Exercises for BoS

Stephan Euler
Version 0.1

February 26, 2020

Contents

Chapter 1

Exercises

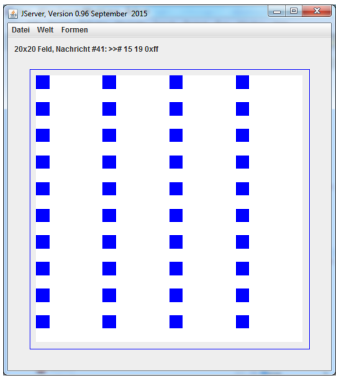
This document contains some exercises for the *Board of Symbols* tool. The order corresponds to the use in an introductory lecture on programming with Java.

1.1 Introduction to BoS

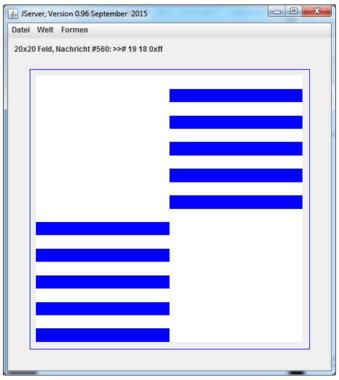
Exercise 1.1.1. Letter
Write code to draw a nice picture with a letter. With very simple letters (*I*, *L*) you should put more effort into the design, e. g. with serifs, shadows or italics. The images will be collected and put online in a gallery.
Hint: first sketch the letter on paper, preferably on a printout with numbered fields.

1.2 Control structures

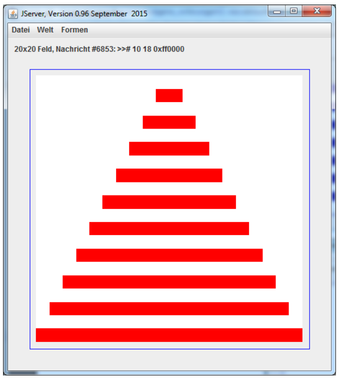
Exercise 1.2.1. Patterns
Write codes that produce the following patterns:



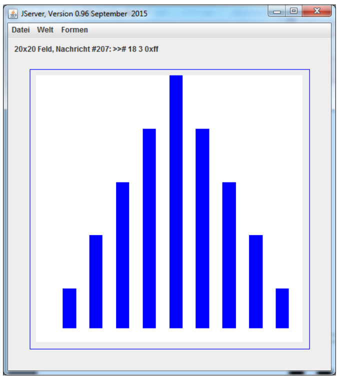
20x20 Feld, Nachricht #41: >># 15 19 0xff



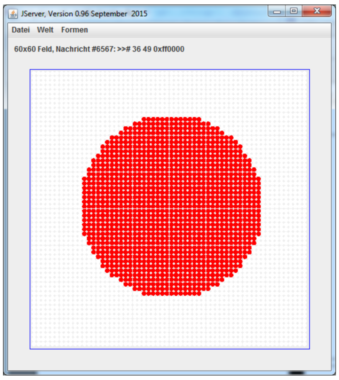
20x20 Feld, Nachricht #560: >># 19 18 0xff



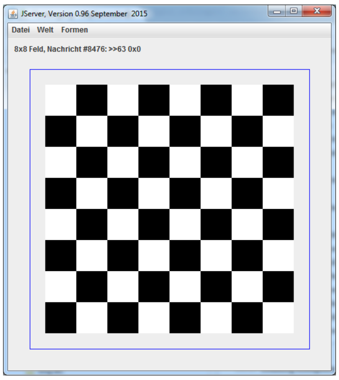
20x20 Feld, Nachricht #6853: >># 10 18 0xff0000



20x20 Feld, Nachricht #207: >># 18 3 0xff



60x60 Feld, Nachricht #6567: >># 36 49 0xff0000



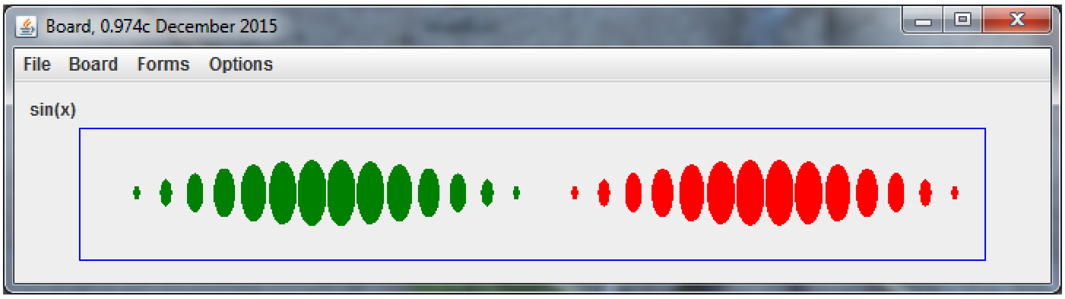
8x8 Feld, Nachricht #8476: >>63 0x0

1.3 Floating point

Exercise 1.3.1. Function
Draw the function $sin(x)$ for values of x and y from 0 to 2π.

