

Fachhochschule Bielefeld

Campus Minden

Objektorientierte Programmierung

Prof. J.Rexilius

Wael Eskeif, 22.12.2022

Inhalt:

9. Aufgabe

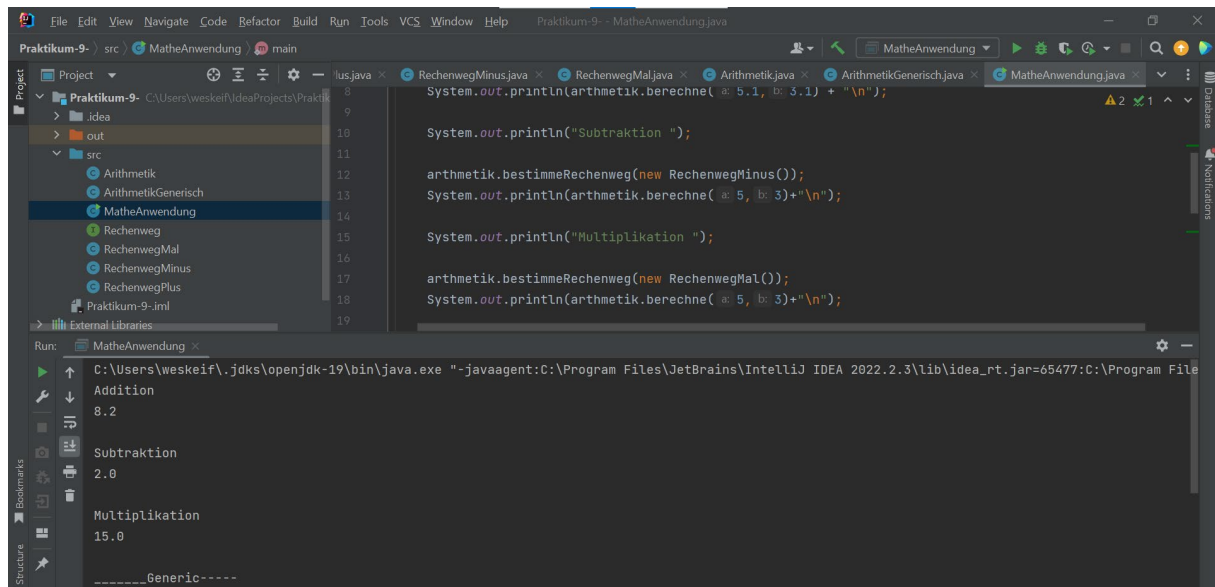
(a). Arithmetik -2-

(b). OOPCode -4-

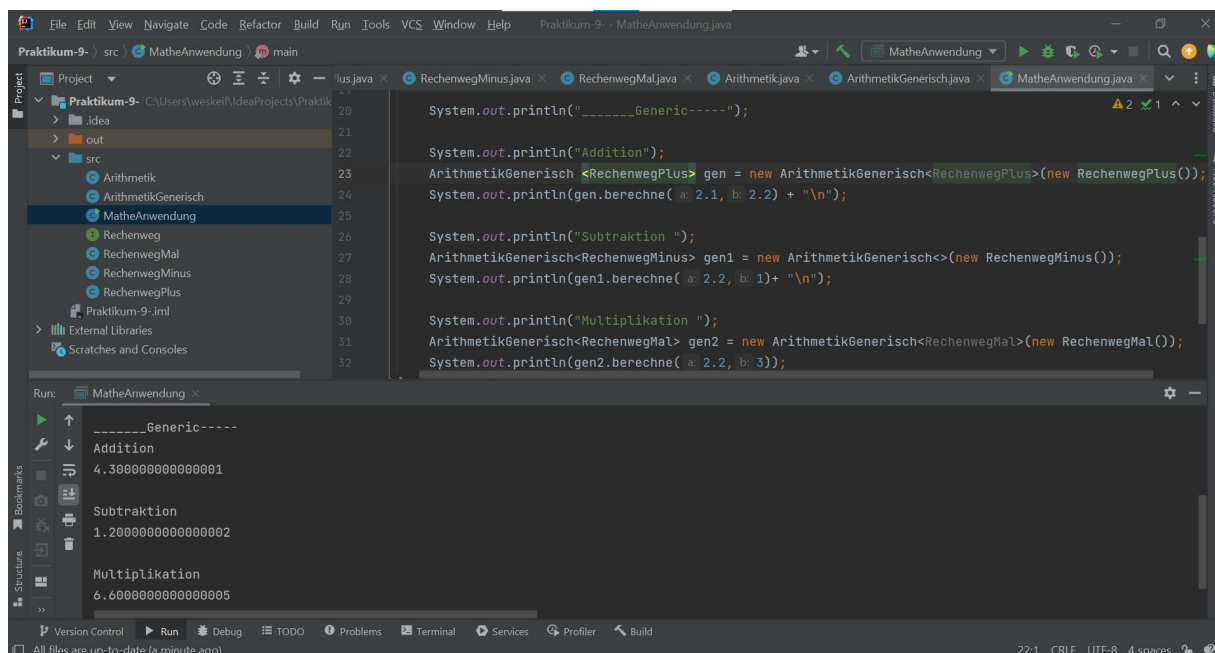
(c). Tic Tac Toe -6-

