Lab Report

EXERCISE 4: PROGRAMMING IN THE SMALL

	Name	Titel des Kurses	Datum
Juri Wiechmann	571085	Prof. Dr. Weber-Wulff Info 2 Group 2	14.11.2019

Index

Introduction

This week’s exercise was about reviewing the basic logic that you needed. It was shorter than the other labs, which is why we worked alone this week. A websites was given to us on which we had to solve some exercices.

Assignments

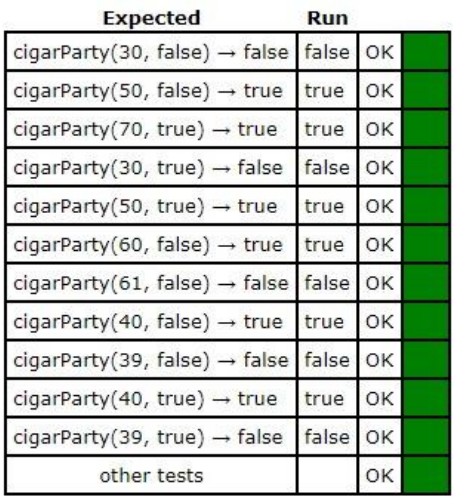
1. **Choose three exercises from the simple logic puzzles. Record the resulting code in your report(S1).**

So I chose:

1. cigarParty:

When squirrels get together for a party, they like to have cigars. A squirrel party is successful when the number of cigars is between 40 and 60, inclusive. Unless it is the weekend, in which case there is no upper bound on the number of cigars. Return true if the party with the given values is successful, or false otherwise.

```
public boolean cigarParty(int cigars, boolean isWeekend) {
  if(cigars <= 60 && cigars >= 40 && !isWeekend)
    return true;
  else if(cigars >= 40 && isWeekend)
    return true;
  else
    return false;
}
```

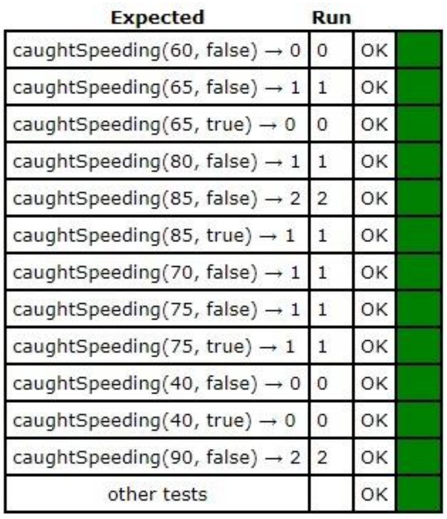


Expected	Run		
cigarParty(30, false) → false	false	OK	🟩
cigarParty(50, false) → true	true	OK	🟩
cigarParty(70, true) → true	true	OK	🟩
cigarParty(30, true) → false	false	OK	🟩
cigarParty(50, true) → true	true	OK	🟩
cigarParty(60, false) → true	true	OK	🟩
cigarParty(61, false) → false	false	OK	🟩
cigarParty(40, false) → true	true	OK	🟩
cigarParty(39, false) → false	false	OK	🟩
cigarParty(40, true) → true	true	OK	🟩
cigarParty(39, true) → false	false	OK	🟩
other tests		OK	🟩

2. CaughtSpeeding:

You are driving a little too fast, and a police officer stops you. Write code to compute the result, encoded as an int value: 0=no ticket, 1=small ticket, 2=big ticket. If speed is 60 or less, the result is 0. If speed is between 61 and 80 inclusive, the result is 1. If speed is 81 or more, the result is 2. Unless it is your birthday -- on that day, your speed can be 5 higher in all cases.

```
public int caughtSpeeding(int speed, boolean isBirthday) {
  speed = ((isBirthday)? speed -5 : speed);
  if(speed <=60) return 0;
  else if(speed > 60 && speed <= 80) return 1;
  else return 2;
}
```

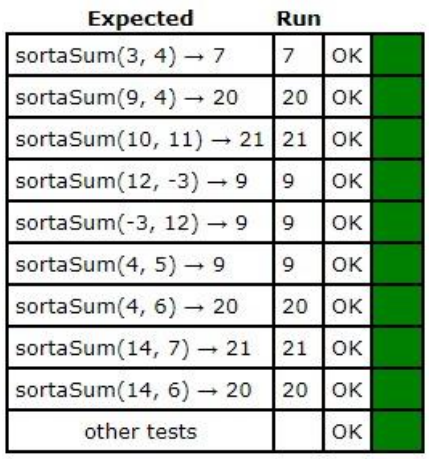


Expected	Run		
caughtSpeeding(60, false) → 0	0	OK	🟩
caughtSpeeding(65, false) → 1	1	OK	🟩
caughtSpeeding(65, true) → 0	0	OK	🟩
caughtSpeeding(80, false) → 1	1	OK	🟩
caughtSpeeding(85, false) → 2	2	OK	🟩
caughtSpeeding(85, true) → 1	1	OK	🟩
caughtSpeeding(70, false) → 1	1	OK	🟩
caughtSpeeding(75, false) → 1	1	OK	🟩
caughtSpeeding(75, true) → 1	1	OK	🟩
caughtSpeeding(40, false) → 0	0	OK	🟩
caughtSpeeding(40, true) → 0	0	OK	🟩
caughtSpeeding(90, false) → 2	2	OK	🟩
other tests		OK	🟩

3. sortaSum:

Given 2 ints, a and b, return their sum. However, sums in the range 10..19 inclusive, are forbidden, so in that case just return 20.

```
public int sortaSum(int a, int b) {
  if(a+b < 21 && a+b > 9  ) return 20;
  else return a+b;
}
```



Expected	Run		
sortaSum(3, 4) → 7	7	OK	🟩
sortaSum(9, 4) → 20	20	OK	🟩
sortaSum(10, 11) → 21	21	OK	🟩
sortaSum(12, -3) → 9	9	OK	🟩
sortaSum(-3, 12) → 9	9	OK	🟩
sortaSum(4, 5) → 9	9	OK	🟩
sortaSum(4, 6) → 20	20	OK	🟩
sortaSum(14, 7) → 21	21	OK	🟩
sortaSum(14, 6) → 20	20	OK	🟩
other tests		OK	🟩

