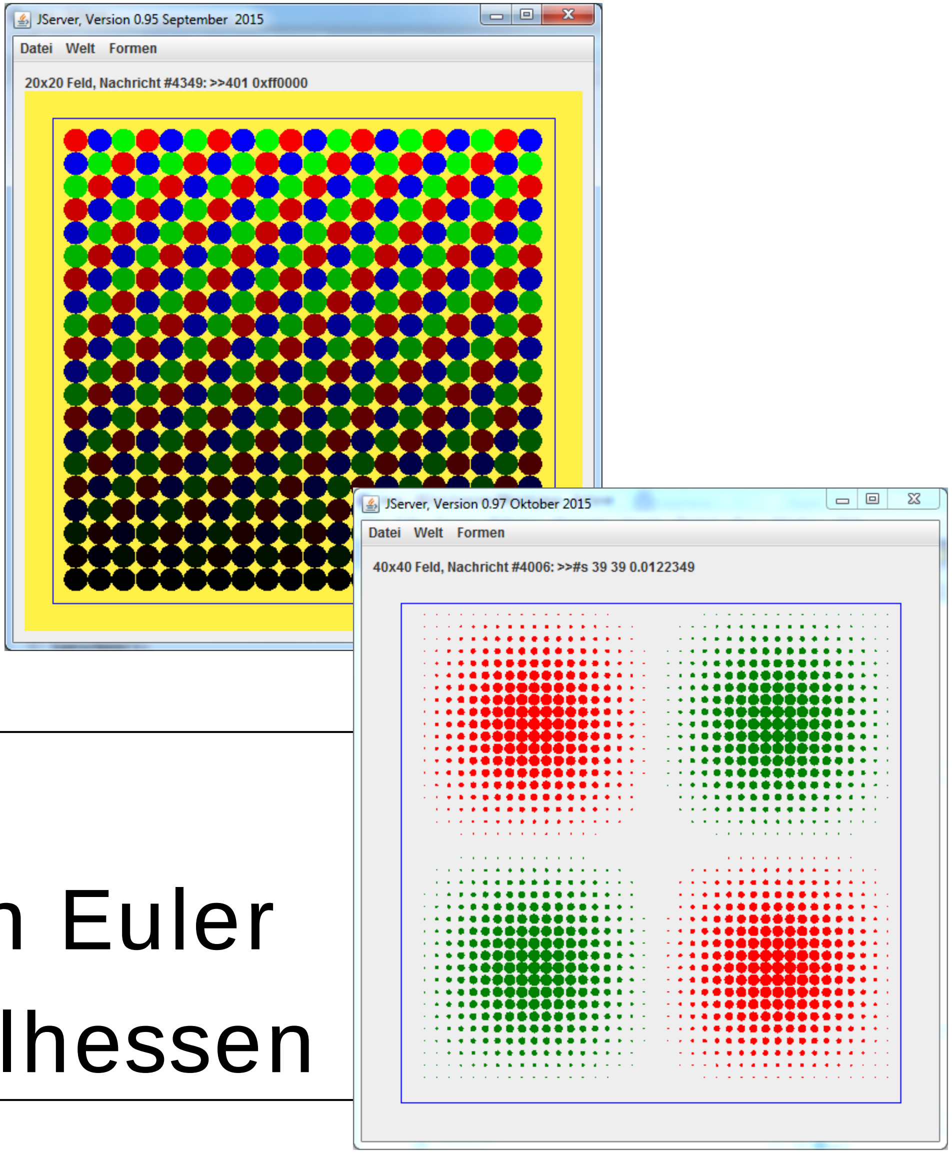
Board of Symbols



JServer, Version 0.95 September 2015

20x20 Feld, Nachricht #4349: >>401 0xff0000

JServer, Version 0.97 Oktober 2015

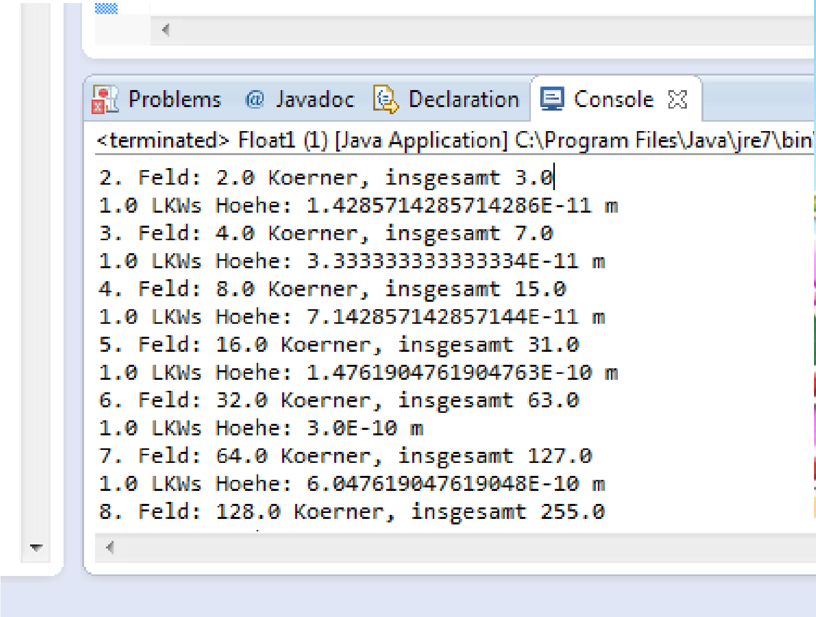
40x40 Feld, Nachricht #4006: >>#s 39 39 0.0122349

Stephan Euler

TH Mittelhessen

TECHNISCHE HOCHSCHULE MITTELHESSEN

Programming:
possibilities and expectations



<terminated> Float1 (1) [Java Application] C:\Program Files\Java\jre7\bin

```
2. Feld: 2.0 Koerner, insgesamt 3.0
1.0 LKWs Hoehe: 1.4285714285714286E-11 m
3. Feld: 4.0 Koerner, insgesamt 7.0
1.0 LKWs Hoehe: 3.3333333333333334E-11 m
4. Feld: 8.0 Koerner, insgesamt 15.0
1.0 LKWs Hoehe: 7.142857142857144E-11 m
5. Feld: 16.0 Koerner, insgesamt 31.0
1.0 LKWs Hoehe: 1.4761904761904763E-10 m
6. Feld: 32.0 Koerner, insgesamt 63.0
1.0 LKWs Hoehe: 3.0E-10 m
7. Feld: 64.0 Koerner, insgesamt 127.0
1.0 LKWs Hoehe: 6.047619047619048E-10 m
8. Feld: 128.0 Koerner, insgesamt 255.0
```