(a). Arithmetik

Beispiel 1:



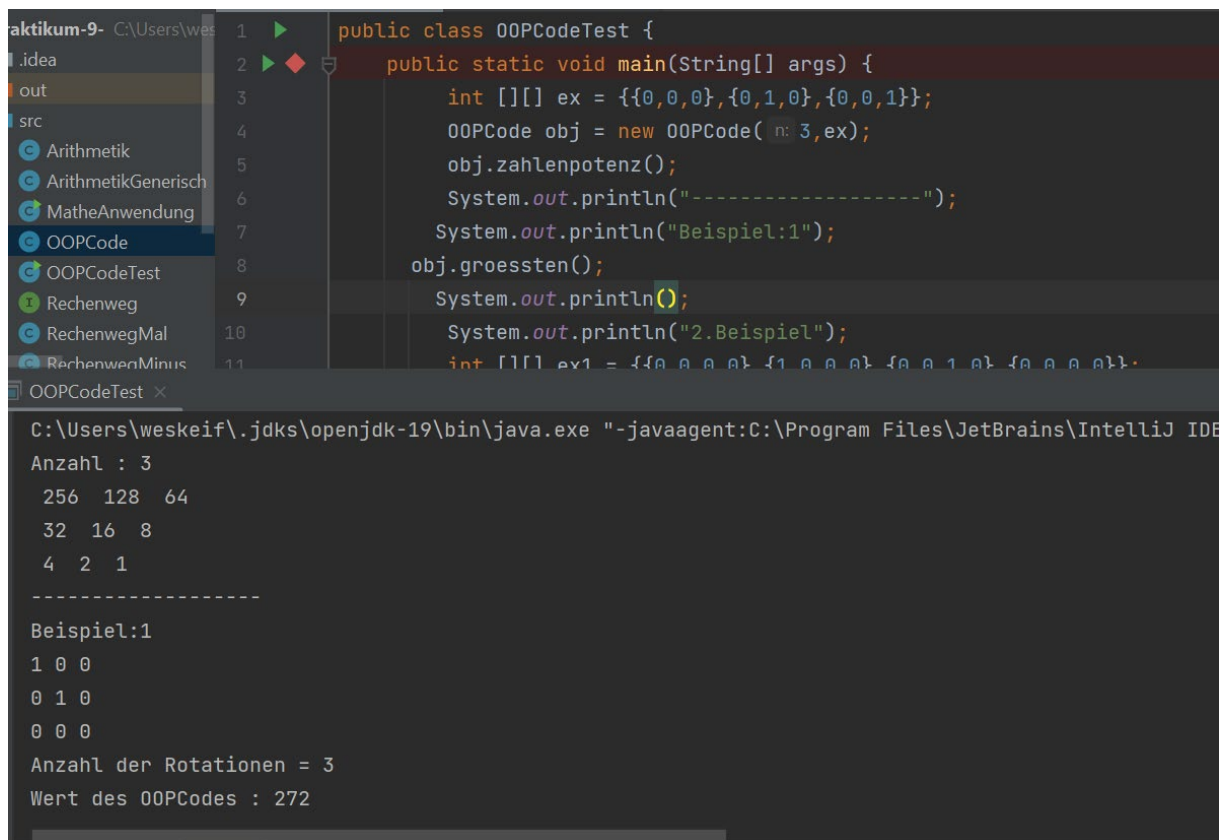
Beispiel 2:



Wael Eskif, 22.12.2022

(b). OOPCode

Beispiel_1 : n=3



The screenshot shows the IntelliJ IDEA IDE with a project named 'Praktikum-9'. The left sidebar displays a file tree with the following structure:

- src
 - Arithmetik
 - ArithmetikGenerisch
 - MatheAnwendung
 - OOPCode
 - OOPCodeTest
 - Rechenweg
 - RechenwegMal
 - RechenwegMinus

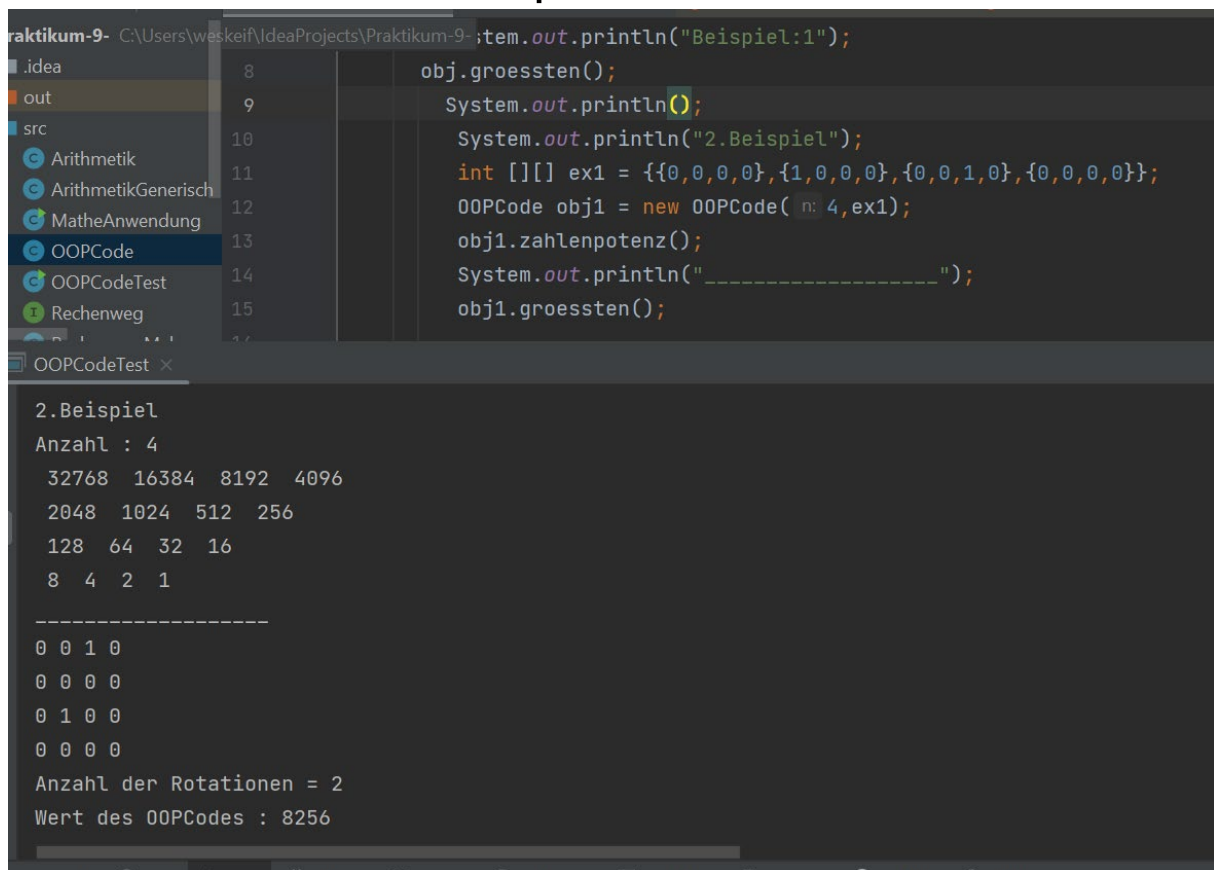
The main editor displays the code for `OOPCodeTest.java`:

```
1 public class OOPCodeTest {  
2     public static void main(String[] args) {  
3         int [][] ex = {{0,0,0},{0,1,0},{0,0,1}};  
4         OOPCode obj = new OOPCode( n: 3,ex);  
5         obj.zahlenpotenz();  
6         System.out.println("-----");  
7         System.out.println("Beispiel:1");  
8         obj.groessten();  
9         System.out.println();  
10        System.out.println("2.Beispiel");  
11        int [][] ex1 = {{0,0,0},{1,0,0},{0,1,0},{0,0,1}};
```

