Template Week 2 – Logic

Student number: 486707

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
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Assignment 2.2: Android or iPhone

Which gates do you need?

A XOR gate

Complete this table

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

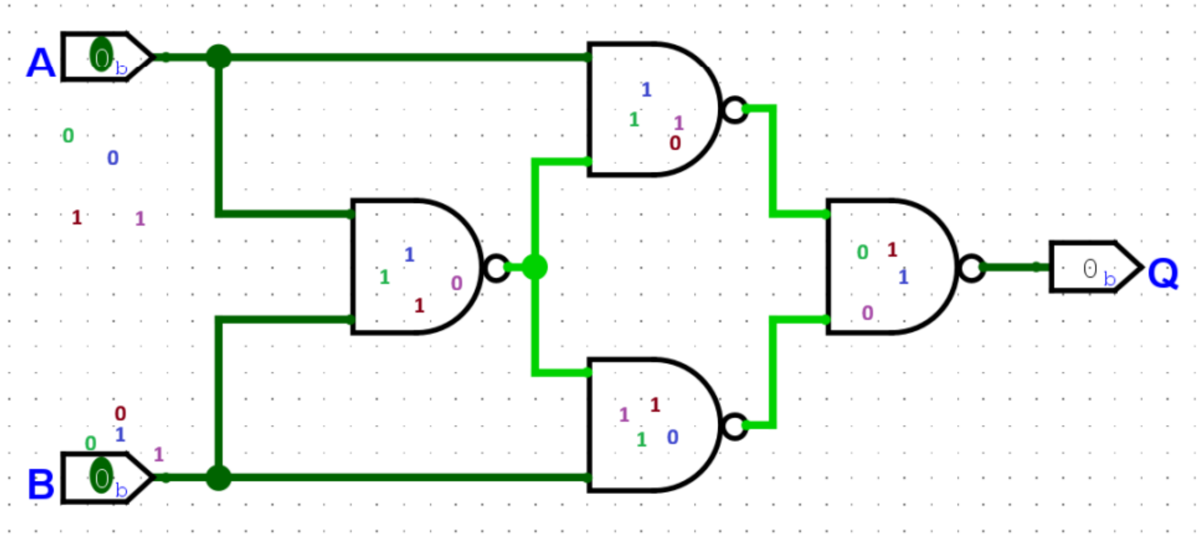
Assignment 2.3: Four NAND gates

Complete this table

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

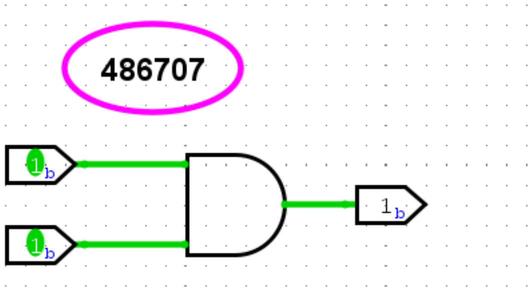
How can the design be simplified?

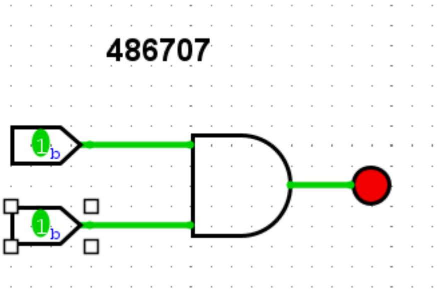
A single XOR gate



Assignment 2.4: Getting to know Logisim evolution

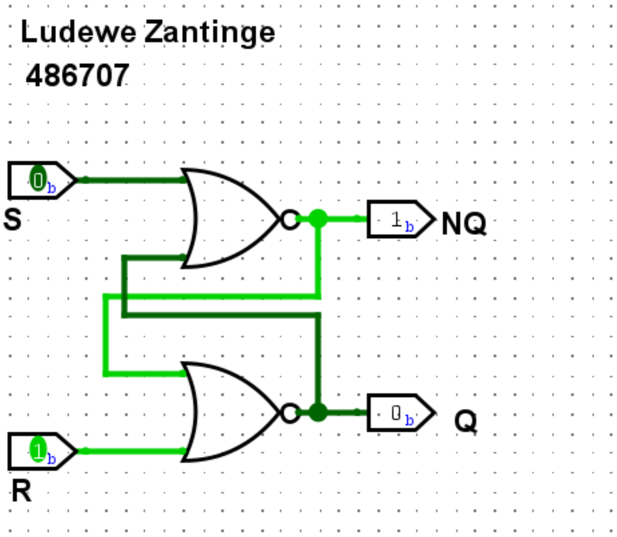
Screenshot of the design with your name and student number in it:





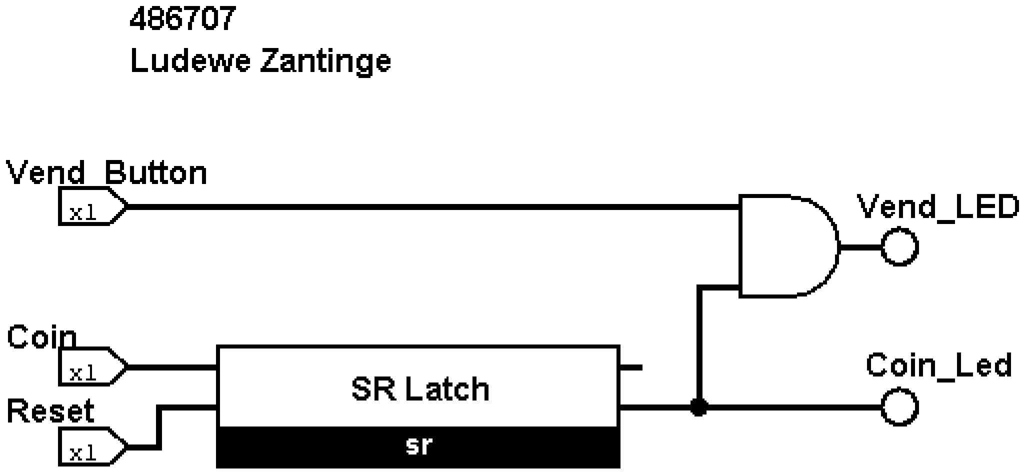
Assignment 2.5: SR Latch

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.

#1

```
public class Main {
public static void main(String[] args) {
int number = 17;
if(((number & 1) == 0)) System.out.println("number is even");
else System.out.println("number is odd");
}
}
```

#2

```
public class Main {
public static void main(String[] args) {
int number = 8;
if((number & (number-1)) == 0) System.out.println("Number is power of two");
else System.out.println("Number is not power of two");
}
}
```

This works because if a binary number is a power of two and you substract 1 it will change the 1 into 0 and the following 0’s into 1’s. Then this makes an AND operation return 0.

#3

```
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) & READ)> 0 ) System.out.println("User has read permissions");

else System.out.println("User can't read. No permissions.");

}

}
```

This works because it checks if the bit set at 4 (read) matches with the bit set in the userPermission variable which should be set if it is equal or bigger than 4 by doing a binary AND calculation. The result should be bigger than 0 if they do match.

#4

```
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);

}
}
```

This does an OR operation on the binary READ which is 4 or binary 100 and EXECUTE which is binary 001. Doing an OR operation on 100 and 001 makes it 101 which is 5.

#5

```
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 = userPermissions;
System.out.println("User permissions: "+(userPermissions ^ WRITE));

}
}
```

By doing a XOR (^ in java) on the userPermission variable which is 6 binary 110, and WRITE which is 2 binary 010. 110 XOR 010 makes 100 which is 4. The permission is now 4 which is only read, the write permission has been removed.