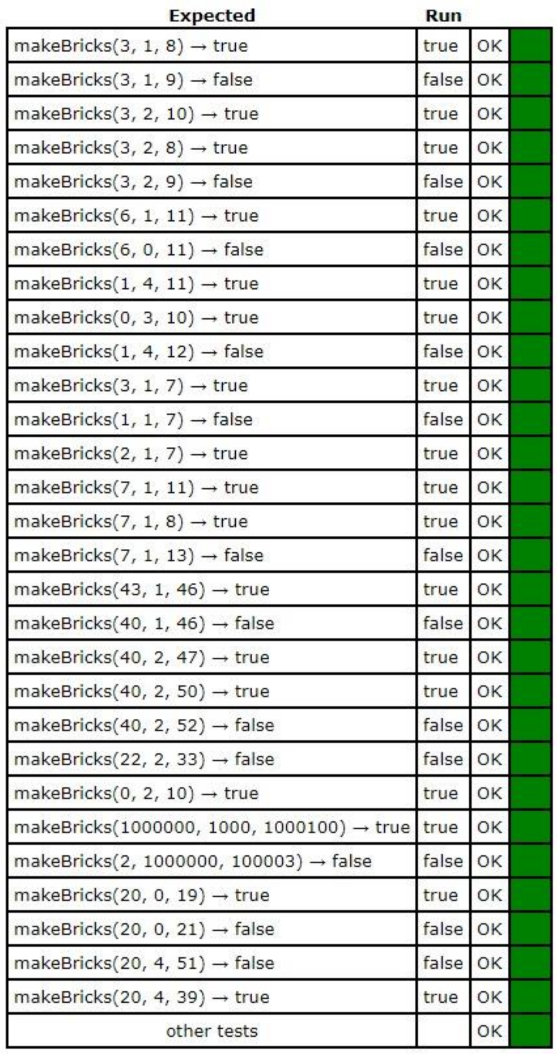
2. **Choose two exercises from the <u>medium logic</u> puzzles. Record the resulting code in your report(S2).**

I Chose:

1. makeBricks:

We want to make a row of bricks that is goal inches long. We have a number of small bricks (1 inch each) and big bricks (5 inches each). Return true if it is possible to make the goal by choosing from the given bricks. This is a little harder than it looks and can be done without any loops. See also: <u>Introduction to MakeBricks</u>(S3).

```
public boolean makeBricks(int small, int big, int
goal) {
  int mod = goal;
  if(big != 0)
    mod = goal % 5;
  if(goal > small + big*5)
    return false;
  if(mod <= small)
    return true;

  return false;
}
```



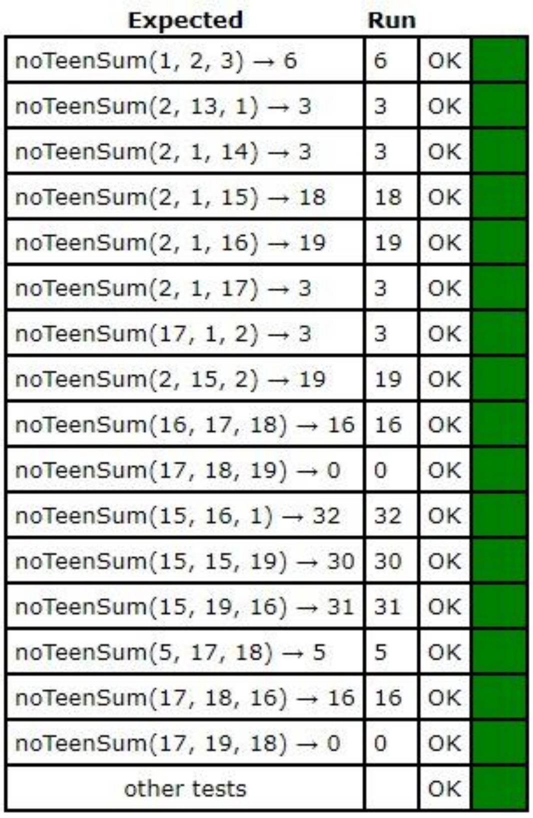
Expected	Run		
makeBricks(3, 1, 8) → true	true	OK	🟩
makeBricks(3, 1, 9) → false	false	OK	🟩
makeBricks(3, 2, 10) → true	true	OK	🟩
makeBricks(3, 2, 8) → true	true	OK	🟩
makeBricks(3, 2, 9) → false	false	OK	🟩
makeBricks(6, 1, 11) → true	true	OK	🟩
makeBricks(6, 0, 11) → false	false	OK	🟩
makeBricks(1, 4, 11) → true	true	OK	🟩
makeBricks(0, 3, 10) → true	true	OK	🟩
makeBricks(1, 4, 12) → false	false	OK	🟩
makeBricks(3, 1, 7) → true	true	OK	🟩
makeBricks(1, 1, 7) → false	false	OK	🟩
makeBricks(2, 1, 7) → true	true	OK	🟩
makeBricks(7, 1, 11) → true	true	OK	🟩
makeBricks(7, 1, 8) → true	true	OK	🟩
makeBricks(7, 1, 13) → false	false	OK	🟩
makeBricks(43, 1, 46) → true	true	OK	🟩
makeBricks(40, 1, 46) → false	false	OK	🟩
makeBricks(40, 2, 47) → true	true	OK	🟩
makeBricks(40, 2, 50) → true	true	OK	🟩
makeBricks(40, 2, 52) → false	false	OK	🟩
makeBricks(22, 2, 33) → false	false	OK	🟩
makeBricks(0, 2, 10) → true	true	OK	🟩
makeBricks(1000000, 1000, 1000100) → true	true	OK	🟩
makeBricks(2, 1000000, 100003) → false	false	OK	🟩
makeBricks(20, 0, 19) → true	true	OK	🟩
makeBricks(20, 0, 21) → false	false	OK	🟩
makeBricks(20, 4, 51) → false	false	OK	🟩
makeBricks(20, 4, 39) → true	true	OK	🟩
other tests		OK	🟩

This one was a bit harder. In “mod” we safed our value corresponding to the number of small bricks needed. So we set our mod equal goal. If we have at least 1 big brick we calculate the rest for any value of big. Then we catch the case if all our bricks are too short for our goal and return false. At the end if our small bricks are enough and more than mod we return true. For every other case we return false.