The bottom panel shows the execution output:

```
C:\Users\weskeif\.jdk\openjdk-19\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDE  
Anzahl : 3  
256 128 64  
32 16 8  
4 2 1  
-----  
Beispiel:1  
1 0 0  
0 1 0  
0 0 0  
Anzahl der Rotationen = 3  
Wert des OOPCodes : 272
```

Beispiel-2 : n=4



The screenshot shows an IDE with a project named 'Praktikum-9'. The left sidebar displays a file tree with folders like '.idea', 'out', and 'src', and files including 'Arithmetik', 'ArithmetikGenerisch', 'MatheAnwendung', 'OOPCode', 'OOPCodeTest', and 'Rechenweg'. The main editor window shows the following Java code:

```
1: System.out.println("Beispiel:1");
2:
3:
4:
5:
6:
7:
8: obj.groessten();
9: System.out.println();
10: System.out.println("2.Beispiel");
11: int [][] ex1 = {{0,0,0,0},{1,0,0,0},{0,0,1,0},{0,0,0,0}};
12: OOPCode obj1 = new OOPCode( n: 4,ex1);
13: obj1.zahlenpotenz();
14: System.out.println("-----");
15: obj1.groessten();
```

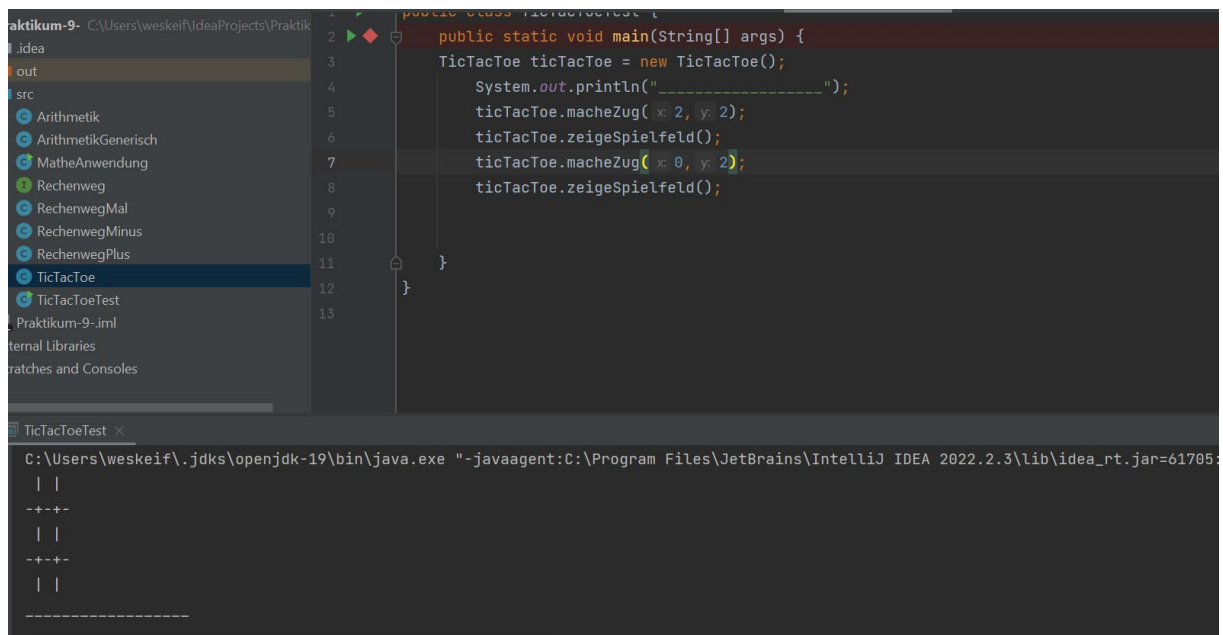
The output window below the code shows the following results:

```
2.Beispiel
Anzahl : 4
32768 16384 8192 4096
2048 1024 512 256
128 64 32 16
8 4 2 1

-----
0 0 1 0
0 0 0 0
0 1 0 0
0 0 0 0
Anzahl der Rotationen = 2
Wert des OOPCodes : 8256
```

