Batch: B-1 **Experiment Number: 5 - Classification of IP addressing**

Roll Number: 16010422234 **Name: Chandana Ramesh Galgali**

Aim of the Experiment: To write a program to identify the class to which a given IP Address belongs to.

Program/ Steps:

1. The program should accept the input IP address in dotted decimal form.
2. Convert this address into binary form and apply the classification algorithm.
3. Display the class of IP address as the output.

Output/Result:

Code:

```
import re

def classify_ip_address(b):
    y= b.split('.')
    z=[format(int(x),'08b') for x in y]
    if int(y[0])>255 or int(y[1])>255 or int(y[2])>255 or int(y[3])>255:
        print("Invalid IP address!")
    binary_ip_address=''.join(z)
    print(binary_ip_address)
    first_bit=binary_ip_address[0]
    second_bit=binary_ip_address[1]
    third_bit=binary_ip_address[2]
    fourth_bit=binary_ip_address[3]
    if first_bit=='0':
        ip_class='A'
    elif first_bit=='1' and second_bit=='0':
        ip_class='B'
    elif first_bit=='1' and second_bit=='1' and third_bit=='0':
        ip_class='C'
    elif first_bit=='1' and second_bit=='1' and third_bit=='1' and
fourth_bit=='0':
        ip_class='D'
```

```
    elif first_bit=='1' and second_bit=='1' and third_bit=='1' and
fourth_bit=='1':
        ip_class='E'
    else:
        ip_class='Invalid!'
    return ip_class

if __name__ == '__main__':
    ip_address = input("Enter the IP address in dotted decimal
form(0.0.0.0 to 255.255.255.255): ")
    if
(re.search(r'[0-2]?[0-9]?[0-9][.][0-2]?[0-9]?[0-9][.][0-2]?[0-9]?[0-9][.][
0-2]?[0-9]?[0-9]',ip_address)):
        a=ip_address.split('.')
        if int(a[0])>255 or int(a[1])>255 or int(a[2])>255 or
int(a[3])>255:
            print("Invalid IP address!")
        else:
            ip_class=classify_ip_address(ip_address)
            print("The class of the IP address is: ", ip_class)
    else:
        print("Invalid IP address!")
```

Output:

```
Enter the IP address in dotted decimal form(0.0.0.0 to 255.255.255.255): 13.51.250.24
00001101001100111111101000011000
The class of the IP address is:  A
```

```
Enter the IP address in dotted decimal form(0.0.0.0 to 255.255.255.255): 129.56.79.20
10000001001110000100111100010100
The class of the IP address is:  B
```

```
Enter the IP address in dotted decimal form(0.0.0.0 to 255.255.255.255): 201.23.45.19
11001001000101110010110100010011
The class of the IP address is:  C
```

```
Enter the IP address in dotted decimal form(0.0.0.0 to 255.255.255.255): 235.76.86.87
11101011010011000101011001010111
The class of the IP address is:  D
```

```
Enter the IP address in dotted decimal form(0.0.0.0 to 255.255.255.255): 245.15.30.45
11110101000011110001111000101101
The class of the IP address is:  E
```

```
Enter the IP address in dotted decimal form(0.0.0.0 to 255.255.255.255): 300.45.67.89
Invalid IP address!
```

Post Lab Question-Answers:

1) Which OSI layer corresponds to the IP Layer?

Ans. The IP (Internet Protocol) layer corresponds to the Network layer in the OSI (Open Systems Interconnection) model. The Network layer is responsible for addressing, routing, and delivering data packets across different networks. IP is a network protocol that operates at this layer and provides the necessary functions for packet forwarding and addressing in the Internet.

2) Compare IPv4 and IPv6 header.

Ans. IPv4 and IPv6 are two versions of the Internet Protocol, and they have some differences in their header formats. Here's a comparison of the IPv4 and IPv6 headers:

1. Header Length:
    - IPv4: The header length is fixed at 20 bytes.
    - IPv6: The header length is fixed at 40 bytes.
2. Addressing:
    - IPv4: It uses 32-bit addresses, represented in decimal format (e.g., 192.168.0.1).
    - IPv6: It uses 128-bit addresses, represented in hexadecimal format (e.g., 2001:0db8:85a3:0000:0000:8a2e:0370:7334).
3. Header Fields:
    - IPv4: The header includes fields such as source and destination IP addresses, protocol type, time-to-live (TTL), checksum, and options (if any).
    - IPv6: The header includes fields such as source and destination IP addresses, traffic class, flow label, payload length, next header, and hop limit.
4. Fragmentation:
    - IPv4: It supports fragmentation at the sender and reassembly at the receiver.
    - IPv6: It does not support fragmentation at the sender; instead, it relies on the network layer to handle fragmentation if necessary.
5. Options:
    - IPv4: It supports variable-length options in the header.
    - IPv6: It uses extension headers for optional information, which are placed after the main header.

6. Security:
    - IPv4: It does not have built-in security features, although additional protocols like IPsec can be used for security.
    - IPv6: It includes built-in support for IPsec, which provides authentication and encryption for network traffic.

