

CMSC131, Spring 2023, QUIZ #2 (DURATION: 25 MINUTES) – 30 pts

FIRSTNAME, LASTNAME (PRINT IN UPPERCASE):

STUDENT ID (e.g. 123456789):

INSTRUCTIONS:

- Write a program that will prompt for a start value (Assume you will be given a positive integer). Print to the screen a “rectangle” with 20 rows, where each row has 20 symbols. A symbol is defined as an \* or an integer. Odd rows will have integers starting with the start value and even rows will have 20 \*. The first row that prints is considered row 1, and therefore will be an odd row. When printing integers in odd rows, each successive integer should be printed once more than the previous integer. For example, in the sample run, 7 (the start value) is printed once, but 8 (the next integer) is printed twice, and 9 is printed three times, and so forth. Notice that when printing an integer a certain number of times, you might have to start a row of \* before you can resume printing that integer. For example, this happens to the 6 copies of 12 you have to print, as the last 12 shows up in row 3 since all 20 symbols have been used up in row 1 before you can print the last 12.

Keep in mind that as each symbol takes up a different amount of room, the figure will not look like a typical rectangle. For example, there are 20 \* in the even rows but it does not line up with the 20 integers above and below. You don’t need to put extra spaces to make it look like the output, just print 20 symbols per row and it will match the output below. Also, due to the fixed dimension of 20 by 20, the last integer (in the sample output, 26) will not have all of its copies printed out as the algorithm finishes.

Here is a sample run (with user input of 7 in italic and output in bold just to make it easier to understand how it works).

**Sample run #1**

```
Enter start int: 7
7 8 8 9 9 9 10 10 10 10 11 11 11 11 11 12 12 12 12 12
* * * * *
12 13 13 13 13 13 13 13 14 14 14 14 14 14 14 15 15 15 15
* * * * *
15 15 15 15 15 16 16 16 16 16 16 16 16 16 16 17 17 17 17
* * * * *
17 17 17 17 17 17 18 18 18 18 18 18 18 18 18 18 18 18 19 19
* * * * *
19 19 19 19 19 19 19 19 19 19 19 20 20 20 20 20 20 20 20
* * * * *
20 20 20 20 20 21 21 21 21 21 21 21 21 21 21 21 21 21 21
* * * * *
22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 23 23 23 23
* * * * *
23 23 23 23 23 23 23 23 23 23 23 23 23 24 24 24 24 24 24
* * * * *
24 24 24 24 24 24 24 24 24 24 24 25 25 25 25 25 25 25 25
* * * * *
25 25 25 25 25 25 25 25 25 25 26 26 26 26 26 26 26 26 26
* * * * *
```

**WRITE THE CODE ON THE BACK**

```

import java.util.Scanner;
public class Q2 {
    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter start int: ");

        int start = scanner.nextInt();

        int current = start;
        int count =1;

        for (int row = 1; row <= 20; row++) {
            for (int col = 1; col <= 20; col++) {

                if(row%2==0) //even row
                {
                    System.out.print("* ");
                }
                else { //odd row

                    System.out.print(current + " ");
                    count--;

                    if (count==0)
                    {
                        current++; //next int
                        count = current -start +1;
                        //print it once more than previous int
                    }

                }

            }

            System.out.println(); // Next line
        }

        scanner.close();

    }

}

```