2. noTeenSum:

Given 3 int values, a b c, return their sum. However, if any of the values is a teen -- in the range 13..19 inclusive -- then that value counts as 0, except 15 and 16 do not count as a teens. Write a separate helper "public int fixTeen(int n) {"that takes in an int value and returns that value fixed for the teen rule. In this way, you avoid repeating the teen code 3 times (i.e. "decomposition"). Define the helper below and at the same indent level as the main noTeenSum().

```
public int noTeenSum(int a, int b, int c) {
  return fixTeen(a)+fixTeen(b)+fixTeen(c);
}
public int fixTeen(int n){
    if(n >= 13 && n <= 19 && n != 15 && n != 16) return 0;
    else return n;
}
```



Expected	Run		
noTeenSum(1, 2, 3) → 6	6	OK	🟩
noTeenSum(2, 13, 1) → 3	3	OK	🟩
noTeenSum(2, 1, 14) → 3	3	OK	🟩
noTeenSum(2, 1, 15) → 18	18	OK	🟩
noTeenSum(2, 1, 16) → 19	19	OK	🟩
noTeenSum(2, 1, 17) → 3	3	OK	🟩
noTeenSum(17, 1, 2) → 3	3	OK	🟩
noTeenSum(2, 15, 2) → 19	19	OK	🟩
noTeenSum(16, 17, 18) → 16	16	OK	🟩
noTeenSum(17, 18, 19) → 0	0	OK	🟩
noTeenSum(15, 16, 1) → 32	32	OK	🟩
noTeenSum(15, 15, 19) → 30	30	OK	🟩
noTeenSum(15, 19, 16) → 31	31	OK	🟩
noTeenSum(5, 17, 18) → 5	5	OK	🟩
noTeenSum(17, 18, 16) → 16	16	OK	🟩
noTeenSum(17, 19, 18) → 0	0	OK	🟩
other tests		OK	🟩

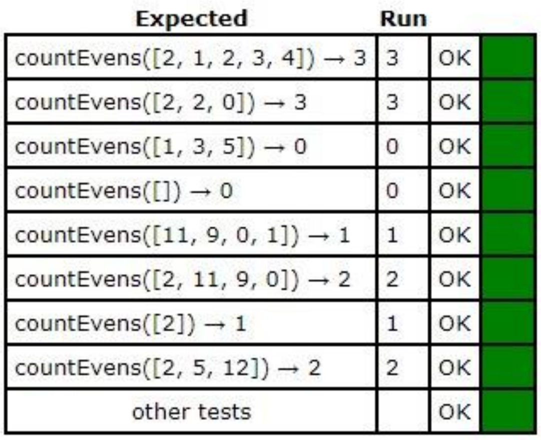
3. **Choose two exercises from the medium array puzzles. Record the resulting code in your report(S4).**

I Chose:

1. countEvents:

Return the number of even ints in the given array. Note: the % "mod" operator computes the remainder, e.g. 5 % 2 is 1.

```
public int countEvens(int[] nums) {
  int count = 0;
  for(int i : nums){
    if(i%2==0)
      count++;
  }
  return count;
}
```

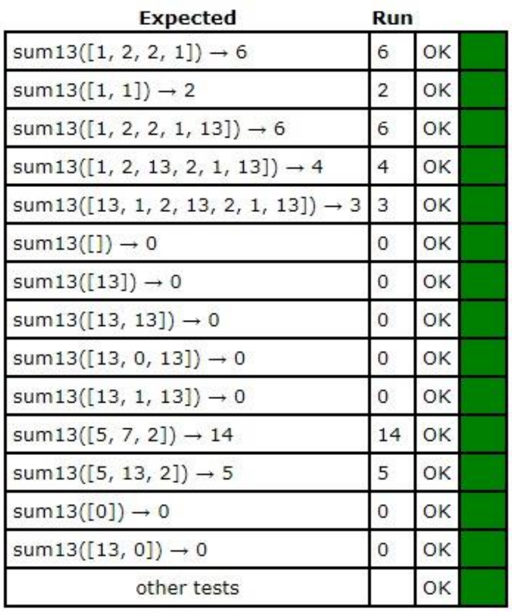


Expected	Run		
countEvens([2, 1, 2, 3, 4]) → 3	3	OK	🟩
countEvens([2, 2, 0]) → 3	3	OK	🟩
countEvens([1, 3, 5]) → 0	0	OK	🟩
countEvens([]) → 0	0	OK	🟩
countEvens([11, 9, 0, 1]) → 1	1	OK	🟩
countEvens([2, 11, 9, 0]) → 2	2	OK	🟩
countEvens([2]) → 1	1	OK	🟩
countEvens([2, 5, 12]) → 2	2	OK	🟩
other tests		OK	🟩

2. sum13

Return the sum of the numbers in the array, returning 0 for an empty array. Except the number 13 is very unlucky, so it does not count and numbers that come immediately after a 13 also do not count.

```
public int sum13(int[] nums) {
  int sum = 0;
  if(nums == null) return 0;
  for(int i = 0; i < nums.length; i++){
    if(nums[i] == 13)
      i++;
    else
      sum += nums[i];
  }
  return sum;
}
```



Expected	Run		
sum13([1, 2, 2, 1]) → 6	6	OK	🟩
sum13([1, 1]) → 2	2	OK	🟩
sum13([1, 2, 2, 1, 13]) → 6	6	OK	🟩
sum13([1, 2, 13, 2, 1, 13]) → 4	4	OK	🟩
sum13([13, 1, 2, 13, 2, 1, 13]) → 3	3	OK	🟩
sum13([]) → 0	0	OK	🟩
sum13([13]) → 0	0	OK	🟩
sum13([13, 13]) → 0	0	OK	🟩
sum13([13, 0, 13]) → 0	0	OK	🟩
sum13([13, 1, 13]) → 0	0	OK	🟩
sum13([5, 7, 2]) → 14	14	OK	🟩
sum13([5, 13, 2]) → 5	5	OK	🟩
sum13([0]) → 0	0	OK	🟩
sum13([13, 0]) → 0	0	OK	🟩
other tests		OK	🟩

4. **Chose one exercise from the harder array puzzles. Record the resulting code in your report.**

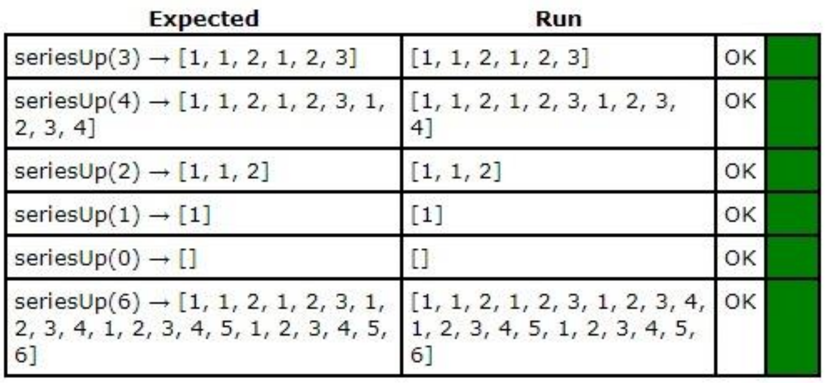
I Chose:

1. seriesUp:

Given n>=0, create an array with the pattern {1,   1, 2,   1, 2, 3,   ... 1, 2, 3 .. n} (spaces added to show the grouping). Note that the length of the array will be 1 + 2 + 3 ... + n, which is known to sum to exactly n*(n + 1)/2.

```
public int[] seriesUp(int n) {
  int lenght = n*(n+1)/2;
  int [] series = new int[lenght];
  int index = 0;
  for(int i = 0; i<n;i++){
    for(int k = 0; k <= i;k++){
      series[index] = k+1;
      index++;
    }
  }
  return series;
}
```

This was the hardest one. First we created a int Array with the given length-formula. Then we create an index which is counting up so that we point to every container. With the outer for loop we go through each separate sequence. A separate sequence consists of only numbers that are counting up, for example: 1,2,3,4 or 1,2,3. The amount of these is equal to n. In the inner loop we take every digit from these sequences and put them into the Array.