(C). Tic Tac Toe

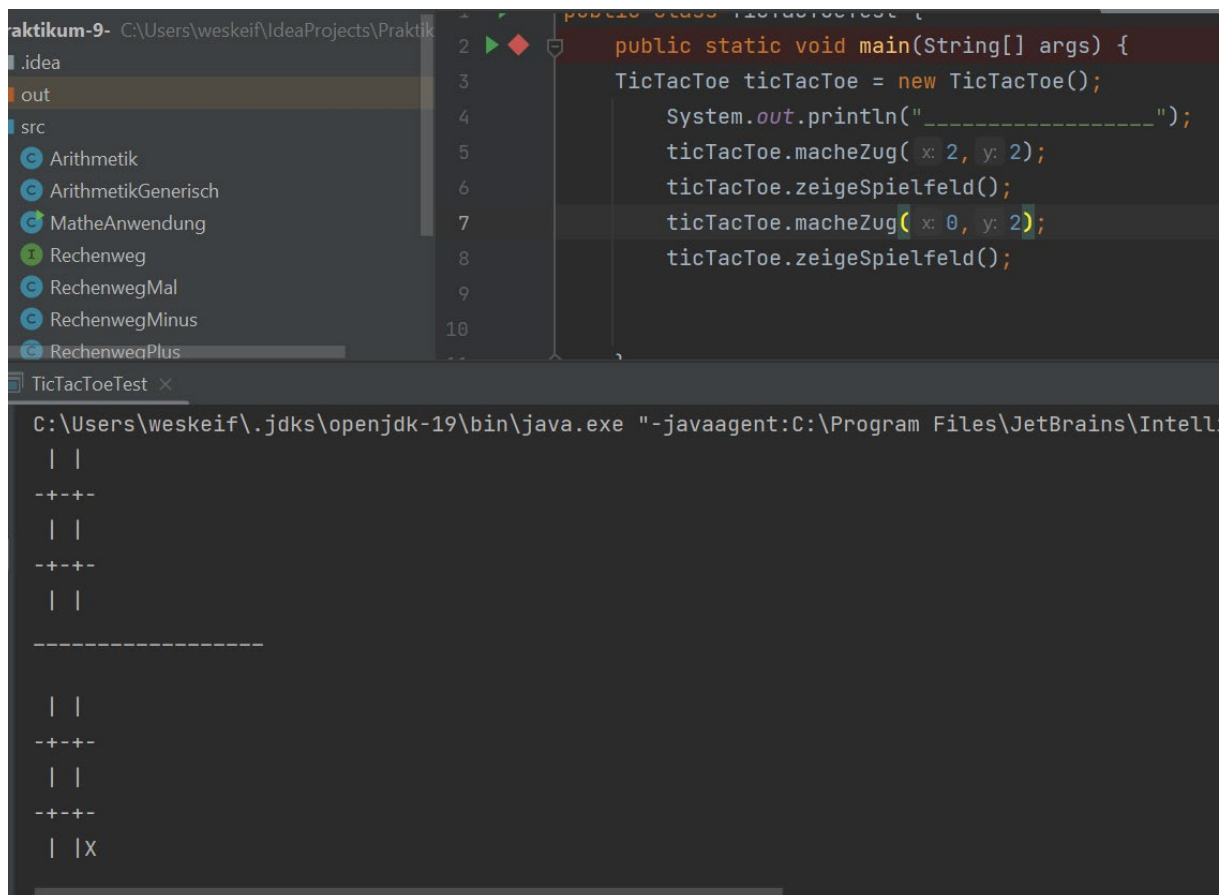
Beispiel: Initial



```
1 public class TicTacToeTest {
2     public static void main(String[] args) {
3         TicTacToe ticTacToe = new TicTacToe();
4         System.out.println("-----");
5         ticTacToe.macheZug(x: 2, y: 2);
6         ticTacToe.zeigeSpielfeld();
7         ticTacToe.macheZug(x: 0, y: 2);
8         ticTacToe.zeigeSpielfeld();
9     }
10 }
11
12 }
13 }
```

```
TicTacToeTest x
C:\Users\weskeif\.jdk\openjdk-19\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2022.2.3\lib\idea_rt.jar=61705:
| |
-+-+
| |
-+-+
| |
-----
```

Beispiel:macheZug(2,2)