- Map the board coordinates $0, 1, \ldots, N-1$ to corresponding values from the range $[0, 2\pi]$.
- Calculate for each value $sin(x)$.
- Negative values should be colored red, positive values green.
- The function returns values between -1 and 1. The size of the symbols should be chosen accordingly. Note that a radius of 0.5 fills the entire field. The program also accepts negative values.

The result could look like this:



sin(x)

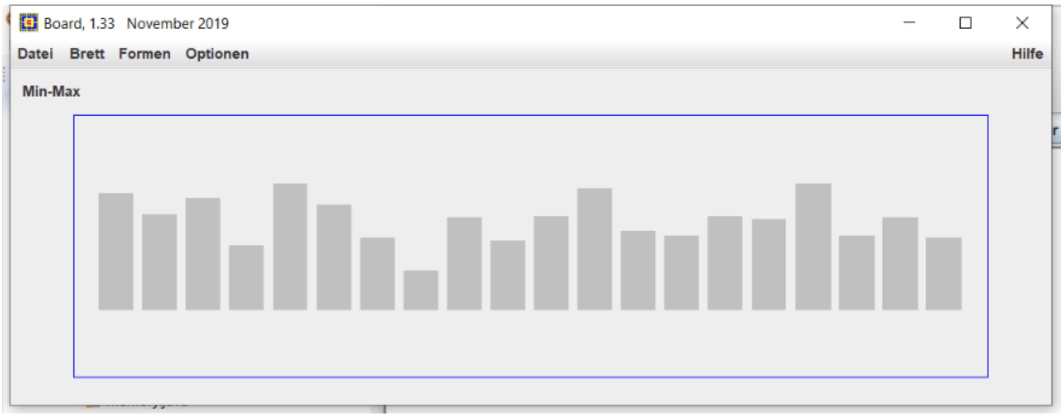
Exercise 1.3.2. Pattern (Voluntary)
Extend the solution to the task1.3.1 to two dimensions and draw the function $sin(x) * cos(y)$ for values of x and y from 0 to 2π.

1.4 Arrays

Exercise 1.4.1. Computing with arrays
In the following program snippet, an array is filled with random numbers between 0.1 and 0.5. Appropriate symbols are set to the respective size.

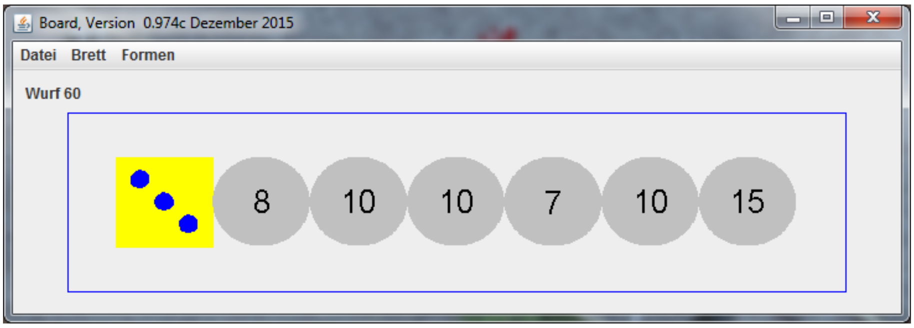
```
double[] a = new double[20];
size( a.length, 1);
forms( "b" );
for( int i=0; i<a.length; i++ ) {
   a[i] = 0.1 + 0.4* Math.random();
   symbolSize( i, a[i] );
}
```

The result should look like this:



- Find the minimum and maximum of the field and mark it in color (e. g. minimum red, maximum green).
- Also output the minimum, maximum and mean value of the field with `print`.