Programming

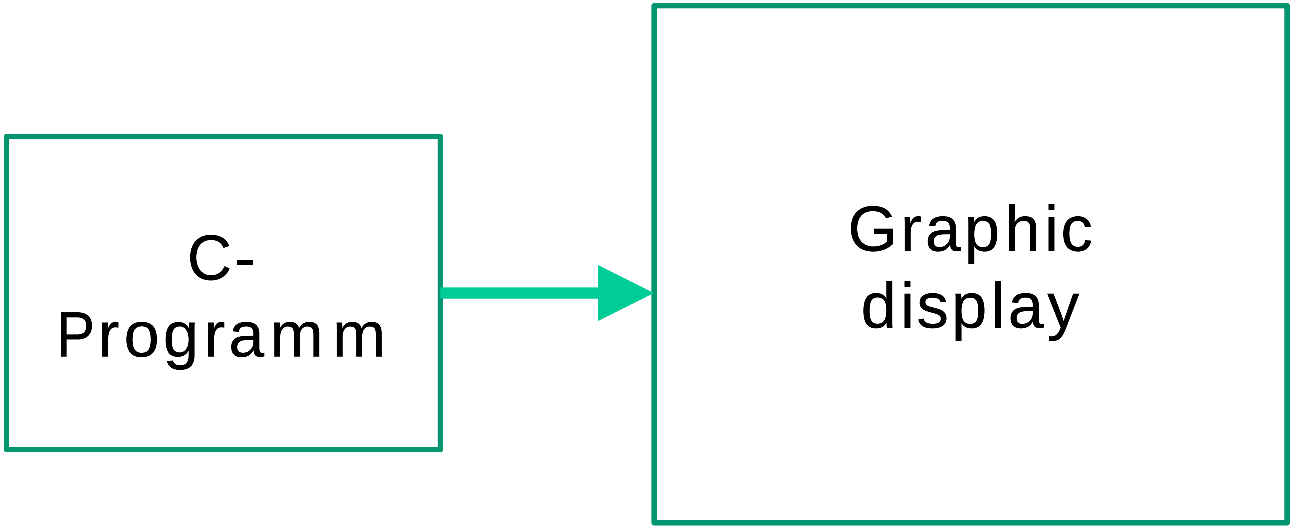
Experience from lectures:

- Many students have problems with programming
- Low motivation

Improvement

- Start with graphical output
- Not only text output in console window

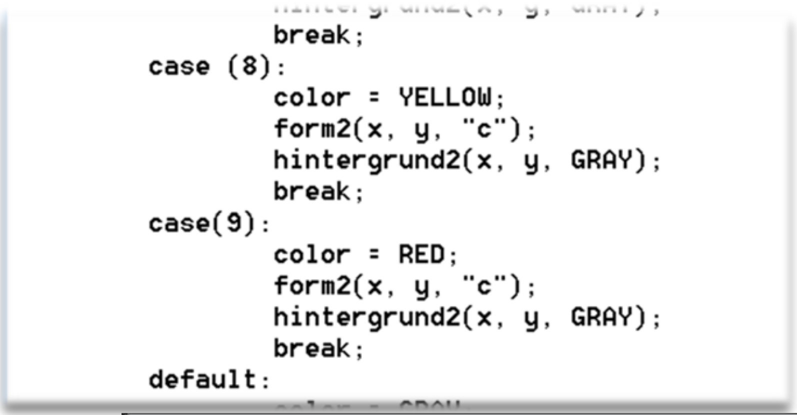
First idea



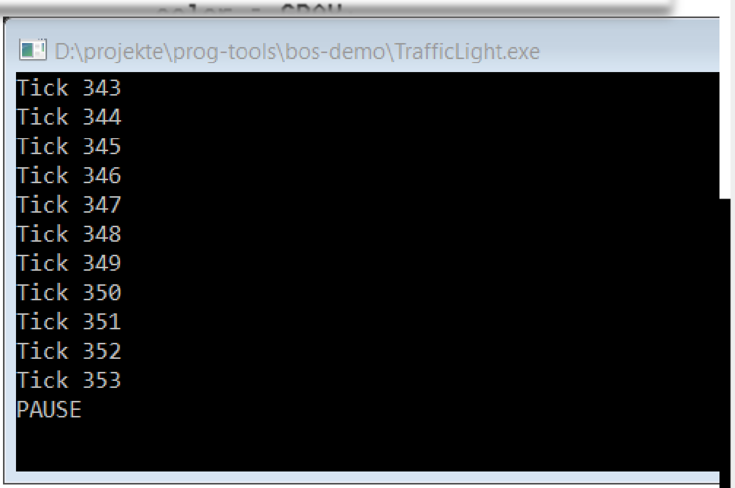
C-Programm

Graphic display

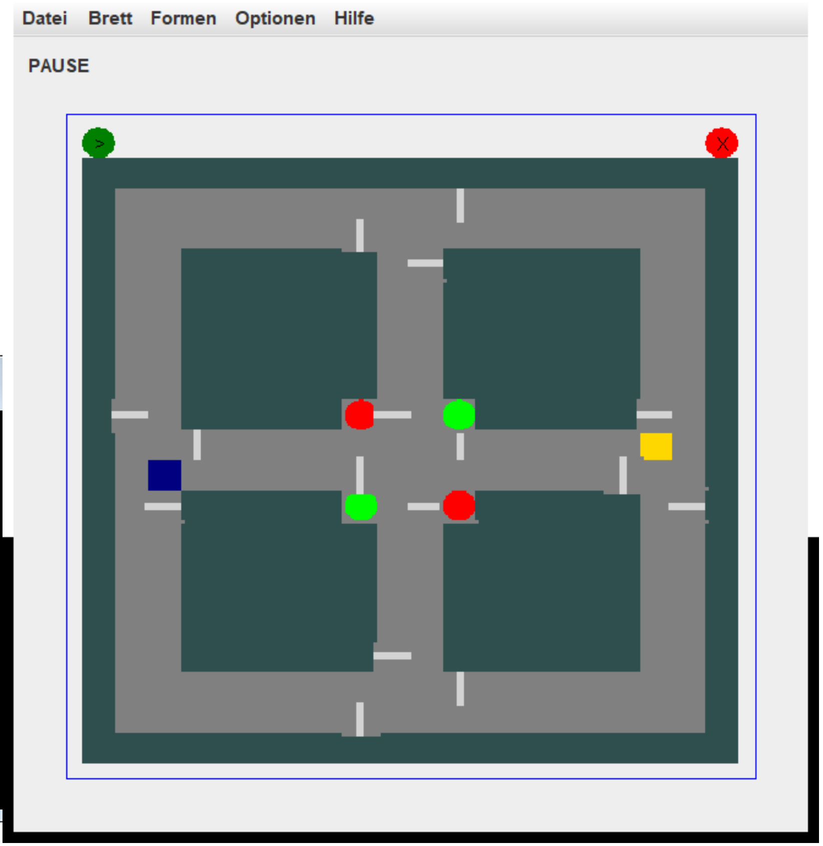
What we have now



```
                hintergrund2(x, y, GRAY);
                break;
        case (8):
                color = YELLOW;
                form2(x, y, "c");
                hintergrund2(x, y, GRAY);
                break;
        case(9):
                color = RED;
                form2(x, y, "c");
                hintergrund2(x, y, GRAY);
                break;
        default:
                color = GRAY;
```



```
Tick 343
Tick 344
Tick 345
Tick 346
Tick 347
Tick 348
Tick 349
Tick 350
Tick 351
Tick 352
Tick 353
PAUSE
```



Requirements

- Easy to use
- Accompany users from an simple entry to fairly challenging projects
- Room for creativity
- Java-based
- Interface to C

Concept

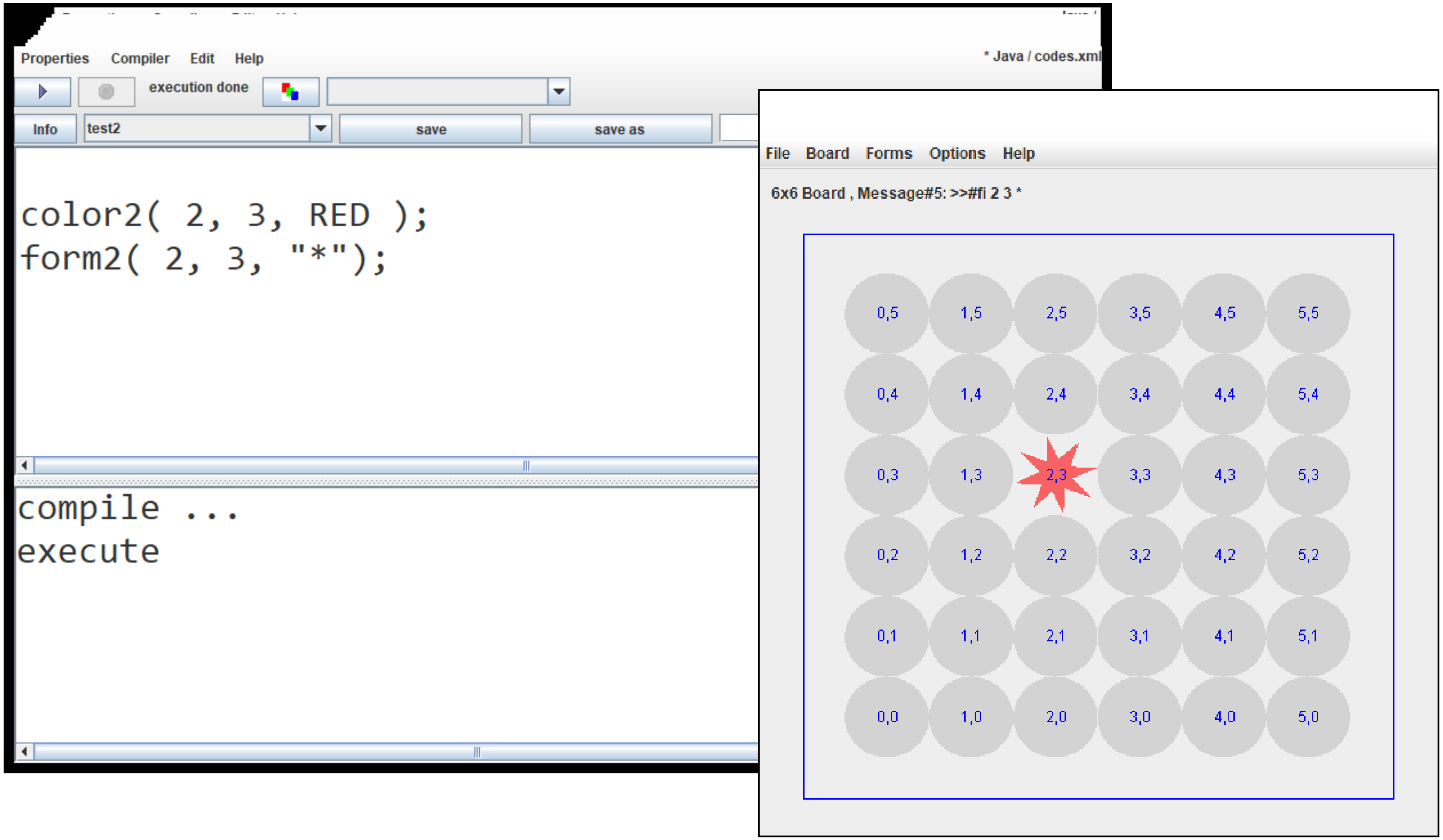
- Board with N x M squares (similar to board in games such as chess or checkers)
- Symbols are placed on squares
- Symbols have
  - Shape
  - Color
  - Background color
  - Size
  - Text content
  - …
- Functions to change symbol properties:

```
    color2( x, y, BLUE )
```

Snippet mode

- Build-in editor
- Provides color chooser and function templates
- Adds boilerplate code e. g. to complete Java class with main method
- Stores and retrieves code snippets (XML file)

