

# Template Week 2 – Logic

Student number: 585902

## Assignment 2.1: Parking lot

Which gates do you need? Two AND gates

Complete this table

Parking lot 1	Parking lot 2	Parking lot 3	Result (full)
0	0	0	0
0	0	1	0
0	1	0	0
1	0	0	0
0	1	1	0
1	1	0	0
1	0	1	0
1	1	1	1

## Assignment 2.2: Android or iPhone

Which gates do you need? XOR – 4 NANDS

Complete this table

Android phone	iPhone	Result (Phone in possession)
0	0	0
1	0	1
0	1	1
1	1	0

### Assignment 2.3: Four NAND gates

Complete this table

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

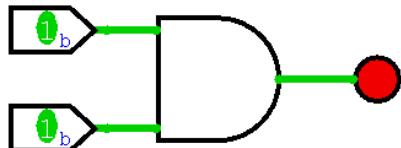
How can the design be simplified? One XOR

### Assignment 2.4: Getting to know Logisim evolution

Screenshot of the design with your name and student number in it:

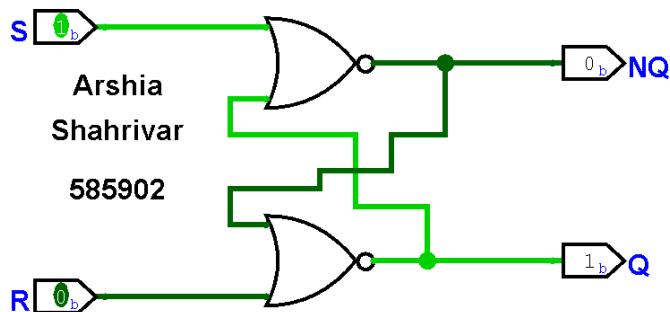
**585902**

**Arshia Shahrvār**



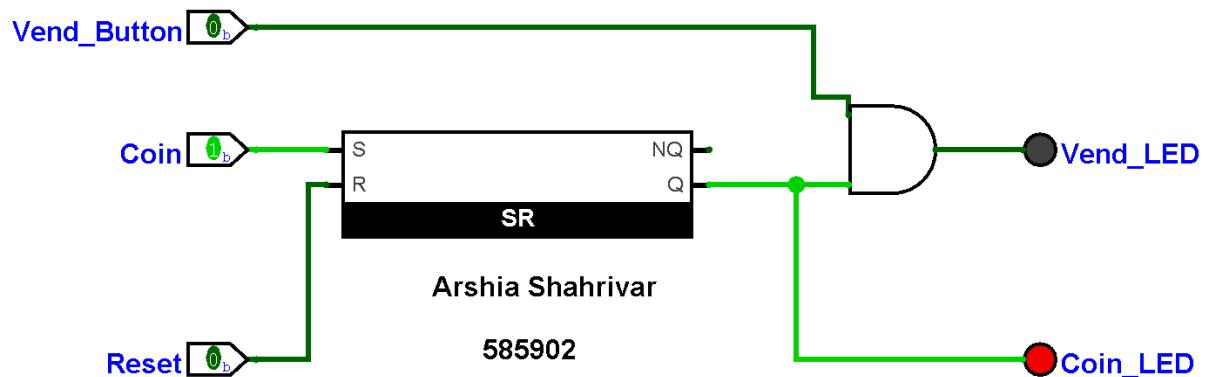
### Assignment 2.5: SR Latch

Screenshot SR Latch in Logisim with your name and student number:



### Assignment 2.6: Vending Machine

Screenshot Vending Machine in Logisim with your name and student number:



### Assignment 2.7: Bitwise operators

Complete the java source code for bitwise operators. Put the source code here.

- Checking if the number is even or odd:

```
• public class Main {  
    public static void main(String[] args) {  
        int number = 5;  
        if((number & 1) == 1){  
            System.out.println("number is odd");  
        } else {  
            System.out.println("number is even");  
        }  
    }  
}
```

- Checking if the number is power of two:

```
• public class Main {  
    public static void main(String[] args) {  
        int number = 4;  
        if((number & number - 1) == 0){  
            System.out.println("it is power of 2");  
        } else {  
            System.out.println("its not power of 2");  
        }  
    }  
}
```

- Checking if user has read permissions:

```
• public class Main {
    public static void main(String[] args) {
        final int READ = 4;
        final int WRITE = 2;
        final int EXECUTE = 1;
        int userPermissions = 7;
        if ((userPermissions & 4) == 4) {
            System.out.println("user has read permissions");
        } else {
            System.out.println("user doesnt have read permission");
        }
    }
}
```

- Changing the user permissions to have read + execute:

```
• public class Main {
    public static void main(String[] args) {
        final int READ = 4;
        final int WRITE = 2;
        final int EXECUTE = 1;
        int userPermissions = READ | EXECUTE;
        System.out.println("user permissions: " + userPermissions);
    }
}
```