Expected	Run		
seriesUp(3) → [1, 1, 2, 1, 2, 3]	[1, 1, 2, 1, 2, 3]	OK	🟩
seriesUp(4) → [1, 1, 2, 1, 2, 3, 1, 2, 3, 4]	[1, 1, 2, 1, 2, 3, 1, 2, 3, 4]	OK	🟩
seriesUp(2) → [1, 1, 2]	[1, 1, 2]	OK	🟩
seriesUp(1) → [1]	[1]	OK	🟩
seriesUp(0) → []	[]	OK	🟩
seriesUp(6) → [1, 1, 2, 1, 2, 3, 1, 2, 3, 4, 1, 2, 3, 4, 5, 1, 2, 3, 4, 5, 6]	[1, 1, 2, 1, 2, 3, 1, 2, 3, 4, 1, 2, 3, 4, 5, 1, 2, 3, 4, 5, 6]	OK	🟩

5. **Assume that you are a member of the programming committee for implementing a method to print the new EU flag for after Brexit. The surprising decision is given below:**

```
         1         2         3         4
1---5----0----5----0----5----0----5----0
????????????????????????????????????????
????????????????????????????????????????
??/= /= /= /= /=|  (    (    (    (  +??
??= /= /= /= /= |  (    (    (    ( + ??
?? /= /= /= /= /|  (    (    (    (+  ??  -5
??/= /= /= /= /=|  (    (    (    +)))??
??= /= /= /= /= |  (    (    (   +    ??
?? /= /= /= /= /|  (    (    (  +     ??
??/= /= /= /= /=|  (    (    ( +))))))??
??= /= /= /= /= |  (    (    (+       ??  -10
?? /= /= /= /= /|  (    (    +        ??
??/= /= /= /= /=|  (    (   +)))))))))??
??= /= /= /= /= |  (    (  +          ??
?? /= /= /= /= /|  (    ( +           ??
??/= /= /= /= /=|  (    (+))))))))))))??  -15
??= /= /= /= /= |  (    +             ??
??---------------  (   +              ??
??  (    (    (    (  +)))))))))))))))??
??  (    (    (    ( +                ??
??  (    (    (    (+                 ??  -20
??  (    (    (    +))))))))))))))))))??
??  (    (    (   +                   ??
??  (    (    (  +                    ??
??  (    (    ( +)))))))))))))))))))))??
??  (    (    (+                      ??  -25
??  (    (    +                       ??
??  (    (   +))))))))))))))))))))))))??
??  (    (  +                         ??
??  (    ( +                          ??
??  (    (+)))))))))))))))))))))))))))??  -30
??  (    +                            ??
??  (   +                             ??
??  (  +))))))))))))))))))))))))))))))??
??  ( +                               ??
??  (+                                ??  -35
??  +)))))))))))))))))))))))))))))))))??
?? +                                  ??
??+                                   ??
????????????????????????????????????????
????????????????????????????????????????  -40
```

The method

```
public char determineCharacter (int column, int row);
```

needs implementation, so that it can be called from the nested loop

```
String outputLine;
for (int row = 1; row <= 40; row++){
    outputLine = "";
    for (int column = 1; column <= 40; column++){
        outputLine = outputLine+determineCharacter (column,
row);
    }
    System.out.println (outputLine);
}
```

Document the body of the method in your report, including a screenshot of it working.

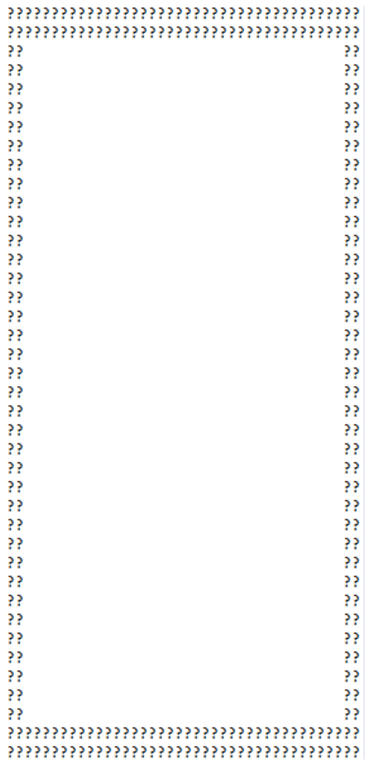
First I implemented everything:

```
package Übung4_5;

public class New_EU_Flagv_1 {

        public static void main(String[] args) {

                String outputLine;

                for (int row = 1; row <= 40; row++){

                   outputLine = "";

                   for (int column = 1; column <= 40; column++){

                      outputLine = outputLine+determineCharacter (column, row);

                   }

                   System.out.println (outputLine);

                }

        }

        public static char determineCharacter (int column, int row) {

        }

}
```

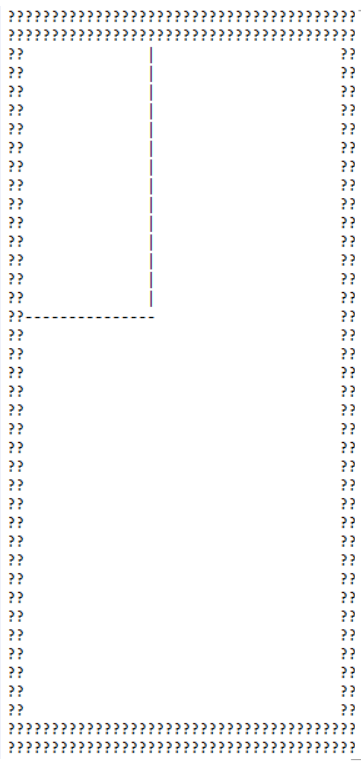
Now I thought about how to solve it. I came up with the idea of useing different layers. These layers are sorted by priority. The top one is the layer which is in front. The next layer is the layer on the same priority or less. We can sort them by using the if-else-if-else...-statement. Layer one corresponds to the first if-statement and so on. Our top-layer is the Border:

```
public static char determineCharacter (int column, int row) {
        //Border:
        int BThick = 2;
        if(column <= BThick || column >= 41- BThick || row <= BThick || row
        >= 41-BThick )
                return '?';

        //rectangle Border:
        else if(column == 17 && row < 17)
                return '|';
        else if(column <= 17 && row == 17)
                return '-';
        //rectangle surface
        else if(column < 17 && row < 17) {
                if((row+column)%3 == 0)
                return '/';
                else if((row+column+1)%3 == 0)
                return ' ';
                else
                return '=';
        }
        // the diagonale +
        else  if(row+column == 41)
                return '+';
        //the vertical porentences
        else if(row+column < 41 && column%5 == 0)
                return '(';
        //the horizontal porentences
        else if(row+column > 41 &&row % 3 == 0)
                return ')';
        return ' ';
}
```