Exercise 1.4.2. Array as counter
Implement a simulation of a die. Based on the method `Math.random ()` generate the numbers between 1 and 6. Count the number of times each number occurs in an array of appropriate size and output the result with `print`. Display the counted frequencies displayed with BoS. Use the BoS method `text()` in the form `text(i, "Text")` to write text to symbols. One solution:

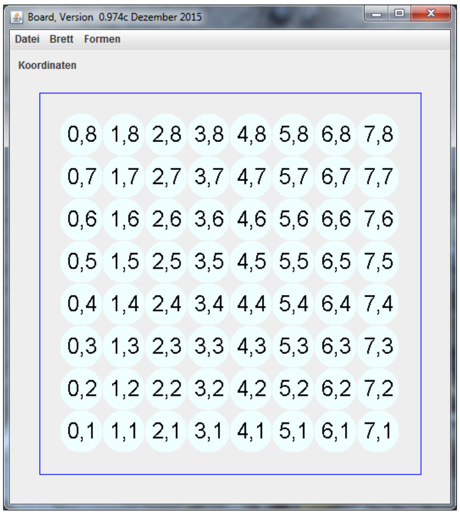


Wurf 60

8 10 10 7 10 15

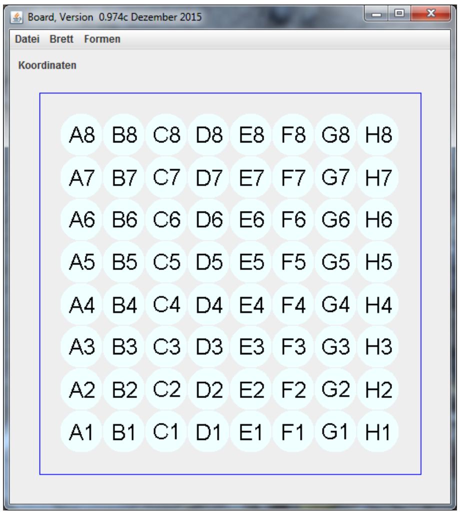
1.5 Strings

Exercise 1.5.1. String pattern
Write codes that produce the following patterns:



Koordinaten

0,8	1,8	2,8	3,8	4,8	5,8	6,8	7,8
0,7	1,7	2,7	3,7	4,7	5,7	6,7	7,7
0,6	1,6	2,6	3,6	4,6	5,6	6,6	7,6
0,5	1,5	2,5	3,5	4,5	5,5	6,5	7,5
0,4	1,4	2,4	3,4	4,4	5,4	6,4	7,4
0,3	1,3	2,3	3,3	4,3	5,3	6,3	7,3
0,2	1,2	2,2	3,2	4,2	5,2	6,2	7,2
0,1	1,1	2,1	3,1	4,1	5,1	6,1	7,1



Koordinaten

A8	B8	C8	D8	E8	F8	G8	H8
A7	B7	C7	D7	E7	F7	G7	H7
A6	B6	C6	D6	E6	F6	G6	H6
A5	B5	C5	D5	E5	F5	G5	H5
A4	B4	C4	D4	E4	F4	G4	H4
A3	B3	C3	D3	E3	F3	G3	H3
A2	B2	C2	D2	E2	F2	G2	H2
A1	B1	C1	D1	E1	F1	G1	H1

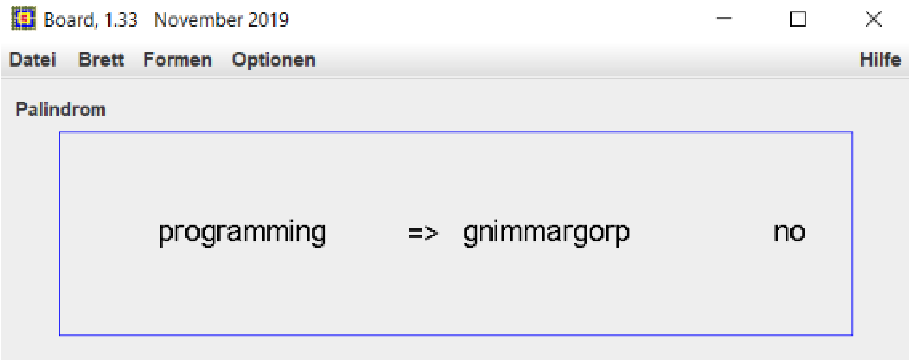
Use the BoS methods `text` or `text2` to write strings to the corresponding positions.

Exercise 1.5.2. Interactive
In BoS you can with

```
String text = Dialogs.askString( "Palindrom-Test" );
```

read strings.