- Making the permissions to be read only:

```
• public class Main {
    public static void main(String[] args) {
        final int READ = 4;
        final int WRITE = 2;
        final int EXECUTE = 1;
        int userPermissions = 6;
        userPermissions ^= 2;
        System.out.println("user permissions: " + userPermissions);
    }
}
```

- reversing the sign of a number from positive to negative and vice versa:

```
• public class Main {
    public static void main(String[] args) {
        int number = 5;
        number = ~number + 1;
        System.out.println("Number became: " + number);
    }
}
```

## Assignment 2.8: Java Application Bit Calculations

Create a java program that accepts user input and presents a menu with options.

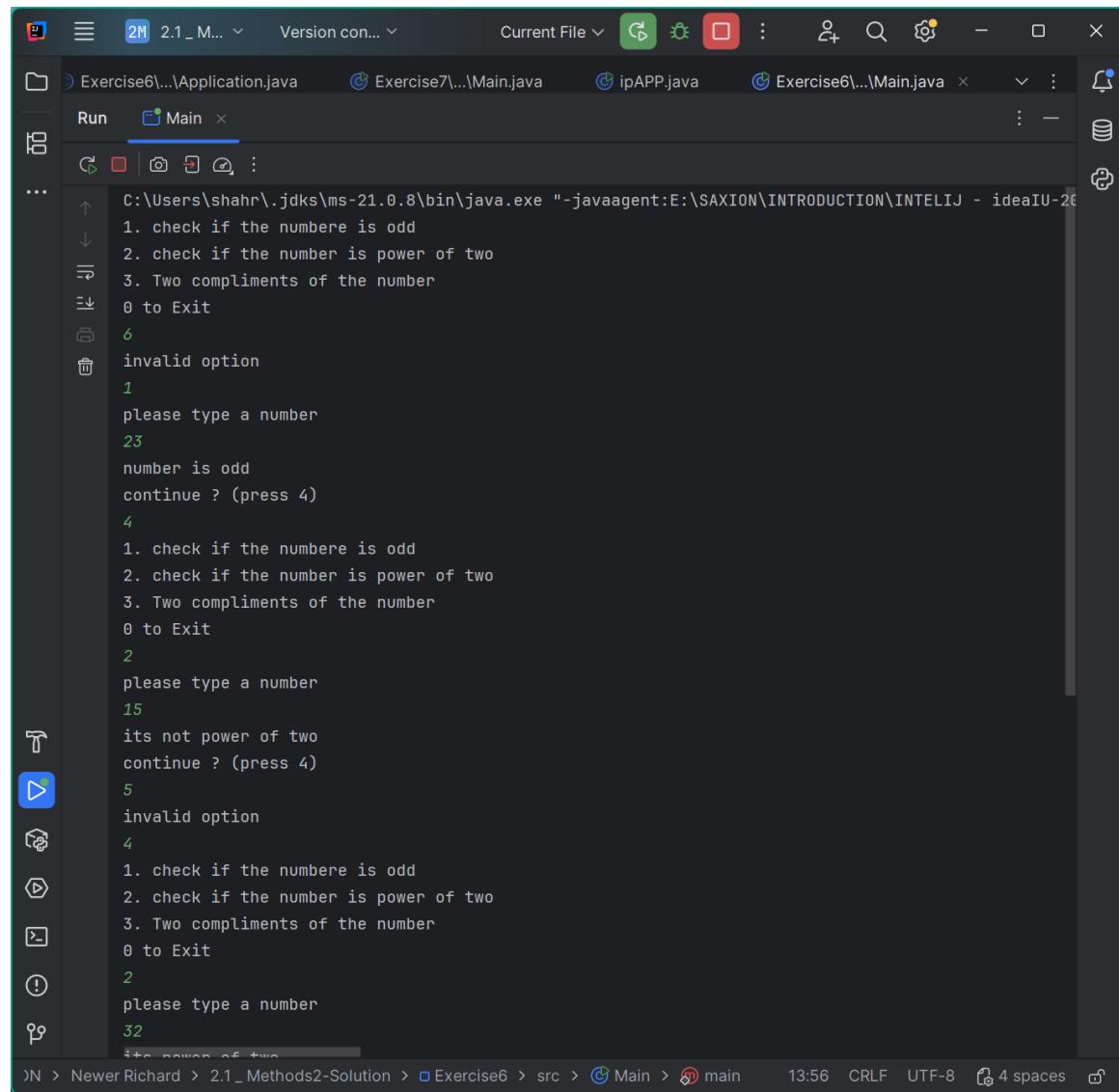
1. Is number odd?
2. Is number a power of 2?
3. Two's complement of number?

Implement the methods by using the bitwise operators you have just learned.

Organize your source code in a readable manner with the use of control flow and methods.