Example snippet



```
color2( 2, 3, RED );
form2( 2, 3, "*");
```

```
compile ...
execute
```

6x6 Board , Message#5: >>#fi 2 3 *

0,5	1,5	2,5	3,5	4,5	5,5
0,4	1,4	2,4	3,4	4,4	5,4
0,3	1,3	2,3	3,3	4,3	5,3
0,2	1,2	2,2	3,2	4,2	5,2
0,1	1,1	2,1	3,1	4,1	5,1
0,0	1,0	2,0	3,0	4,0	5,0

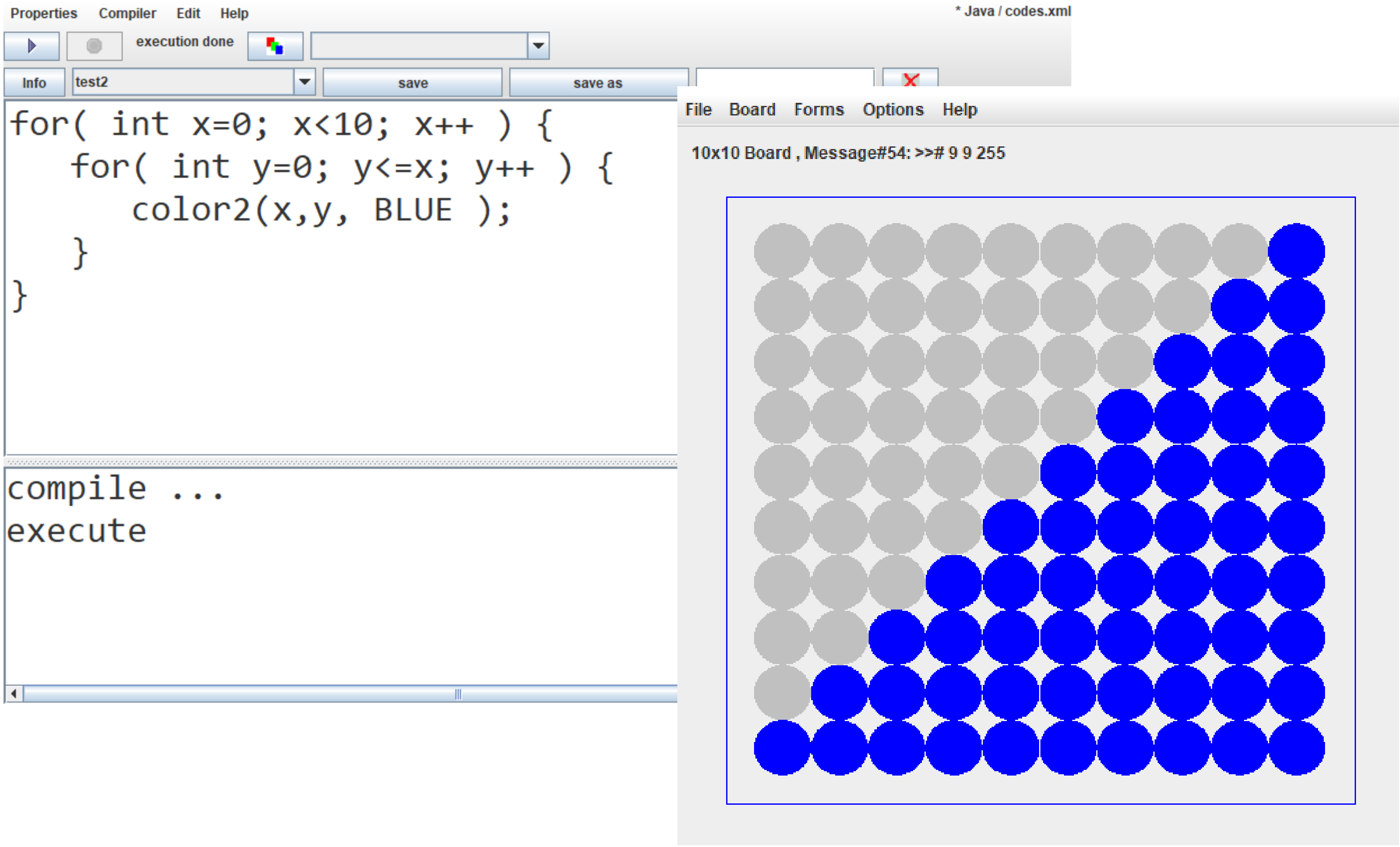
Where we use BoS

Code snippets	Variables, data types, operands
Trainer	Control structures
	Arrays, methods
	algorithms
Application mode	GUI

time

First task: program a letter

Visualisation of loops



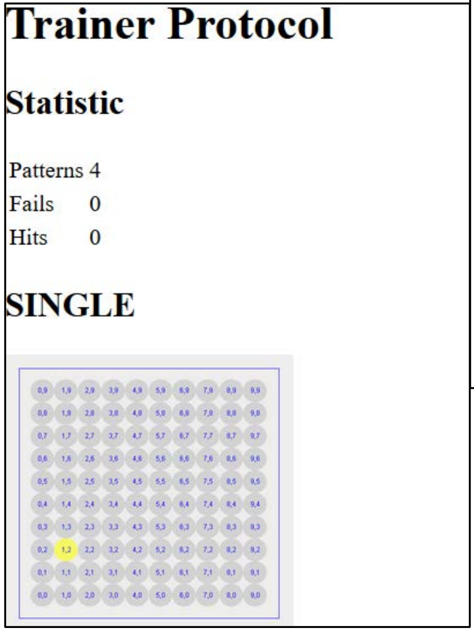
```
for( int x=0; x<10; x++ ) {
   for( int y=0; y<=x; y++ ) {
      color2(x,y, BLUE );
   }
}
```

```
compile ...
execute
```

