BansilalRamnathAgarwalCharitableTrust’s

# VishwakarmaInstituteofTechnology, Pune-37

*(AnautonomousInstituteofSavitribaiPhulePuneUniversity)*

VISHWAKARMA INSTITUTES VI

## ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

| | |
|---|---|
| **Division** | **AI-B** |
| **Batch** | **1** |
| **GR-no** | **12320050** |
| **Rollno** | **75** |
| **Name** | **Rushikesh Nitin Tanpure** |

**Lab Assignment:** 8

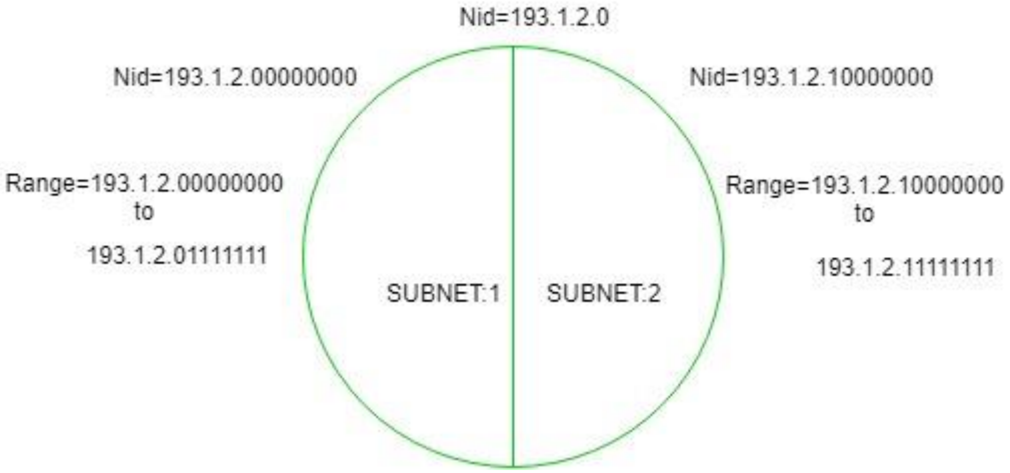
**Problem Statement:** Write a program to demonstrate subnetting and find the subnet masks..

When a bigger network is divided into smaller networks, to maintain security, then that is known as Subnetting. So, maintenance is easier for smaller networks. For example, if we consider a class A address, the possible number of hosts is 224 for each network, it is obvious that it is difficult to maintain such a huge number of hosts, but it would be quite easier to maintain if we divide the network into small parts.

Uses of Subnetting

- Subnetting helps in organizing the network in an efficient way which helps in expanding the technology for large firms and companies.
- Subnetting is used for specific staffing structures to reduce traffic and maintain order and efficiency.
- Subnetting divides domains of the broadcast so that traffic is routed efficiently, which helps in improving network performance.
- Subnetting is used in increasing network security.

The network can be divided into two parts: To divide a network into two parts, you need to choose one bit for each Subnet from the host ID part.



## How Does Subnetting Work?

The working of subnets starts in such a way that firstly it divides the subnets into smaller subnets. For communicating between subnets, routers are used. Each subnet allows its linked devices to communicate with each other. Subnetting for a network should be done in such a way that it does not affect the network bits.

In class C the first 3 octets are network bits so it remains as it is.

- **For Subnet-1:** The first bit which is chosen from the host id part is zero and the range will be from (193.1.2.00000000 till you get all 1's in the host ID part i.e, 193.1.2.01111111) except for the first bit which is chosen zero for subnet id part.

Thus, the range of subnet 1 is: **193.1.2.0 to 193.1.2.127**

```
Subnet id of Subnet-1 is : 193.1.2.0
The direct Broadcast id of Subnet-1 is: 193.1.2.127
The total number of hosts possible is: 126 (Out of 128,
2 id's are used for Subnet id & Direct Broadcast id)
The subnet mask of Subnet- 1 is: 255.255.255.128
```

- **For Subnet-2:** The first bit chosen from the host id part is one and the range will be from (193.1.2.100000000 till you get all 1's in the host ID part i.e, 193.1.2.11111111).

Thus, the range of subnet-2 is: **193.1.2.128 to 193.1.2.255**

```
Subnet id of Subnet-2 is : 193.1.2.128
The direct Broadcast id of Subnet-2 is: 193.1.2.255
The total number of hosts possible is: 126 (Out of 128,
2 id's are used for Subnet id &  Direct Broadcast id)
The subnet mask of Subnet- 2 is: 255.255.255.128
The best way to find out the subnet mask of a subnet
is to set the fixed bit of host-id to 1 and the rest to 0.
```

Finally, after using the subnetting the total number of usable hosts is reduced from 254 to 252.

Code:-

```
import java.util.Scanner;

public class SubnetCalculator {

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

        // Get IP address and subnet prefix length from user
        System.out.print("Enter IP address (in format xxx.xxx.xxx.xxx): ");
        String ipAddress = scanner.next();
        System.out.print("Enter subnet prefix length: ");
        int subnetPrefixLength = scanner.nextInt();

        // Calculate subnet mask
        int[] subnetMask = calculateSubnetMask(subnetPrefixLength);

        // Display subnet mask
        System.out.println("Subnet Mask: " + subnetMask[0] + "." +
subnetMask[1] + "." + subnetMask[2] + "." + subnetMask[3]);

        // Display subnet ranges
        calculateSubnetRanges(ipAddress, subnetMask, subnetPrefixLength);
    }

    public static int[] calculateSubnetMask(int prefixLength) {
        int[] mask = new int[4];
        for (int i = 0; i < 4; i++) {
            int bits = Math.min(prefixLength, 8);
            mask[i] = (int) (Math.pow(2, 8) - Math.pow(2, 8 - bits));
            prefixLength -= 8;
        }
        return mask;
    }

    public static void calculateSubnetRanges(String ipAddress, int[]
subnetMask, int prefixLength) {
        String[] ipParts = ipAddress.split("\\.");
        int[] ipOctets = new int[4];
        for (int i = 0; i < 4; i++) {
            ipOctets[i] = Integer.parseInt(ipParts[i]);
        }

        // Calculate network address
        int[] networkAddress = new int[4];
        for (int i = 0; i < 4; i++) {
            networkAddress[i] = ipOctets[i] & subnetMask[i];
        }

        // Calculate broadcast address
        int[] broadcastAddress = new int[4];
        for (int i = 0; i < 4; i++) {
            broadcastAddress[i] = networkAddress[i] | ~subnetMask[i];
        }

        // Display network address and broadcast address
        System.out.println("Network Address: " + networkAddress[0] + "." +
networkAddress[1] + "." + networkAddress[2] + "." + networkAddress[3]);
        System.out.println("Broadcast Address: " + broadcastAddress[0] + "." +
broadcastAddress[1] + "." + broadcastAddress[2] + "." + broadcastAddress[3]);
    }
}
```

Output:-

```
$ java SubnetCalculator

Enter IP address (in format xxx.xxx.xxx.xxx): 192.168.1.1

Enter subnet prefix length: 24

Subnet Mask: 255.255.255.0, Network Address: 192.168.1.0, Broadcast Address: 192.168.1.255
```