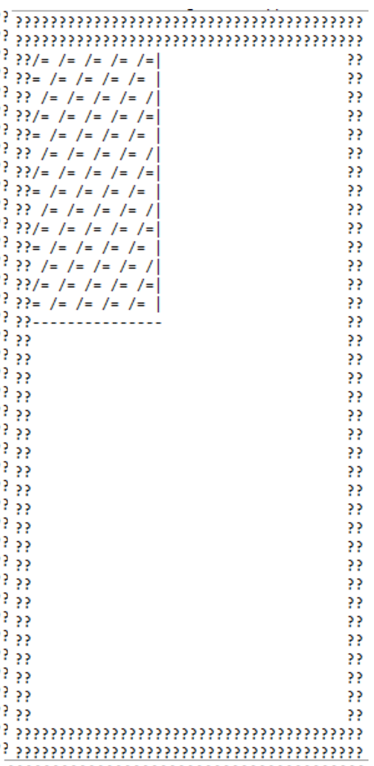
Now I will show you the results by adding each layer step by step.



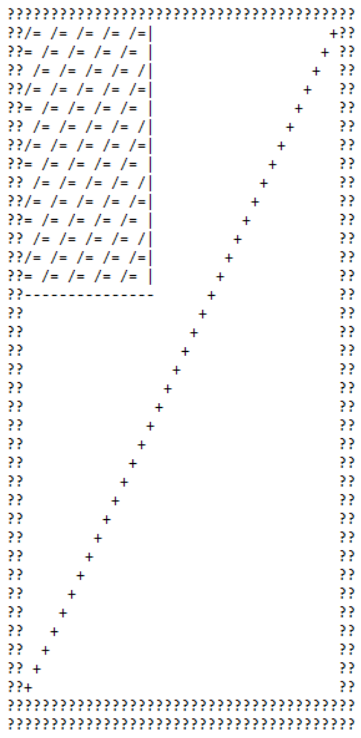
```
????????????????????????????????????????
????????????????????????????????????????
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
????????????????????????????????????????
????????????????????????????????????????
```



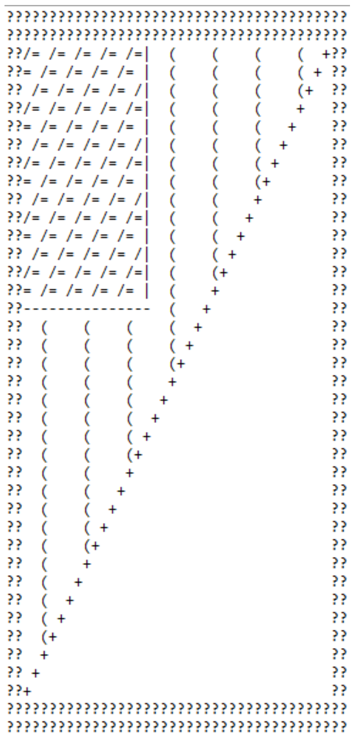
```
????????????????????????????????????????
????????????????????????????????????????
??              |                     ??
??              |                     ??
??              |                     ??
??              |                     ??
??              |                     ??
??              |                     ??
??              |                     ??
??              |                     ??
??              |                     ??
??              |                     ??
??              |                     ??
??              |                     ??
??              |                     ??
??              |                     ??
??---------------                     ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
????????????????????????????????????????
????????????????????????????????????????
```



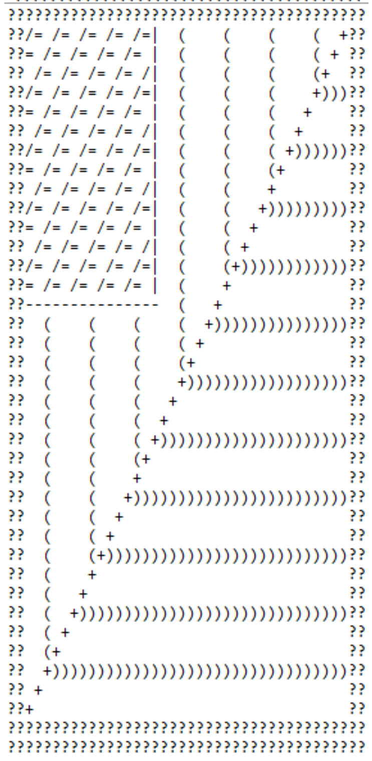
```
????????????????????????????????????????
????????????????????????????????????????
??/= /= /= /= /=|                     ??
??= /= /= /= /= |                     ??
?? /= /= /= /= /|                     ??
??/= /= /= /= /=|                     ??
??= /= /= /= /= |                     ??
?? /= /= /= /= /|                     ??
??/= /= /= /= /=|                     ??
??= /= /= /= /= |                     ??
?? /= /= /= /= /|                     ??
??/= /= /= /= /=|                     ??
??= /= /= /= /= |                     ??
?? /= /= /= /= /|                     ??
??/= /= /= /= /=|                     ??
??= /= /= /= /= |                     ??
??---------------                     ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
??                                    ??
????????????????????????????????????????
????????????????????????????????????????
```



```
????????????????????????????????????????
??/= /= /= /= /=|                    +??
??= /= /= /= /= |                   + ??
?? /= /= /= /= /|                  +  ??
??/= /= /= /= /=|                 +   ??
??= /= /= /= /= |                +    ??
?? /= /= /= /= /|               +     ??
??/= /= /= /= /=|              +      ??
??= /= /= /= /= |             +       ??
?? /= /= /= /= /|            +        ??
??/= /= /= /= /=|           +         ??
??= /= /= /= /= |          +          ??
?? /= /= /= /= /|         +           ??
??/= /= /= /= /=|        +            ??
??= /= /= /= /= |       +             ??
??---------------      +              ??
??                    +               ??
??                   +                ??
??                  +                 ??
??                 +                  ??
??                +                   ??
??               +                    ??
??              +                     ??
??             +                      ??
??            +                       ??
??           +                        ??
??          +                         ??
??         +                          ??
??        +                           ??
??       +                            ??
??      +                             ??
??     +                              ??
??    +                               ??
??   +                                ??
??  +                                 ??
?? +                                  ??
??+                                   ??
????????????????????????????????????????
????????????????????????????????????????
```



```
????????????????????????????????????????
????????????????????????????????????????
??/= /= /= /= /=|  (    (    (    (  +??
??= /= /= /= /= |  (    (    (    ( + ??
?? /= /= /= /= /|  (    (    (    (+  ??
??/= /= /= /= /=|  (    (    (    +   ??
??= /= /= /= /= |  (    (    (   +    ??
?? /= /= /= /= /|  (    (    (  +     ??
??/= /= /= /= /=|  (    (    ( +      ??
??= /= /= /= /= |  (    (    (+       ??
?? /= /= /= /= /|  (    (    +        ??
??/= /= /= /= /=|  (    (   +         ??
??= /= /= /= /= |  (    (  +          ??
?? /= /= /= /= /|  (    ( +           ??
??/= /= /= /= /=|  (    (+            ??
??= /= /= /= /= |  (    +             ??
??---------------  (   +              ??
??  (    (    (    (  +               ??
??  (    (    (    ( +                ??
??  (    (    (    (+                 ??
??  (    (    (    +                  ??
??  (    (    (   +                   ??
??  (    (    (  +                    ??
??  (    (    ( +                     ??
??  (    (    (+                      ??
??  (    (    +                       ??
??  (    (   +                        ??
??  (    (  +                         ??
??  (    ( +                          ??
??  (    (+                           ??
??  (    +                            ??
??  (   +                             ??
??  (  +                              ??
??  ( +                               ??
??  (+                                ??
??  +                                 ??
?? +                                  ??
??+                                   ??
????????????????????????????????????????
????????????????????????????????????????
```