10x10 Board , Message#54: >># 9 9 255

Trainer

- Generates pattern
- Different levels
- Checks solution
- Protocol



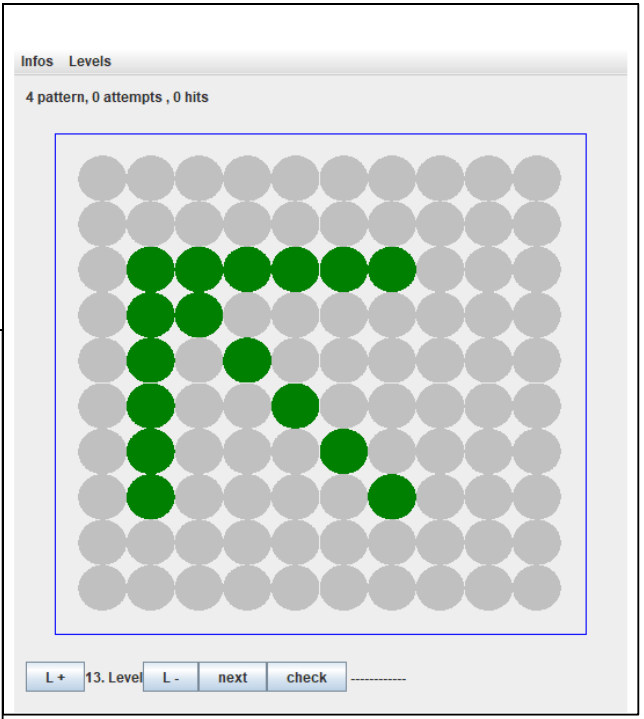
Trainer Protocol

Statistic

Patterns	4
Fails	0
Hits	0

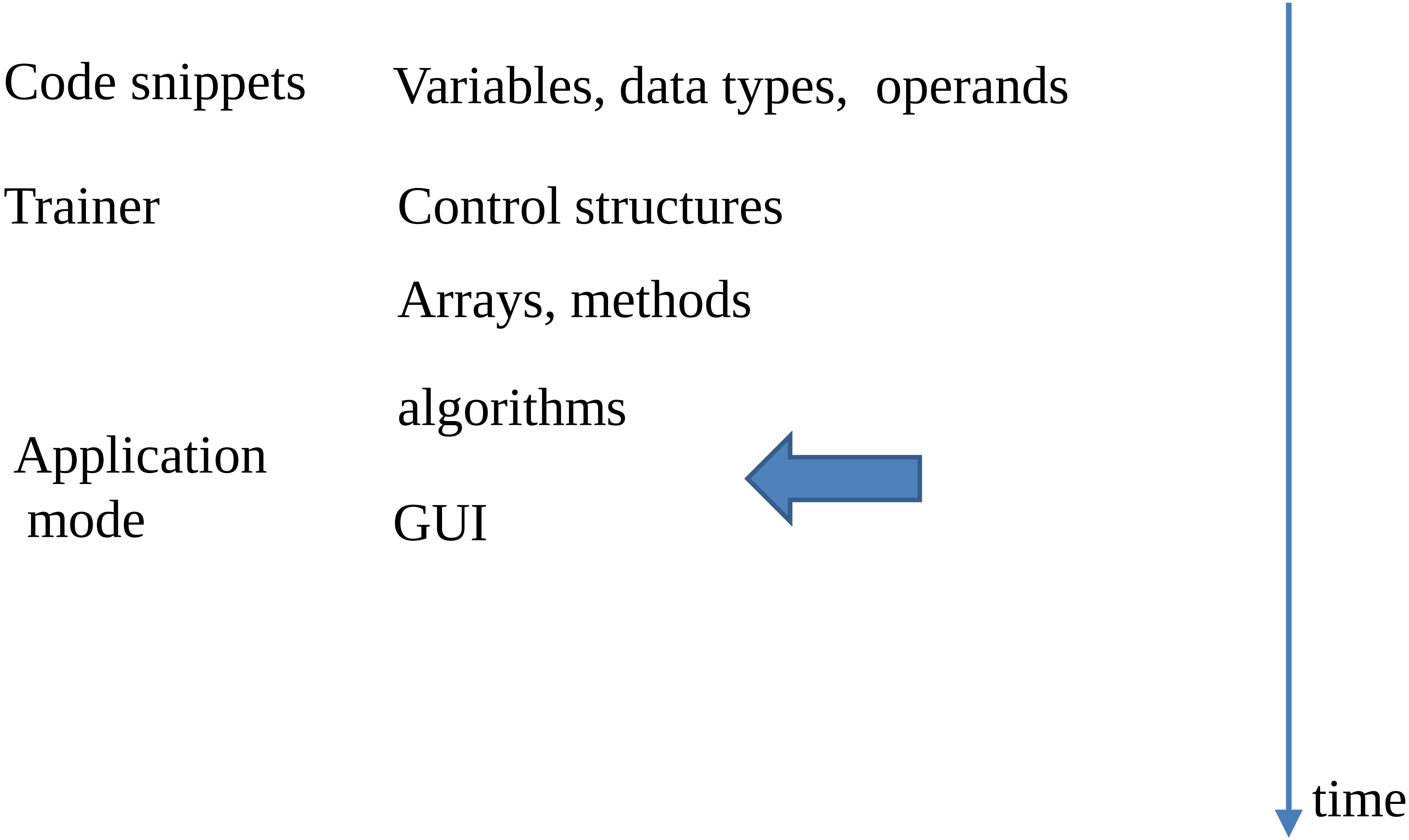
SINGLE

0,9	1,9	2,9	3,9	4,9	5,9	6,9	7,9	8,9	9,9
0,8	1,8	2,8	3,8	4,8	5,8	6,8	7,8	8,8	9,8
0,7	1,7	2,7	3,7	4,7	5,7	6,7	7,7	8,7	9,7
0,6	1,6	2,6	3,6	4,6	5,6	6,6	7,6	8,6	9,6
0,5	1,5	2,5	3,5	4,5	5,5	6,5	7,5	8,5	9,5
0,4	1,4	2,4	3,4	4,4	5,4	6,4	7,4	8,4	9,4
0,3	1,3	2,3	3,3	4,3	5,3	6,3	7,3	8,3	9,3
0,2	1,2	2,2	3,2	4,2	5,2	6,2	7,2	8,2	9,2
0,1	1,1	2,1	3,1	4,1	5,1	6,1	7,1	8,1	9,1
0,0	1,0	2,0	3,0	4,0	5,0	6,0	7,0	8,0	9,0



