Formulating algorithm Guessing Your Number (Bonus2.java)

1 I import a class Scanner. Scanner is in **java.util** package used for obtaining the input of the primitive types like int, double, etc. and strings.

2 Initilize an integer data type variable **UserGuess** equal to zero and declare **UserInput** variable.

3 Declare a boolean data type to the **CorrectGuess and stepBack** which set to **false** and **stepForward** to **true**.

4 Initialize an integer data type to the **attempts** variable and set to zero.

5 Initialize an integer data type to the **step** variable and set to twenty, so it will start make step up to twenty.

6 Initialize integer variable **midRange** to zero, therefore it will start range from 0 to user.

7 Declare **Scanner** variable **scan** which is used to specify the type of data that the user will enter.

8 Create a list with the data type of an integer named **list** and fill it with dimensions from zero to one hundred cells.

9 Create a **for** loop in which we declare an integer **i** and assign it to zero. Create a limit to which the **for** loop will work with an addition to plus one until it reaches one hundred and fills our list.

10 Create a **while** statement which should continue looping as long as **CorrectGuess**'s value will not equal true.

11 We already created a variable **UserGuess** to which I have equated **list** with **midRange**, which will be filled with **midRange** values. Initially there will be a value of zero so that it is convenient for the user to start calculating his hidden number.

12 I applied a postfix increment (**++**) to the variable attempts (**attempts++**), which means that with each iteration of the **while** loop, one will be added to the counter variable, which was initially zero, and so on, one by one will be added until the **while** loop exits its loop body (until **CorrectGuess** is true). This will show the user how many attempts it took for the program to guess its unknown value.

13 I will write pair of outputs to show the user what to click to understand and what result to show.

14 I apply the **UserInput** variable to the **Scanner**, that is, this variable will take on the value that the user enters. Using the previously created **scan** variable, I say values will be accepted as input by the user.

15 Create **if** statement. If user input equal one, he/she will see output in the form of text and **while** loop will finish, because **CorrectGuess** will be equal **true**.

16 Create **else if** statement. If user input equal two, he/she will see on the console a step greater than the previous one and user will be closer to the true number that he/she guessed. I initialized **true** to **stepForward**, because I will add **if** statement(String 71), it means that if **stepForward** equal **true**, will work formula (**midrange += step**), which will do changing in the value of **midRange**. Another statement **if** which is in **else if,** making calculating of step only when **stepBack** equal true. And then it quit from **else if** statement and go String 71.

17 Create **else if** statement. If user input equal three, he/she will see on the console a step lower than the previous one and user will be closer to the true number that he/she guessed. I initialized **true** to **stepBack**, because I will add **else** statement(String 74), it means that if **stepBack** equal **true**, will work formula (**midrange -= step**), which will do changing in the value of **midRange**. Another statement **if** which is in **else if,** making calculating of step only when **stepForward** equal true(that is why I did **stepForward** equal to **true**, because we have to get number always in two times less(step=step/2) when user's input equal 3. And then it quit from **else if** statement and go String 74.

18 Like any object that works with I/O Streams, the scanner must be closed at the end of its work in order to no longer consume the resources of our computer.