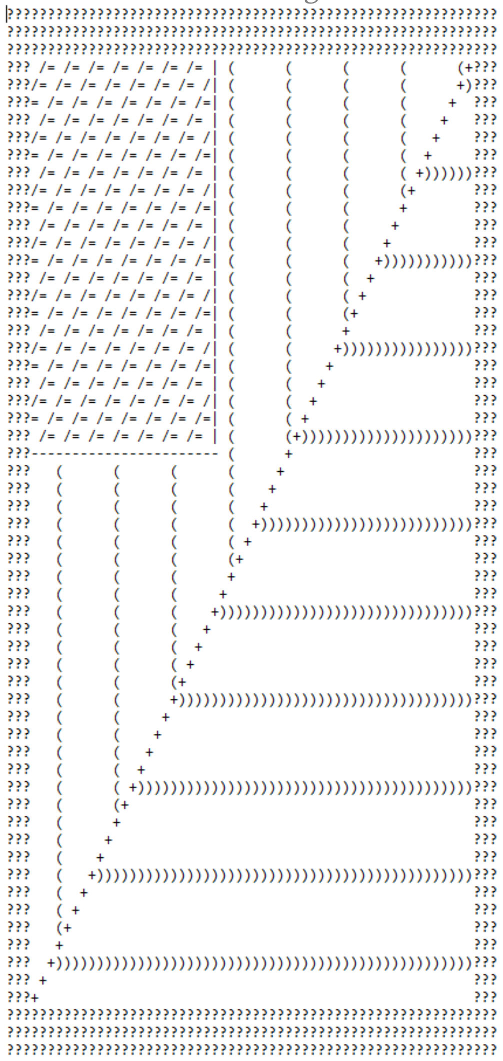


```
????????????????????????????????????????
??/= /= /= /= /=|  (    (    (    (  +??
??= /= /= /= /= |  (    (    (    ( + ??
?? /= /= /= /= /|  (    (    (    (+  ??
??/= /= /= /= /=|  (    (    (    +)))??
??= /= /= /= /= |  (    (    (   +    ??
?? /= /= /= /= /|  (    (    (  +     ??
??/= /= /= /= /=|  (    (    ( +))))))??
??= /= /= /= /= |  (    (    (+       ??
?? /= /= /= /= /|  (    (    +        ??
??/= /= /= /= /=|  (    (   +)))))))))??
??= /= /= /= /= |  (    (  +          ??
?? /= /= /= /= /|  (    ( +           ??
??/= /= /= /= /=|  (    (+))))))))))))??
??= /= /= /= /= |  (    +             ??
??---------------  (   +              ??
??  (    (    (    (  +)))))))))))))))??
??  (    (    (    ( +                ??
??  (    (    (    (+                 ??
??  (    (    (    +))))))))))))))))))??
??  (    (    (   +                   ??
??  (    (    (  +                    ??
??  (    (    ( +)))))))))))))))))))))??
??  (    (    (+                      ??
??  (    (    +                       ??
??  (    (   +))))))))))))))))))))))))??
??  (    (  +                         ??
??  (    ( +                          ??
??  (    (+)))))))))))))))))))))))))))??
??  (    +                            ??
??  (   +                             ??
??  (  +))))))))))))))))))))))))))))))??
??  ( +                               ??
??  (+                                ??
??  +)))))))))))))))))))))))))))))))))??
?? +                                  ??
??+                                   ??
????????????????????????????????????????
????????????????????????????????????????
```