4 pattern, 0 attempts , 0 hits

Where we use BoS



Code snippets

Variables, data types, operands

Trainer

Control structures

Arrays, methods

algorithms

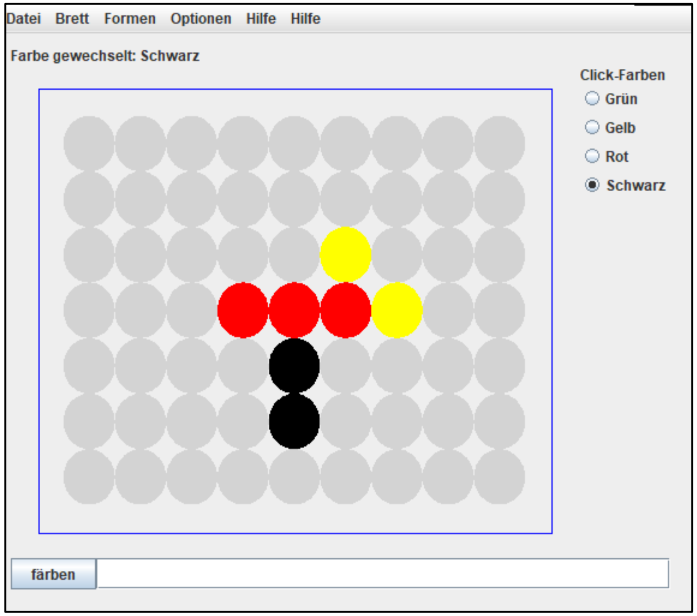
Application mode

GUI

time

Application mode

- Switch to IDE (eclipse)
- Same API methods as snippets
- Add GUI elements such as labels, buttons, …
- Example board painter