#6

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

}
```

}

This flips the bits by using the java ~ on the number and then adds +1. This makes the positive number into a negative one. Then doing the same thing again on the negative number changes it back to positive.

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?

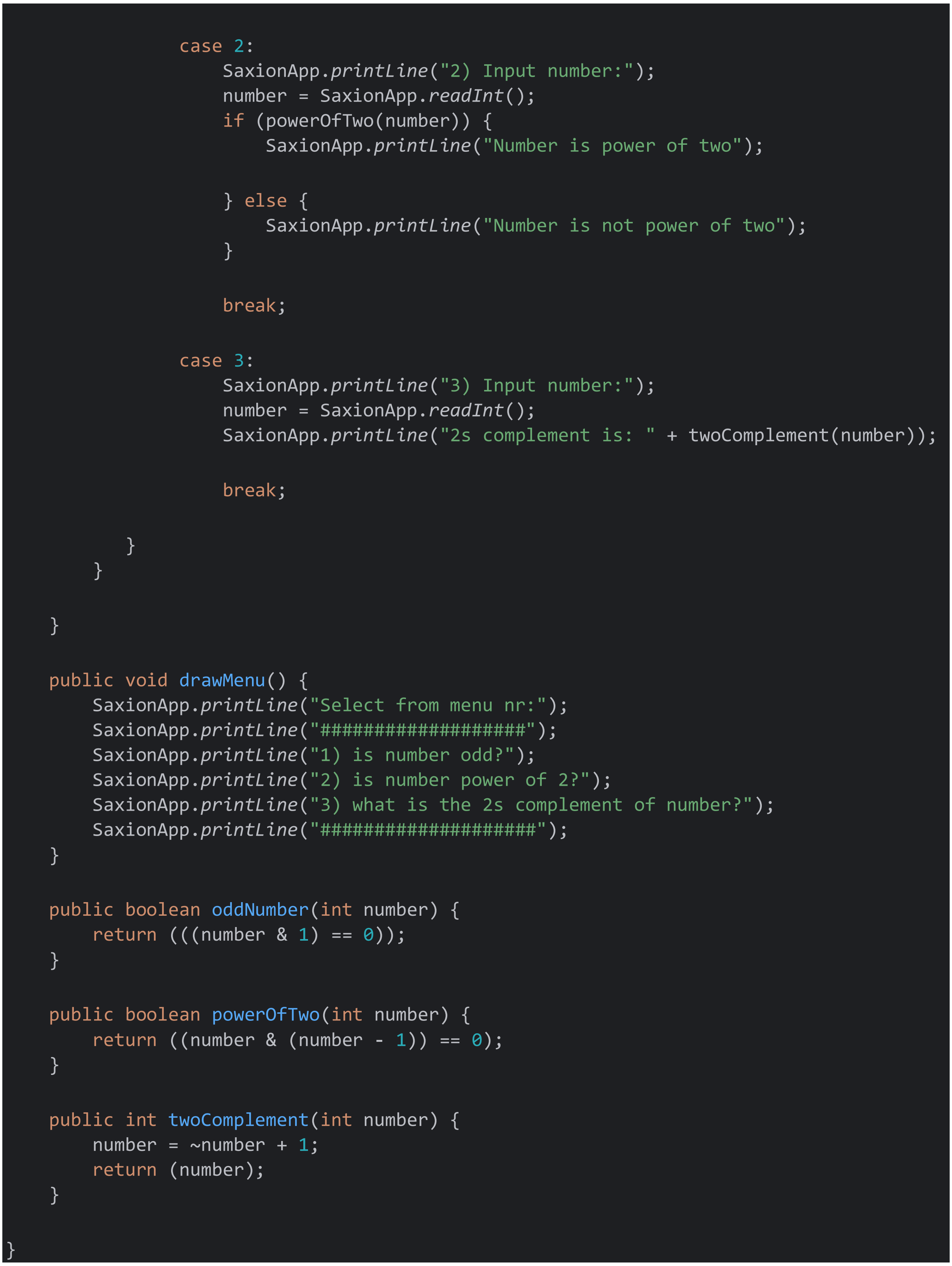
Implement the methods by using the bitwise operators you have just learned.

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

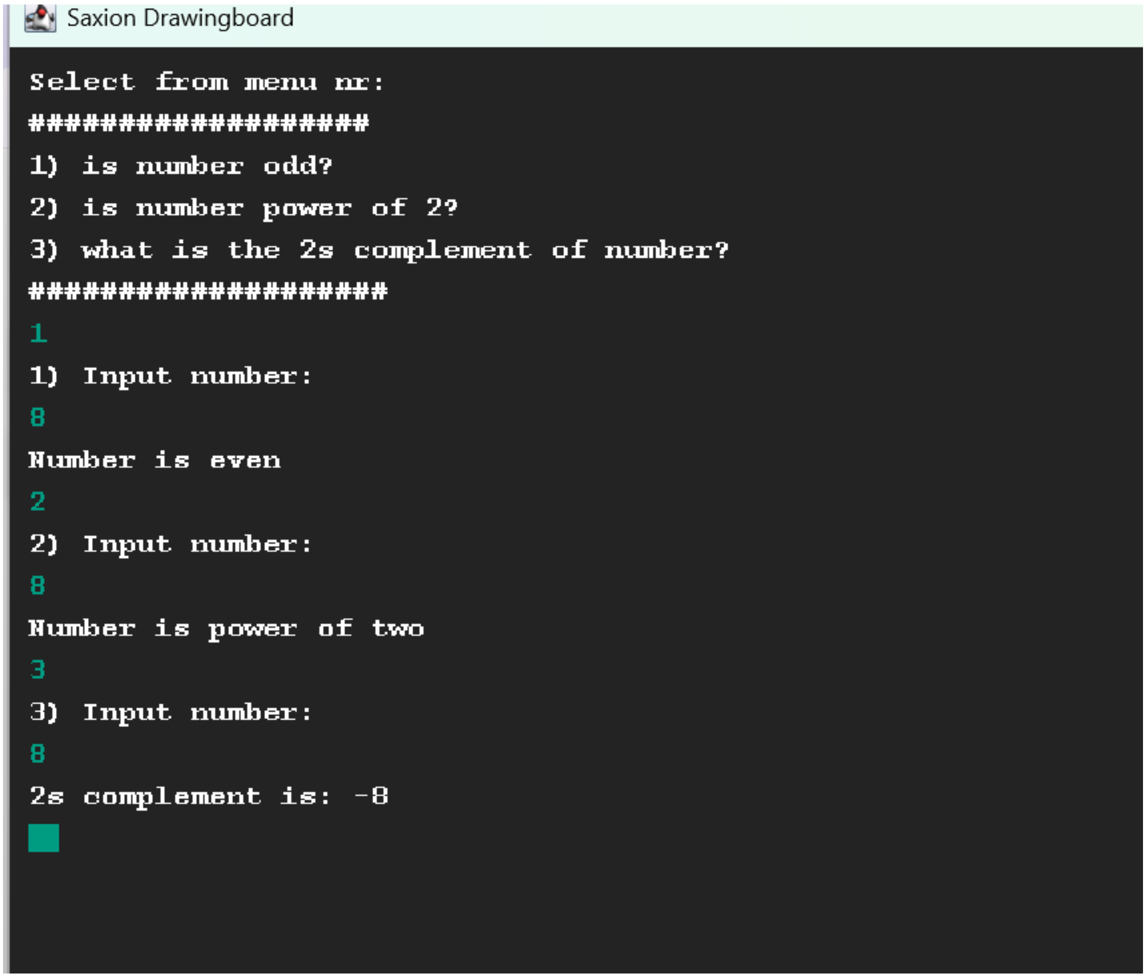
Keep this application because you need to expand it in week 6 for calculating network segments.

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