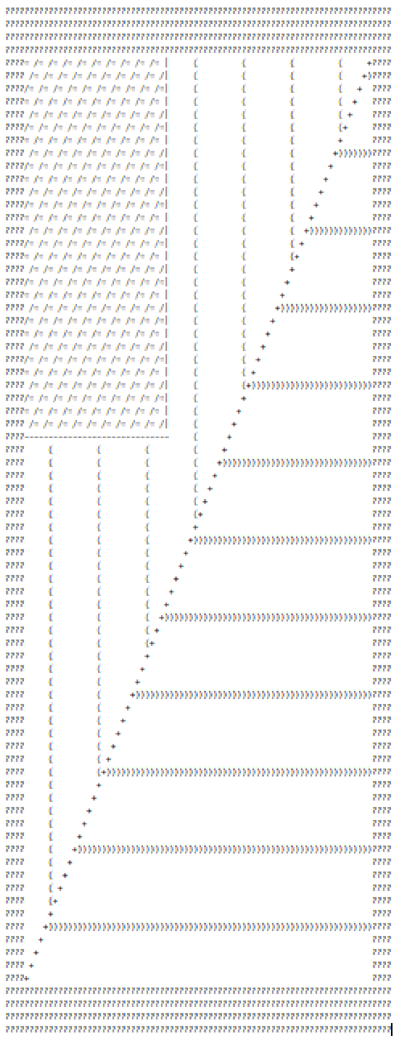
I don't like to have a static size of the flag, so I changed the Code a bit to change the size of the flag, while it´s pattern doesn't change. For example the border is 2 fields thick. Then I wanted to make it dynamic, so that if we double the size of the flag, the border would double too. Same with the rectangle and the other stuff. I achieved that by using the intercept theorem:

```java
package Übung4_5;

public class New_EU_Flagv_2 {

	private static int size =60;

	public static void main(String[] args) {
		String outputLine;


		for (int row = 1; row <= size; row++){
		    outputLine = "";
		    for (int column = 1; column <= size; column++){
			outputLine = outputLine+determineCharacter (column, row);
		    }
		    System.out.println (outputLine);
		}
	}




	public static char determineCharacter (int column, int row) {
		//Border:
		//40/2 = 20
		int BThick = size/20;
		//17/40 = 0.425
		int Rec_Size = (int) Math.round((double)size*0.425);
		//40/5 = 8	Vporen = Vertical porentences
		int Vporen = size/8;
		//3/40 = 0.075Hporen = Horizontal porentences
		int Hporen_Size = (int) Math.round((double)size*0.075);

		if(column <= BThick || column >= size+1- BThick || row <= BThick || row >=
size+1-BThick )
			return '?';
		//rectangle Border:
		else if(column == Rec_Size && row < Rec_Size)
			return '|';
		else if(column <= Rec_Size && row == Rec_Size)
			return '-';
		//rectangle surface
		else if(column < Rec_Size && row < Rec_Size) {
			if((row+column)%3 == 0)
			return '/';
			else if((row+column+1)%3 == 0)
			return ' ';
			else
			return '=';

		}
		// the diagonale +
		else  if(row+column == size+1)
			return '+';
		//the vertical porentences
		else if(row+column < size+1 && column%Vporen == 0)
			return '(';
		//the horizontal porentences
		else if(row+column > size+1 &&row % Hporen_Size == 0)
			return ')';
		return ' ';
	}
}
```

To show that its working I tested it with a size of 60 and 80:



```
????????????????????????????????????????????????????????????
????????????????????????????????????????????????????????????
????????????????????????????????????????????????????????????
??? /= /= /= /= /= /= /= | (      (      (      (      (+???
???/= /= /= /= /= /= /= /| (      (      (      (      +)???
???= /= /= /= /= /= /= /=| (      (      (      (     +  ???
??? /= /= /= /= /= /= /= | (      (      (      (    +   ???
???/= /= /= /= /= /= /= /| (      (      (      (   +    ???
???= /= /= /= /= /= /= /=| (      (      (      (  +     ???
??? /= /= /= /= /= /= /= | (      (      (      ( +))))))???
???/= /= /= /= /= /= /= /| (      (      (      (+       ???
???= /= /= /= /= /= /= /=| (      (      (      +        ???
??? /= /= /= /= /= /= /= | (      (      (     +         ???
???/= /= /= /= /= /= /= /| (      (      (    +          ???
???= /= /= /= /= /= /= /=| (      (      (   +)))))))))))???
??? /= /= /= /= /= /= /= | (      (      (  +            ???
???/= /= /= /= /= /= /= /| (      (      ( +             ???
???= /= /= /= /= /= /= /=| (      (      (+              ???
??? /= /= /= /= /= /= /= | (      (      +               ???
???/= /= /= /= /= /= /= /| (      (     +))))))))))))))))???
???= /= /= /= /= /= /= /=| (      (    +                 ???
??? /= /= /= /= /= /= /= | (      (   +                  ???
???/= /= /= /= /= /= /= /| (      (  +                   ???
???= /= /= /= /= /= /= /=| (      ( +                    ???
??? /= /= /= /= /= /= /= | (      (+)))))))))))))))))))))???
???----------------------- (      +                      ???
???   (      (      (      (     +                       ???
???   (      (      (      (    +                        ???
???   (      (      (      (   +                         ???
???   (      (      (      (  +))))))))))))))))))))))))))???
???   (      (      (      ( +                           ???
???   (      (      (      (+                            ???
???   (      (      (      +                             ???
???   (      (      (     +                              ???
???   (      (      (    +)))))))))))))))))))))))))))))))???
???   (      (      (   +                                ???
???   (      (      (  +                                 ???
???   (      (      ( +                                  ???
???   (      (      (+                                   ???
???   (      (      +))))))))))))))))))))))))))))))))))))???
???   (      (     +                                     ???
???   (      (    +                                      ???
???   (      (   +                                       ???
???   (      (  +                                        ???
???   (      ( +)))))))))))))))))))))))))))))))))))))))))???
???   (      (+                                          ???
???   (      +                                           ???
???   (     +                                            ???
???   (    +                                             ???
???   (   +))))))))))))))))))))))))))))))))))))))))))))))???
???   (  +                                               ???
???   ( +                                                ???
???   (+                                                 ???
???   +                                                  ???
???  +)))))))))))))))))))))))))))))))))))))))))))))))))))???
??? +                                                    ???
???+                                                     ???
????????????????????????????????????????????????????????????
????????????????????????????????????????????????????????????
????????????????????????????????????????????????????????????
```



```
????????????????????????????????????????????????????????????????????????????????
????????????????????????????????????????????????????????????????????????????????
????????????????????????????????????????????????????????????????????????????????
????????????????????????????????????????????????????????????????????????????????
????= /= /= /= /= /= /= /= /= /= |     (         (         (         (     +????
???? /= /= /= /= /= /= /= /= /= /|     (         (         (         (    +)????
????/= /= /= /= /= /= /= /= /= /=|     (         (         (         (   +  ????
????= /= /= /= /= /= /= /= /= /= |     (         (         (         (  +   ????
???? /= /= /= /= /= /= /= /= /= /|     (         (         (         ( +    ????
????/= /= /= /= /= /= /= /= /= /=|     (         (         (         (+     ????
????= /= /= /= /= /= /= /= /= /= |     (         (         (         +      ????
???? /= /= /= /= /= /= /= /= /= /|     (         (         (        +)))))))????
????/= /= /= /= /= /= /= /= /= /=|     (         (         (       +        ????
????= /= /= /= /= /= /= /= /= /= |     (         (         (      +         ????
???? /= /= /= /= /= /= /= /= /= /|     (         (         (     +          ????
????/= /= /= /= /= /= /= /= /= /=|     (         (         (    +           ????
????= /= /= /= /= /= /= /= /= /= |     (         (         (   +            ????
???? /= /= /= /= /= /= /= /= /= /|     (         (         (  +)))))))))))))????
????/= /= /= /= /= /= /= /= /= /=|     (         (         ( +              ????
????= /= /= /= /= /= /= /= /= /= |     (         (         (+               ????
???? /= /= /= /= /= /= /= /= /= /|     (         (         +                ????
????/= /= /= /= /= /= /= /= /= /=|     (         (        +                 ????
????= /= /= /= /= /= /= /= /= /= |     (         (       +                  ????
???? /= /= /= /= /= /= /= /= /= /|     (         (      +)))))))))))))))))))????
????/= /= /= /= /= /= /= /= /= /=|     (         (     +                    ????
????= /= /= /= /= /= /= /= /= /= |     (         (    +                     ????
???? /= /= /= /= /= /= /= /= /= /|     (         (   +                      ????
????/= /= /= /= /= /= /= /= /= /=|     (         (  +                       ????
????= /= /= /= /= /= /= /= /= /= |     (         ( +                        ????
???? /= /= /= /= /= /= /= /= /= /|     (         (+)))))))))))))))))))))))))????
????/= /= /= /= /= /= /= /= /= /=|     (         +                          ????
????= /= /= /= /= /= /= /= /= /= |     (        +                           ????
???? /= /= /= /= /= /= /= /= /= /|     (       +                            ????
????------------------------------     (      +                             ????
????     (         (         (         (     +                              ????
????     (         (         (         (    +)))))))))))))))))))))))))))))))????
????     (         (         (         (   +                                ????
????     (         (         (         (  +                                 ????
????     (         (         (         ( +                                  ????
????     (         (         (         (+                                   ????
????     (         (         (         +                                    ????
????     (         (         (        +)))))))))))))))))))))))))))))))))))))????
????     (         (         (       +                                      ????
????     (         (         (      +                                       ????
????     (         (         (     +                                        ????
????     (         (         (    +                                         ????
????     (         (         (   +                                          ????
????     (         (         (  +)))))))))))))))))))))))))))))))))))))))))))????
????     (         (         ( +                                            ????
????     (         (         (+                                             ????
????     (         (         +                                              ????
????     (         (        +                                               ????
????     (         (       +                                                ????
????     (         (      +)))))))))))))))))))))))))))))))))))))))))))))))))????
????     (         (     +                                                  ????
????     (         (    +                                                   ????
????     (         (   +                                                    ????
????     (         (  +                                                     ????
????     (         ( +                                                      ????
????     (         (+)))))))))))))))))))))))))))))))))))))))))))))))))))))))????
????     (         +                                                        ????
????     (        +                                                         ????
????     (       +                                                          ????
????     (      +                                                           ????
????     (     +                                                            ????
????     (    +)))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))????
????     (   +                                                              ????
????     (  +                                                               ????
????     ( +                                                                ????
????     (+                                                                 ????
????     +                                                                  ????
????    +)))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))????
????   +                                                                    ????
????  +                                                                     ????
???? +                                                                      ????
????+                                                                       ????
????????????????????????????????????????????????????????????????????????????????
????????????????????????????????????????????????????????????????????????????????
????????????????????????????????????????????????????????????????????????????????
????????????????????????????????????????????????????????????????????????????????
```