These are some of the key differences between the IPv4 and IPv6 headers. IPv6 was developed to address the limitations of IPv4, such as address exhaustion and the need for enhanced security features.

3) What is fragmentation?

Ans. Fragmentation is a process in computer networking where large data packets are divided or fragmented into smaller pieces called fragments. This process occurs when the maximum transmission unit (MTU) size of a network link is smaller than the size of the original packet. When a packet is too large to be transmitted over a network link with a smaller MTU, it needs to be fragmented into smaller fragments that can fit within the MTU limit. The process of fragmentation involves breaking the original packet into smaller fragments at the sender's end. Each fragment contains a portion of the original packet's data, along with a fragment header that provides information for reassembling the fragments at the receiver's end.

At the receiver's end, the fragments are reassembled based on the information in the fragment headers. The reassembly process involves arranging the fragments in the correct order and combining them to reconstruct the original packet.

Fragmentation can occur in IPv4 networks, where it is supported by the protocol. In IPv6 networks, fragmentation is generally avoided, and it is the responsibility of the sender to ensure that packets are appropriately sized to fit within the MTU of the network links along the path.

Fragmentation can introduce additional overhead and processing requirements on both the sender and receiver, and it can also impact network performance. Therefore, it is generally preferred to avoid fragmentation whenever possible by using Path MTU Discovery techniques to determine the maximum MTU size along the network path and adjust the packet size accordingly.

4) What is Subnetting?

Ans. Subnetting is the process of dividing a single network into smaller subnetworks, known as subnets. It is a technique used in computer networking to efficiently allocate IP addresses and manage network resources.

In subnetting, a network administrator takes a larger network, typically identified by its network address and subnet mask, and further divides it into smaller subnets. Each subnet is assigned a unique subnet address and subnet mask, allowing it to function as an independent network within the larger network.

The primary benefits of subnetting include:

1. Efficient IP Address Allocation: Subnetting allows for the efficient allocation of IP addresses by dividing a large network into smaller subnets. This helps conserve IP address space and ensures that addresses are used effectively.

2. Improved Network Performance: By dividing a large network into smaller subnets, network traffic can be localized within each subnet. This reduces the amount of broadcast traffic and improves network performance by limiting the scope of network communications.

3. Enhanced Network Security: Subnetting can enhance network security by isolating different departments, segments, or devices into separate subnets. This allows for the implementation of access control policies and improves network security by limiting the impact of potential security breaches.

4. Simplified Network Management: Subnetting simplifies network management by providing logical boundaries for network administration. Each subnet can be managed independently, allowing for easier troubleshooting, configuration, and maintenance.

Subnetting is commonly used in IP networks, particularly with IPv4 addressing. It involves dividing the host portion of an IP address into subnet and host identifiers, allowing for efficient routing and management of network resources.

5) What is Supernetting?

Ans. Supernetting, also known as route aggregation or route summarization, is a technique used in computer networking to combine multiple smaller networks into a larger network. It is the opposite of subnetting, where subnetting divides a larger network into smaller subnets.
In supernetting, contiguous network addresses with the same network prefix are combined to create a single, larger network. This is done by extending the network prefix or subnet mask to encompass multiple smaller networks. By doing so, the number of routing table entries can be reduced, leading to more efficient routing and improved network scalability.

The primary benefits of supernetting include:

1. Reduced Routing Table Size: By aggregating multiple smaller networks into a supernet, the number of entries in the routing table can be significantly reduced. This helps to

conserve memory and processing resources on routers and improves the efficiency of routing protocols.

2. Improved Routing Efficiency: Supernetting allows for more efficient routing by reducing the number of routing table lookups required for forwarding packets. With fewer entries in the routing table, routers can make routing decisions more quickly, leading to improved network performance.

3. Simplified Network Design: Supernetting simplifies network design by reducing the complexity of the routing infrastructure. Instead of maintaining individual routes for each smaller network, a single supernet route can be used, resulting in a more streamlined and manageable network architecture.

It's important to note that supernetting requires careful planning and consideration to ensure that the aggregated networks have compatible characteristics and routing policies. Additionally, supernetting can introduce challenges in terms of network address allocation and potential for increased broadcast traffic within the supernet.

Supernetting is commonly used in Internet Service Provider (ISP) networks and large enterprise networks to optimize routing and improve scalability. It is typically implemented using Border Gateway Protocol (BGP) for inter-domain routing or Interior Gateway Protocols (IGPs) such as OSPF or IS-IS for intra-domain routing.

Outcomes: Build the skills of sub-netting and routing mechanisms.

Conclusion (based on the Results and outcomes achieved):

The experiment successfully achieved its aim of developing a program to identify the class of a given IP address. The program can be used as a tool for network administrators or developers to quickly determine the class of an IP address, which can be useful for network planning, troubleshooting, and configuration purposes.

References:

Books/ Journals/ Websites:

1. Behrouz A Forouzan, Data Communication and Networking, Tata Mc Graw hill, India, 4th Edition
2. A. S. Tanenbaum, ”Computer Networks”, 4th edition, Prentice Hall