Template Week 6 – Networking

Student number:

Assignment 6.1: Working from home

Screenshot installation openssh-server:

Screenshot successful SSH command execution:

Screenshot successful execution SCP command:

Screenshot remmina:

Assignment 6.2: IP addresses websites

Relevant screenshots nslookup command:

Screenshot website visit via IP address:

Assignment 6.3: subnetting

How many IP addresses are in this network configuration 192.168.110.128/25?

What is the usable IP range to hand out to the connected computers?

Check your two previous answers with this calculator:
https://www.calculator.net/ip-subnet-calculator.html

Explain the above calculation in your own words.

Assignment 6.4: HTML

Screenshot IP address Ubuntu VM:

Screenshot of Site directory contents:

Screenshot python3 webserver command:

Screenshot web browser visits your site

Bonus point assignment – week 6

Remember that bitwise java application you've made in week 2? Expand that application so that you can also calculate a network segment as explained in the PowerPoint slides of week 6. Use the bitwise & AND operator. You need to be able to input two Strings. An IP address and a subnet.

IP: 192.168.1.100 and subnet: 255.255.255.224 for /27

```
Example: 192.168.1.100/27
Calculate the network segment
IP Address:   11000000.10101000.00000001.01100100
Subnet Mask:  11111111.11111111.11111111.11100000
-----------------------------------------------
Network Addr: 11000000.10101000.00000001.01100000

This gives 192.168.1.96 in decimal as the network address.
For a /27 subnet, each segment (or subnet) has 32 IP addresses (2⁵).
The range of this network segment is from 192.168.1.96 to 192.168.1.127.
```

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

```
import java.util.Scanner;

public class Main {
   public static void main(String[] args) {
      Scanner scanner = new Scanner(System.in);
```

```
    Main mainInstance = new Main();

    while (true) {
        System.out.println("\nMenu:");
        System.out.println("1. Is number odd?");
        System.out.println("2. Is number a power of 2?");
        System.out.println("3. Two's complement of number?");
        System.out.println("4. Calculate network segment");
        System.out.println("5. Exit");
        System.out.print("Enter your choice: ");

        int choice = scanner.nextInt();
        if (choice == 5) {
            System.out.println("Exiting program. Goodbye!");
            break;
        }

        switch (choice) {
            case 1:
                System.out.print("Enter a number: ");
                int number = scanner.nextInt();
                mainInstance.isOdd(number);
                break;
            case 2:
                System.out.print("Enter a number: ");
                number = scanner.nextInt();
                mainInstance.isPowerOfTwo(number);
                break;
            case 3:
                System.out.print("Enter a number: ");
                number = scanner.nextInt();
                mainInstance.twoComplement(number);
                break;
            case 4:
                System.out.print("Enter IP Address (e.g., 192.168.1.100): ");
                scanner.nextLine(); // Consume newline
                String ipAddress = scanner.nextLine();
                System.out.print("Enter Subnet Mask (e.g., 255.255.255.224): ");
                String subnetMask = scanner.nextLine();
                mainInstance.calculateNetworkSegment(ipAddress, subnetMask);
                break;
            default:
                System.out.println("Invalid choice. Please try again.");
        }
    }

    scanner.close();
}
```

```
  public void isOdd(int number) {
    if ((number & 1) == 1) {
      System.out.println(number + " is odd.");
    } else {
      System.out.println(number + " is even.");
    }
  }

  public void isPowerOfTwo(int number) {
    if (number > 0 && (number & (number - 1)) == 0) {
      System.out.println(number + " is a power of 2.");
    } else {
      System.out.println(number + " isn't a power of 2.");
    }
  }

  public void twoComplement(int number) {
    System.out.println("Original Number: " + number);

    int negative = ~number + 1;
    System.out.println("After Two's Complement (Negative): " + negative);
  }

  public void calculateNetworkSegment(String ipAddress, String subnetMask) {
    try {
      String[] ipParts = ipAddress.split("\\.");
      String[] maskParts = subnetMask.split("\\.");

      if (ipParts.length != 4 || maskParts.length != 4) {
        System.out.println("Invalid IP address or subnet mask format.");
        return;
      }

      int[] ip = new int[4];
      int[] mask = new int[4];
      int[] network = new int[4];

      for (int i = 0; i < 4; i++) {
        ip[i] = Integer.parseInt(ipParts[i]);
        mask[i] = Integer.parseInt(maskParts[i]);
        network[i] = ip[i] & mask[i];
      }

      System.out.println("Network Address: " + network[0] + "." + network[1] + "." + network[2] + "."
+ network[3]);

      int subnetBits = 0;
```

```
        for (int part : mask) {
            while (part > 0) {
                subnetBits += part & 1;
                part >>= 1;
            }
        }

        int totalHosts = (int) Math.pow(2, 32 - subnetBits);
        int usableHosts = totalHosts - 2; // Subtract network and broadcast addresses

        System.out.println("Subnet has " + totalHosts + " total addresses.");
        System.out.println("Usable range: " + network[0] + "." + network[1] + "." + network[2] + "." +
(network[3] + 1)
                + " to " + network[0] + "." + network[1] + "." + network[2] + "." + (network[3] +
usableHosts));
    } catch (Exception e) {
        System.out.println("Error calculating network segment: " + e.getMessage());
    }
  }
}
```

```
Menu:
1. Is number odd?
2. Is number a power of 2?
3. Two's complement of number?
4. Calculate network segment
5. Exit
Enter your choice: 4
Enter IP Address (e.g., 192.168.1.100): 192.168.1.100
Enter Subnet Mask (e.g., 255.255.255.224): 255.255.255.224
Network Address: 192.168.1.96
Subnet has 32 total addresses.
Usable range: 192.168.1.97 to 192.168.1.126
```

Ready? Save this file and export it as a pdf file with the name: **week6.pdf**