Farbe gewechselt: Schwarz

Click-Farben

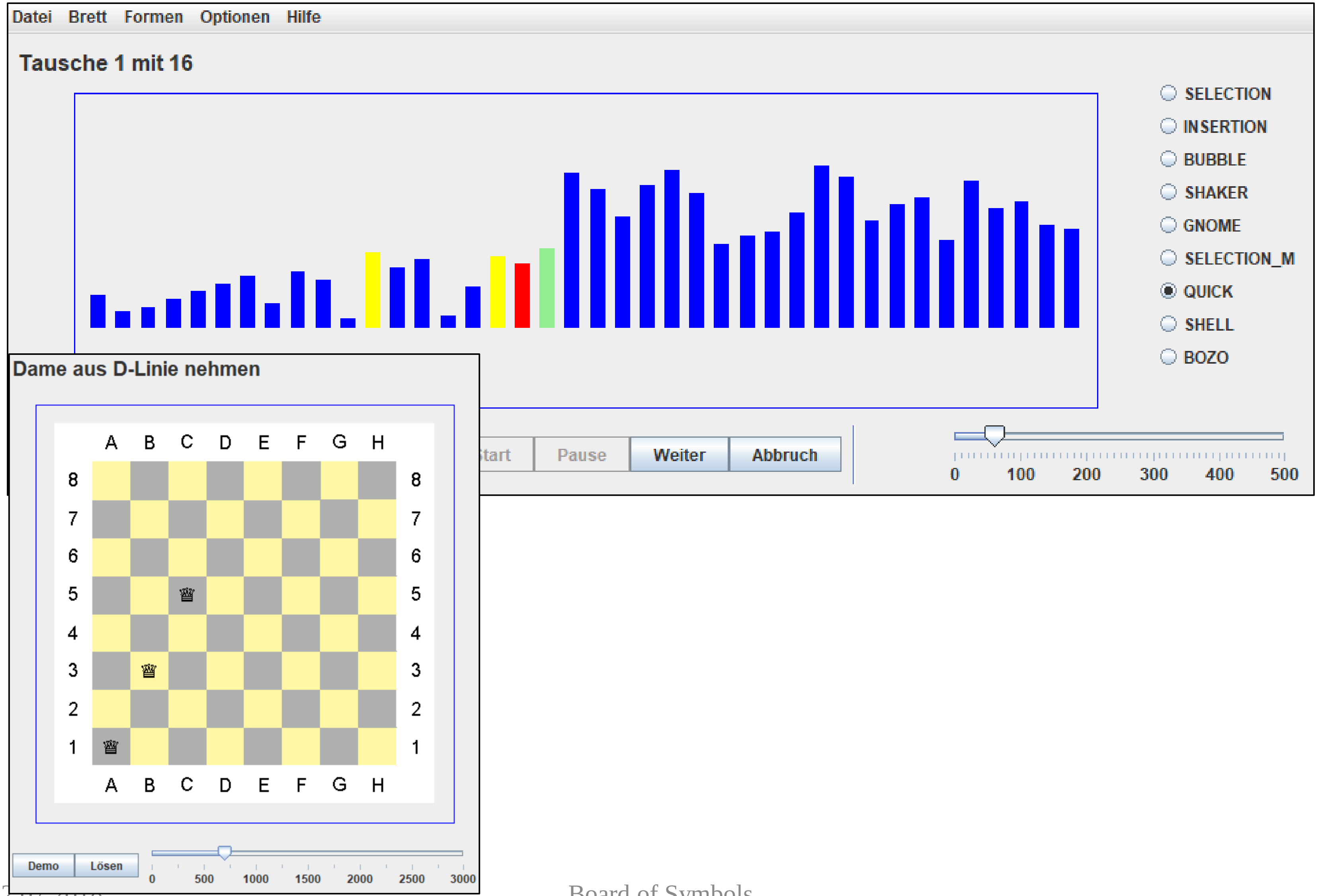
○ Grün

○ Gelb

○ Rot

● Schwarz

Visualisation of algorithms



Tausche 1 mit 16

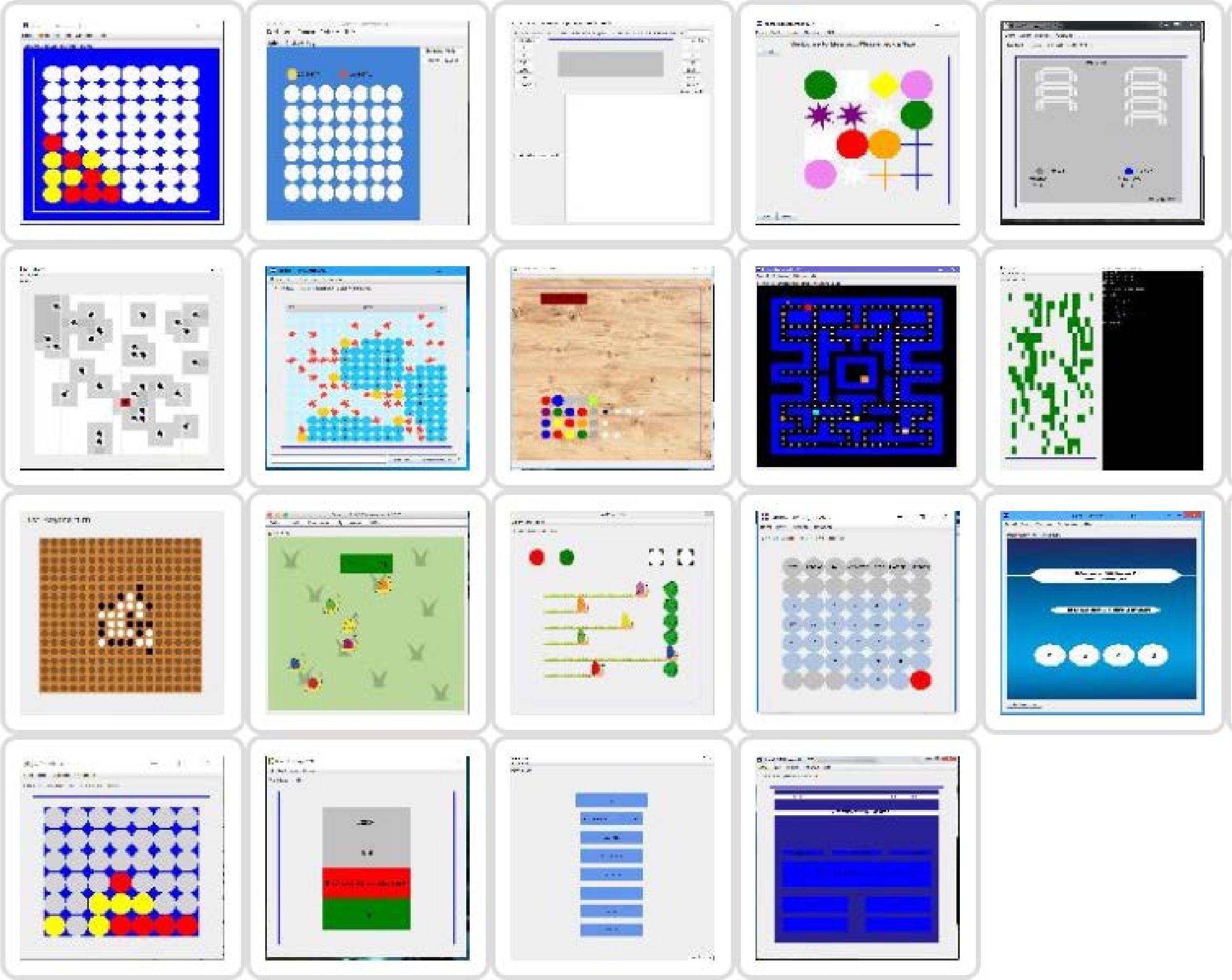
○ SELECTION
○ INSERTION
○ BUBBLE
○ SHAKER
○ GNOME
○ SELECTION_M
● QUICK
○ SHELL
○ BOZO

Dame aus D-Linie nehmen

Projects

- Games:
  4 in a row, snake, Nim, mine sweeper, …
- Simulations:
  Game of Life, traffic lights, …
- Algorithms:
  Sorting, N-queens problem, …