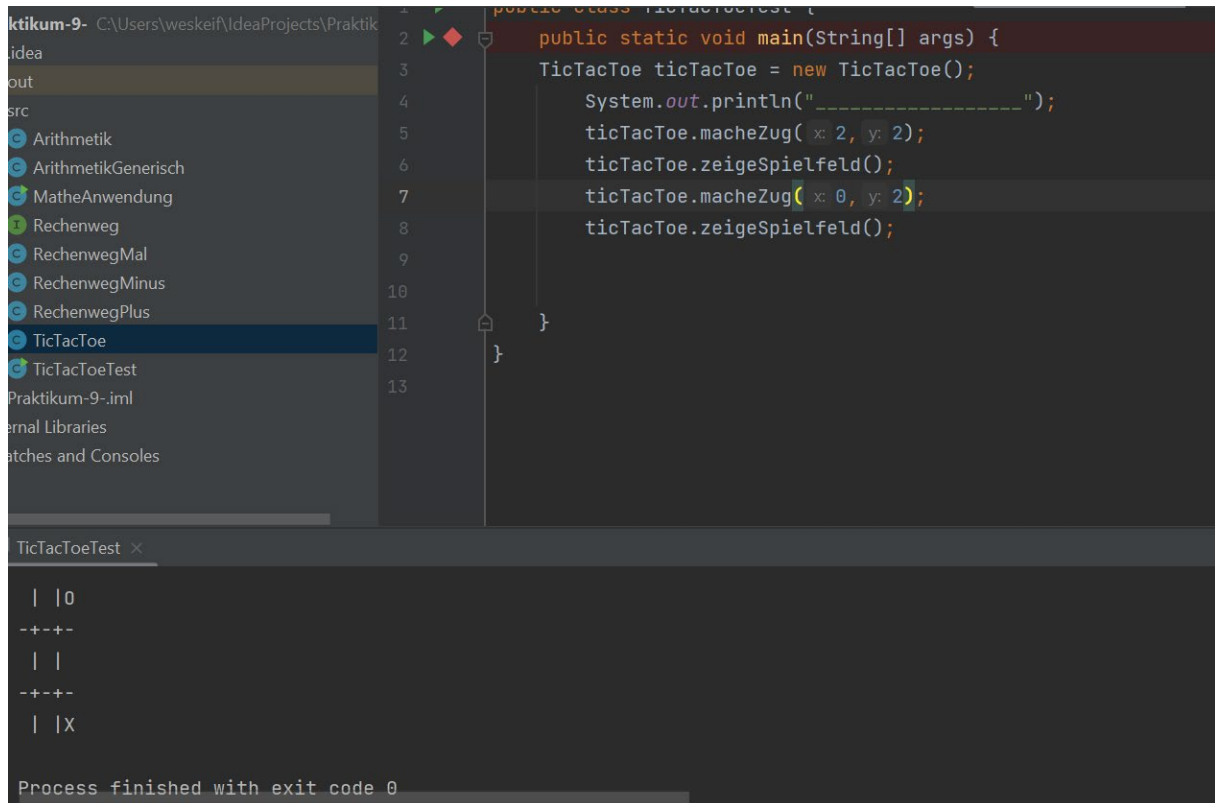


```
1 public class TicTacToeTest {
2     public static void main(String[] args) {
3         TicTacToe ticTacToe = new TicTacToe();
4         System.out.println("-----");
5         ticTacToe.macheZug(x: 2, y: 2);
6         ticTacToe.zeigeSpielfeld();
7         ticTacToe.macheZug(x: 0, y: 2);
8         ticTacToe.zeigeSpielfeld();
9     }
10 }
11
12 }
13 }
```

```
TicTacToeTest x
C:\Users\weskeif\.jdk\openjdk-19\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA 2022.2.3\lib\idea_rt.jar=61705:
| |
-+-+
| |
-+-+
| |
-----

| |
-+-+
| |
-+-+
| |X
```

Beispiel : macheZug(2,0)



The screenshot shows an IDE with a project named 'Praktikum-9'. The left sidebar lists several files, with 'TicTacToeTest' selected. The main editor displays the following Java code:

```
1 public class TicTacToeTest {  
2     public static void main(String[] args) {  
3         TicTacToe ticTacToe = new TicTacToe();  
4         System.out.println("-----");  
5         ticTacToe.macheZug(x: 2, y: 2);  
6         ticTacToe.zeigeSpielfeld();  
7         ticTacToe.macheZug(x: 0, y: 2);  
8         ticTacToe.zeigeSpielfeld();  
9     }  
10 }  
11  
12  
13 }
```

Below the code editor, the 'TicTacToeTest' output window shows the following text:

```
| |0  
--+-  
| |  
--+-  
| |X  
  
Process finished with exit code 0
```