Keep this application because you need to expand it in week 6 for calculating network segments.

Paste source code here, with a screenshot of a working application.



The screenshot shows the IntelliJ IDEA interface with a terminal window open. The terminal displays the output of a Java application named 'Main'. The application presents a menu with three options: 1. check if the number is odd, 2. check if the number is power of two, and 3. Two compliments of the number. It also includes an option to exit (0). If an invalid option is chosen, it prompts the user to type a number. The application then checks if the number is odd or a power of two and continues until the user chooses to exit. The terminal window is part of the IntelliJ IDEA interface, which also shows other projects and files in the background.

```
C:\Users\shahr\.jdks\ms-21.0.8\bin\java.exe "-javaagent:E:\SAXION\INTRODUCTION\INTELIJ - ideaIU-202.4.1.jar" -Dfile.encoding=UTF-8 Main
1. check if the number is odd
2. check if the number is power of two
3. Two compliments of the number
0 to Exit
6
invalid option
1
please type a number
23
number is odd
continue ? (press 4)
4
1. check if the number is odd
2. check if the number is power of two
3. Two compliments of the number
0 to Exit
2
please type a number
15
its not power of two
continue ? (press 4)
5
invalid option
4
1. check if the number is odd
2. check if the number is power of two
3. Two compliments of the number
0 to Exit
2
please type a number
32
its power of two
```

```
invalid option
4
1. check if the number is odd
2. check if the number is power of two
3. Two compliments of the number
0 to Exit
2
please type a number
32
its power of two
continue ? (press 4)
4
1. check if the number is odd
2. check if the number is power of two
3. Two compliments of the number
0 to Exit
1
please type a number
46
number is even
continue ? (press 4)
4
1. check if the number is odd
2. check if the number is power of two
3. Two compliments of the number
0 to Exit
3
please type a number
67
-67
continue ? (press 4)
```

## SCREENSHOTS OF THE SOURCE CODE IN THE NEXT PAGES

The screenshot shows a Java code editor interface with two tabs open: `Application.java` and `Main.java`. The `Main.java` tab is active, displaying the following code:

```
import java.util.Scanner;
public class Main {
    static Scanner scanner = new Scanner(System.in); 4 usages
    public static void main(String[] args){
        printMenu();
        while(true) {
            int userInput = getValidChoice();
            if(userInput == 0){
                break;
            } else if(userInput == 1){
                System.out.println("please type a number");
                checkOdd();
                System.out.println("continue ? (press 4)");
            }else if(userInput == 2){
                System.out.println("please type a number");
                checkPower();
                System.out.println("continue ? (press 4)");
            } else if(userInput == 3){
                System.out.println("please type a number");
                inverse();
                System.out.println("continue ? (press 4)");
            } else if(userInput == 4){
                printMenu();
            }
        }
    }
    public static void printMenu(){ 2 usages
        System.out.println("1. check if the number is odd");
        System.out.println("2. check if the number is power of two");
        System.out.println("3. Two compliments of the number");
        System.out.println("0 to Exit");
    }
    public static int getValidChoice(){ 1 usage
        while(true) {
```

The code implements a menu system with four options: checking if a number is odd, checking if it's a power of two, getting two complements, or exiting. It uses `System.out.println` for output and `Scanner` for input. The `getValidChoice` method uses a `while(true)` loop to ensure valid user input.

The screenshot shows a Java code editor with the following code:

```
2 public class Main {  
3     public static int getValidChoice(){ 1 usage  
4         while(true) {  
5             int choice = scanner.nextInt();  
6             if (choice < 0 || choice > 4) {  
7                 System.out.println("invalid option");  
8                 continue;  
9             }  
10            return choice;  
11        }  
12    }  
13  
14    public static void checkOdd(){ 1 usage  
15        int number = scanner.nextInt();  
16        if((number & 1) == 1){  
17            System.out.println("number is odd");  
18        } else {  
19            System.out.println("number is even");  
20        }  
21    }  
22  
23    public static void checkPower(){ 1 usage  
24        int number = scanner.nextInt();  
25        if((number & number - 1) == 0){  
26            System.out.println("its power of two");  
27        } else {  
28            System.out.println("its not power of two");  
29        }  
30    }  
31  
32    public static void inverse(){ 1 usage  
33        int number = ~scanner.nextInt() + 1;  
34        System.out.println(number);  
35    }  
36}
```

The code consists of four static methods: `getValidChoice`, `checkOdd`, `checkPower`, and `inverse`. The `getValidChoice` method reads an integer from the user and continues to do so until the user enters a value between 0 and 4. The `checkOdd` method checks if a number is odd or even. The `checkPower` method checks if a number is a power of two. The `inverse` method reads an integer from the user and prints its two's complement.

Ready? Then save this file and export it as a pdf file with the name: **week2.pdf**