Projects

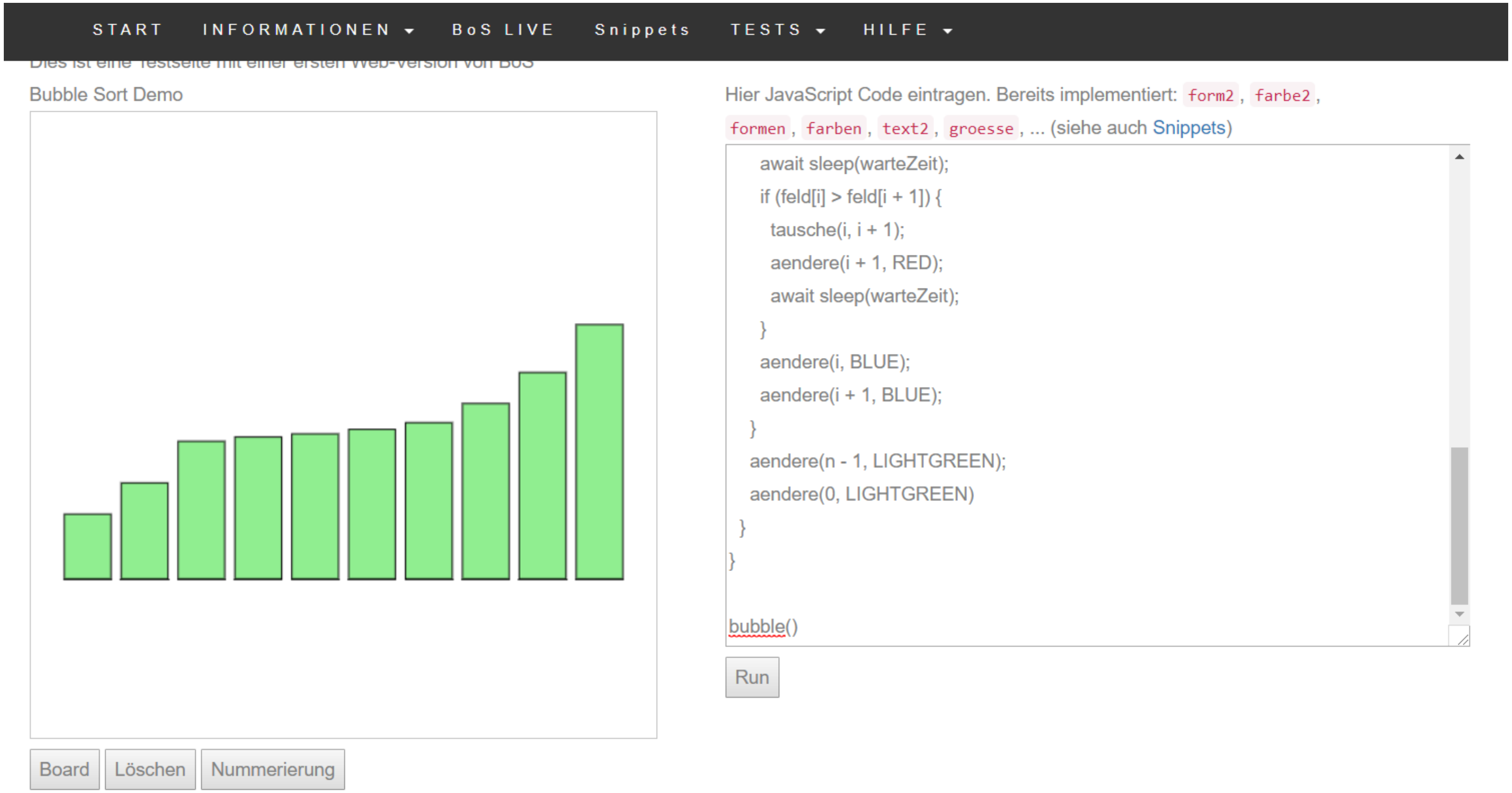


Experiences

- ☺
  - Quick start
  - Provides room for creativity
  - Nice projects
  - Protection against plagiarism
- ☹
  - Performance problems with increasing number of symbols
  - C: problems with different operating systems / versions

Further work

- Online Version



Dies ist eine Testseite mit einer ersten Web-Version von BoS

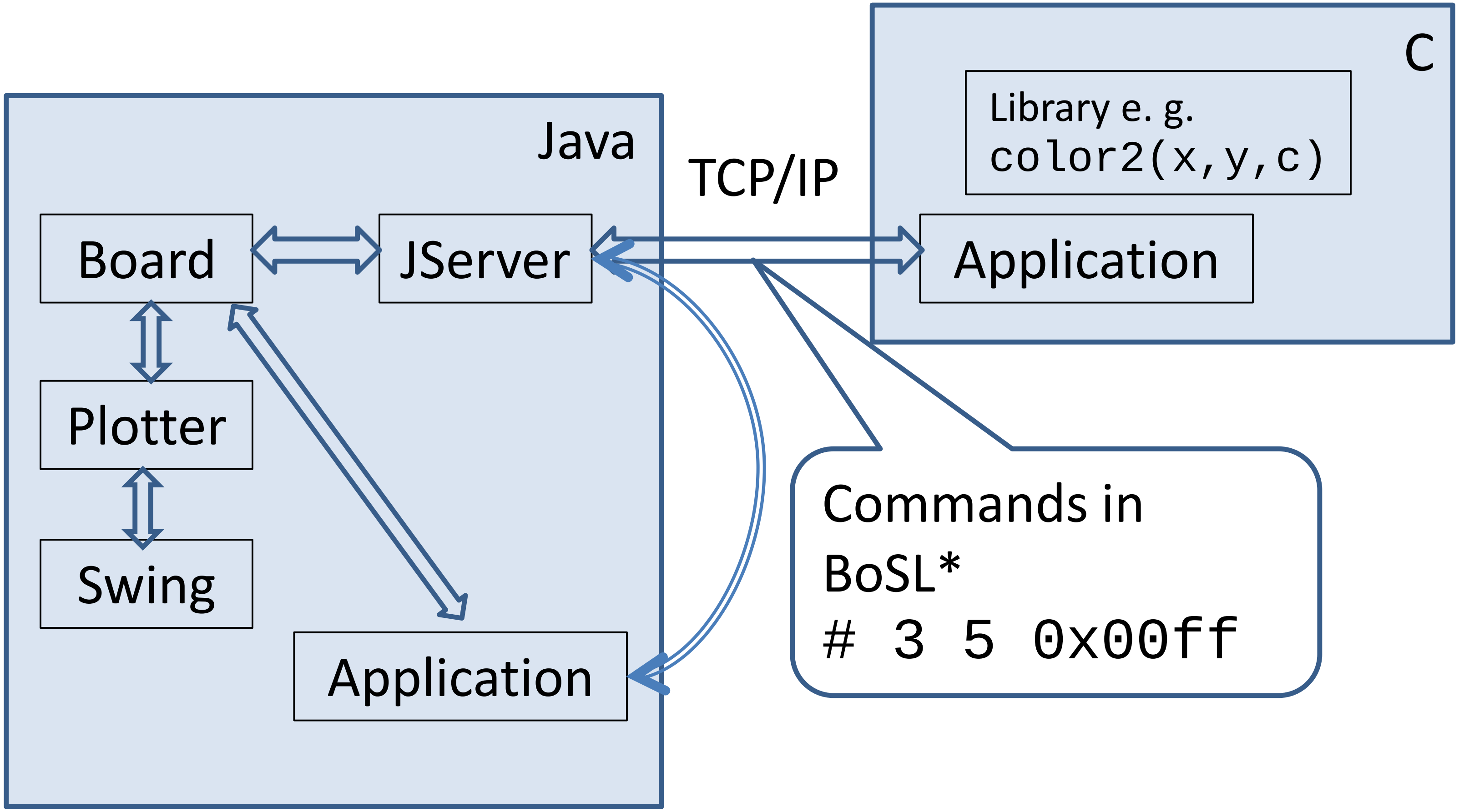
Bubble Sort Demo

Hier JavaScript Code eintragen. Bereits implementiert: `form2`, `farbe2`, `formen`, `farben`, `text2`, `groesse`, ... (siehe auch Snippets)

```
      await sleep(warteZeit);
      if (feld[i] > feld[i + 1]) {
        tausche(i, i + 1);
        aendere(i + 1, RED);
        await sleep(warteZeit);
      }
      aendere(i, BLUE);
      aendere(i + 1, BLUE);
    }
    aendere(n - 1, LIGHTGREEN);
    aendere(0, LIGHTGREEN)
  }
}


bubble()
```

Architecture



Java

Board

JServer

Plotter

Swing

Application

TCP/IP

C

Library e. g.

```
color2(x,y,c)
```

Application

Commands in BoSL*

```
# 3 5 0x00ff
```

* BoS Language