiff + 6;
                else
                        max = schlachtschiff - 4;
                int zerstoerer = min + rgen.nextInt(max-min+1);
                feld[zerstoerer+0] = 1;
                feld[zerstoerer+1] = 1;
                feld[zerstoerer+2] = 1;

```

```
                // Ab hier Ihr Programm:
                // Die Testausgabe koennen Sie loeschen:
                int i = 0;
                while(i < feld.length) {
                        if(feld[i] == 0)
                                System.out.print('.');
                        else
                                System.out.print('#');
                        i = i + 1;
                }
                System.out.println();
        }
}
```

Schreiben Sie das Programm in die Datei “`Schiffe.java`”. Geben Sie diese Datei über die Übungsplattform bis zum Donnerstag, 20.11.2014, 12:00 ab. Beachten Sie, dass nur compilierbare Programme akzeptiert werden. Wir verwenden eine automatische Vorkorrektur, deswegen halten Sie sich bitte genau an die Vorgaben. Vor der Abgabe wählen Sie bitte die Zeile mit “`Random(0)`”, so dass der Zufallszahlengenerator auf einen festen Wert initialisiert ist. Bitte achten Sie auch auf sinnvolle Einrückungen und allgemein ein verständliches Programm. Für schlechten Programmierstil können auch Punkte abgezogen werden.

Übungsaufgabe 4A: (ohne Abgabe)

Bitte bearbeiten Sie die Übungsaufgaben auf dem Hausaufgabenblatt, aber geben Sie diese Aufgaben nicht ab. Diese Aufgaben werden in der Übung besprochen. Sie müssen Ihre Lösung eventuell in der Übung vorführen.

Zeichnen Sie für die folgenden Wertausdrücke die Operatorbäume:

a) `i * (j - 2 * k)`

b) `i - 10 > 0 || i < 0`

c) `i - j + k == 0 || i != j && k != 0`

d) `i - i % j * j`

Übungsaufgabe 4B: (ohne Abgabe)

Was gibt das folgende Programm aus? Begründen Sie!

```
public class Ausgabe
{
    public static void main(String[] args)
    {
        int i = 12;
        int j = 5;
        double x = 7.6;

        x = x + i / j;
        System.out.println("x: " + x);
        i = j / 2 * 3;
        System.out.println("i: " + i);
        j = i = i - j % 3;
        System.out.println("j: " + j + "0");
    }
}
```

Auch diese Aufgabe bitte nicht einsenden, aber bis zur Übung bearbeiten.