Reflections

JURI

There isn't much to say about this week's lab. I think that for the students that need to train a bit it's a nice way to do so. In the last assignment I learned a nice method for structuring and solving such problems, with what I like to call now: the "Layer technique."

S1: https://codingbat.com/java/Logic-1

S2: https://codingbat.com/java/Logic-2

S3: https://codingbat.com/doc/practice/makebricks-introduction.html

S4: https://codingbat.com/java/Array-2

Appendix:

Java Code:

Version 1:

```
package Übung4_5;

public class New_EU_Flagv_1 {

	public static void main(String[] args) {
		String outputLine;


		for (int row = 1; row <= 40; row++){
		    outputLine = "";
		    for (int column = 1; column <= 40; column++){
		        outputLine = outputLine+determineCharacter (column, 
row);
		    }
		    System.out.println (outputLine);
		}
	}



	public static char determineCharacter (int column, int row) {
		//Border:
		int BThick = 2;
		if(column <= BThick || column >= 41- BThick || row <= BThick 
|| row >= 41-BThick )
			return '?';

		//rectangle Border:
		else if(column == 17 && row < 17)
			return '|';
		else if(column <= 17 && row == 17)
			return '-';
		//rectangle surface
		else if(column < 17 && row < 17)
			if((row+column)%2 == 0)
			return '/';
			else
			return '=';
		// the diagonale +
		else  if(row+column == 41)
			return '+';
		//the vertical porentences
		else if(row+column < 41 && column%5 == 0)
			return '(';
		//the horizontal porentences
		else if(row+column > 41 &&row % 3 == 0)
			return ')';
		return ' ';
```

```
        }
}
```

Version2:

```
package Übung4_5;

public class New_EU_Flagv_2 {

        private static int size =80;

        public static void main(String[] args) {
                String outputLine;


                for (int row = 1; row <= size; row++){
                    outputLine = "";
                    for (int column = 1; column <= size; column++){
                        outputLine = outputLine+determineCharacter (column,
row);
                    }
                    System.out.println (outputLine);
                }
        }
        public static char determineCharacter (int column, int row) {
                //Border:
                //40/2 = 20
                int BThick = size/20;
                //17/40 = 0.425
                int Rec_Size = (int) Math.round((double)size*0.425);
                //40/5 = 8    Vporen = Vertical porentences
                int Vporen = size/8;
                //3/40 = 0.075      Hporen = Horizontal porentences
                int Hporen_Size = (int) Math.round((double)size*0.075);

                if(column <= BThick || column >= size+1- BThick || row <=
BThick || row >= size+1-BThick )
                        return '?';
                //rectangle Border:
                else if(column == Rec_Size && row < Rec_Size)
                        return '|';
                else if(column <= Rec_Size && row == Rec_Size)
                        return '-';
                //rectangle surface
                else if(column < Rec_Size && row < Rec_Size)
                        if((row+column)%2 == 0)
                        return '/';
                        else
                        return '=';
                // the diagonale +
                else  if(row+column == size+1)
                        return '+';
                //the vertical porentences
                else if(row+column < size+1 && column%Vporen == 0)
                        return '(';
			//the horizontal porentences
			else if(row+column > size+1 &&row % Hporen_Size == 0)
				return ')';
			return ' ';
		}
	}
```