*Loop App output*

Count is: 1

Count is: 3

Count is: 7

Count is: 15

Count is: 31

Count is: 63

Count is: 127

Count is: 255

Count is: 511

Count is: 1023

Count is: 2047

Count is: 4095

The first line, Public static void main(String[] args), creates the mainline where the program is run. The line, int I; declares the variable “I” and assigns it as an integer. The next line is where the for loop is created, for(i=1; -<5000; i=2\*i+1). The i=1 assigns the variable I a value of one as the loop is initiated. The i<5000 is the condition for the loop to exist, so as long as i is less than 5000 the loop will run/continue to run. i=2\*i+1 is executed at the end of each loop, so after the print line runs and prints the i value, the value is doubled and gets one added to it. This is shown in the output. For example after a few loops it prints 255, before it loops again it does 255\*2+1=511, now i’s new value is 511, and in the next loop 511 will be outputted.

*Dice App output*

Would you like to roll 2 dice? Yes/No -->

Yes

You rolled 5 and 6

Total: 11

Would you like to roll 2 dice? Yes/No -->

Yes

You rolled 5 and 3

Total: 8

Would you like to roll 2 dice? Yes/No -->

Yes

You rolled 3 and 3

Total: 6

Would you like to roll 2 dice? Yes/No -->

No

*Addition Table App output*  (output neat in the IDE but looks a little slopy copy and pasted into word)

Enter a number 3-20 -->

17

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18

2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21

5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22

6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23

7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24

8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25

9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26

10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27

11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28

12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29

13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30

14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32

16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33

17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34