1. Use this method to write a palindrome tester. The text entered should be reversed (`THM` becomes `MHT`) and checked, whether the two texts match (e.g. textttanna). Optionally, you can convert the input to lower case and remove spaces. The result should be displayed in BoS in the form



```
programming      =>  gnimmargorp        no
```

2. Furthermore, a pattern should be repeated until a predetermined length is reached. Use

   ```
   int n = Dialogs.askInteger( "length" );
   ```

   to ask for the length. Then, by repeating the pattern, a string of a given length is generated. Using the pattern `THM` and three examples for `n` results in:

   ```
   n: 3  THM
   n: 5  THMTH
   n: 9  THMTHMTHM
   ```

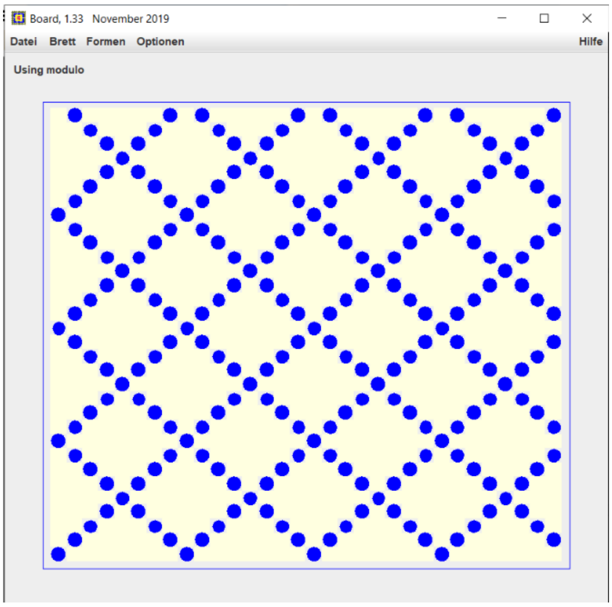
Chapter 2

Quiz

2.1 Patterns

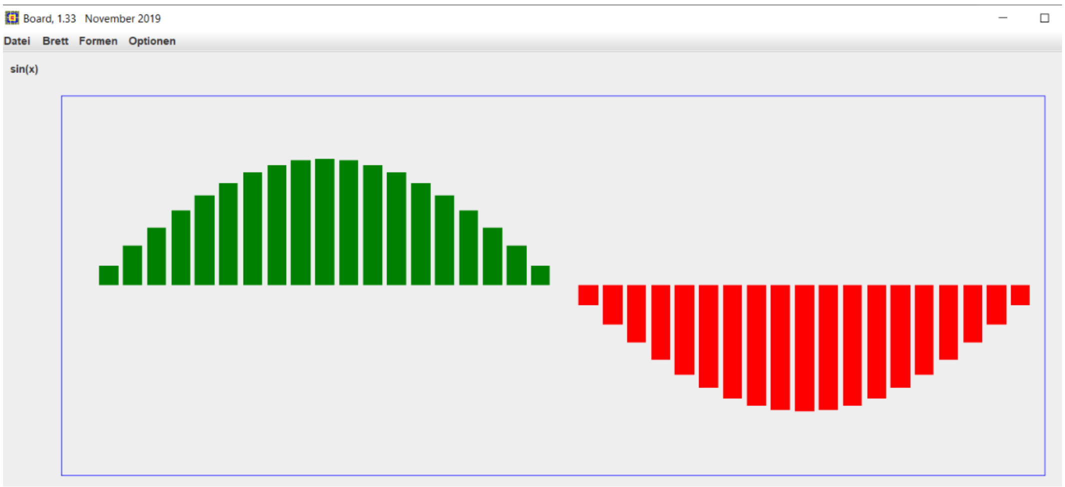
Can you create the following patterns?

Exercise 2.1.1. Modulo

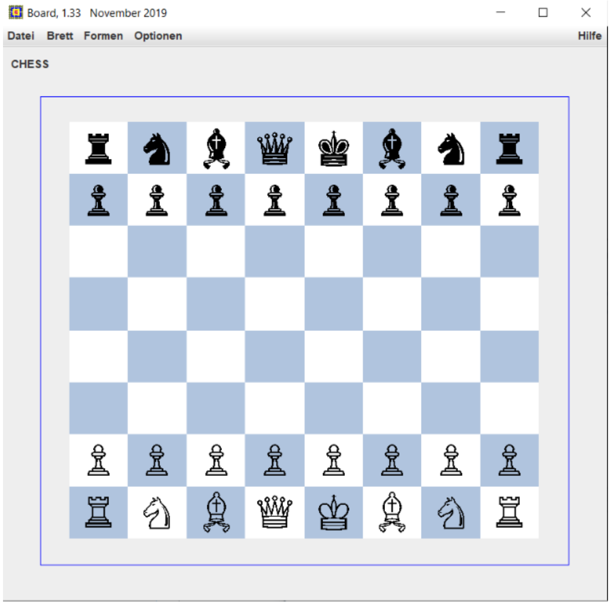


Using modulo

Exercise 2.1.2. Sinus



Exercise 2.1.3. Chess


Board, 1.33 November 2019
Datei
Brett
Formen
Optionen
Hilfe
CHESS