PROGRAM 5

- ASHISH KUMAR
- 2K18/SE/041

Aim:- Write a C++ program to implement and find Network Class, Network Id and Host Id of given IPV4 address.

Theory:- Given a valid IPv4 address in the form of string and it follows Class Full addressing. The 32 bit IP address is divided into five subclasses. These are:
1. Class A
2. Class B
3. Class C
4. Class D
5. Class E

Each of these classes has a valid range of IP addresses. Classes D and E are reserved for multicast and experimental purposes respectively.

- For determining the Class: The idea is to check first octet of IP address. As we know, for class At first octet will range from 1 – 126, for class B first octet will range from 128 – 191, for class C first octet will range from 192- 223, for class D first octet will range from 224 – 239, for class E first octet will range from 240 – 255.
- For determining the Network and Host ID: We know that Subnet Mask for Class A is 8, for Class B is 16 and for Class C is 24 whereas Class D and E is not divided into Network and Host ID.

CODE:-

```
#include<iostream>

#include<bits/stdc++.h>

#include<string.h>

using namespace std;

char findClass(char str[])

{

char arr[4];

int i = 0;
```

```
while (str[i] != '.')
{
arr[i] = str[i];
i++;
}
i--;
int ip = 0, j = 1;
while (i >= 0)
{
ip = ip + (str[i] - '0') * j;
j = j * 10;
i--;
}
if (ip >=0 && ip <= 127)
return 'A';
else if (ip >= 128 && ip <= 191)
return 'B';
else if (ip >= 192 && ip <= 223)
return 'C';
else if (ip >= 224 && ip <= 239)
return 'D';
else
return 'E';
}
```

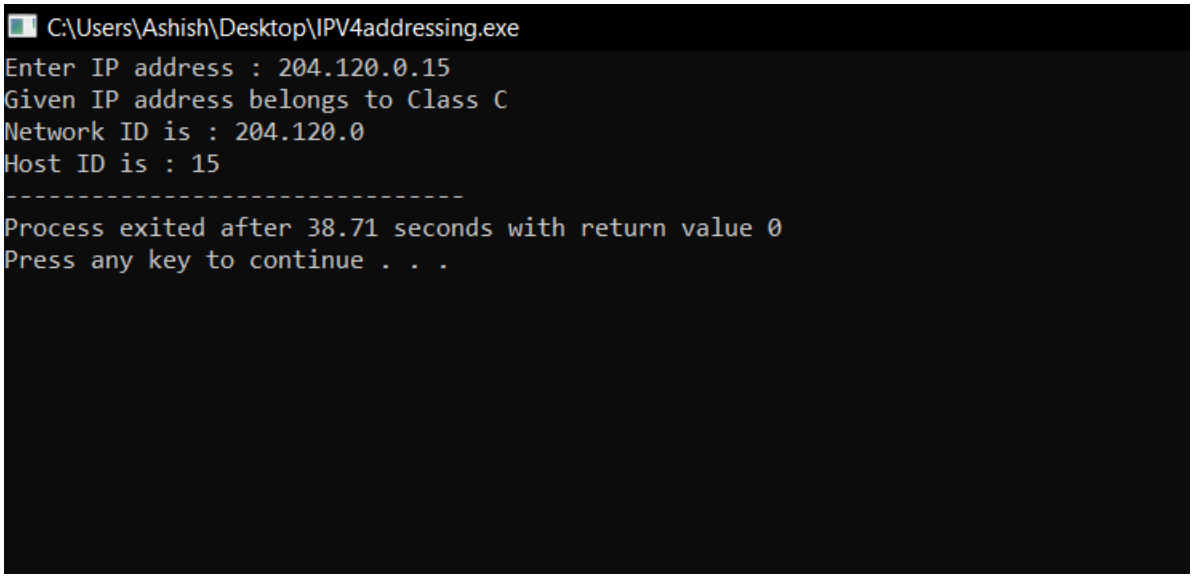
```
void separate(char str[], char ipClass)
{
char network[12], host[12];
for (int k = 0; k < 12; k++)
network[k] = host[k] = '\0';
if (ipClass == 'A')
{
int i = 0, j = 0;
while (str[j] != '.')
network[i++] = str[j++];
i = 0;
j++;
while (str[j] != '\0')
host[i++] = str[j++];
std::cout<<"\nNetwork ID is : "<<network;
std::cout<<"\nHost ID is : "<<host;
}
else if (ipClass == 'B')
{
int i = 0, j = 0, dotCount = 0;
while (dotCount < 2)
{
network[i++] = str[j++];
if (str[j] == '.')
dotCount++;
```

```
}
i = 0;
j++;
while (str[j] != '\0')
host[i++] = str[j++];
std::cout<<"\nNetwork ID is : "<< network;
std::cout<<"\nHost ID is : "<< host;
}
else if (ipClass == 'C')
{
int i = 0, j = 0, dotCount = 0;
while (dotCount < 3)
{
network[i++] = str[j++];
if (str[j] == '.')
dotCount++;
}
i = 0;
j++;
while (str[j] != '\0')
host[i++] = str[j++];
std::cout<<"\nNetwork ID is : "<< network;
std::cout<<"\nHost ID is : "<< host;
}
```

```
else
std::cout<<"\nIn this Class, IP address is not"
"divided into Network and Host ID\n";
}

int main()
{
char str[50];
std::cout<<"Enter IP address : ";
std::cin>>str;
char ipClass = findClass(str);
std::cout<<"Given IP address belongs to Class "<<ipClass;
separate(str, ipClass);
return 0;
}
```

OUTPUT:-



Learning Outcome:- We have learned following things :
IPv4 address is divided into two parts:
• Network ID
• Host ID
The class of IP address is used to determine the bits used for network ID and host ID and the number of total networks and hosts possible in that particular class. Each ISP or network administrator assigns IP address to each device that is connected to its network.