```java
import nl.saxion.app.SaxionApp;

public class Application implements Runnable {
    public static void main(String[] args) {
        SaxionApp.start(new Application(), 1024, 768);
    }

    public void run() {
        drawMenu();
        while (true) {
            int selection = SaxionApp.readInt();
            switch (selection) {

                case 1:
                    SaxionApp.printLine("1) Input number:");
                    int number = SaxionApp.readInt();
                    if (oddNumber(number)) {
                        SaxionApp.printLine("Number is even");

                    } else {
                        SaxionApp.printLine("Number is odd");
                    }

                    break;
```



```
                case 2:
                    SaxionApp.printLine("2) Input number:");
                    number = SaxionApp.readInt();
                    if (powerOfTwo(number)) {
                        SaxionApp.printLine("Number is power of two");

                    } else {
                        SaxionApp.printLine("Number is not power of two");
                    }

                    break;

                case 3:
                    SaxionApp.printLine("3) Input number:");
                    number = SaxionApp.readInt();
                    SaxionApp.printLine("2s complement is: " + twoComplement(number));

                    break;

            }
        }

    }

    public void drawMenu() {
        SaxionApp.printLine("Select from menu nr:");
        SaxionApp.printLine("###################");
        SaxionApp.printLine("1) is number odd?");
        SaxionApp.printLine("2) is number power of 2?");
        SaxionApp.printLine("3) what is the 2s complement of number?");
        SaxionApp.printLine("####################");
    }

    public boolean oddNumber(int number) {
        return (((number & 1) == 0));
    }

    public boolean powerOfTwo(int number) {
        return ((number & (number - 1)) == 0);
    }

    public int twoComplement(int number) {
        number = ~number + 1;
        return (number);
    }

}
```



```
Select from menu nr:
###################
1) is number odd?
2) is number power of 2?
3) what is the 2s complement of number?
####################
1
1) Input number:
8
Number is even
2
2) Input number:
8
Number is power of two
3
3) Input number:
8
2s complement is: -8
```

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