

Name: Shaunak Sensarma

Registration No: 18BCE2527

Network and Communication CSE-1004

Digital Assignment LAB- 3

Faculty – Dr. Asis Kumar Tripathy

Question

Q1. For any class B ip address, go for 16 subnets.

For example, given a Class B network of 130.17.5.0 which has a natural mask of 255.255.0.0, you can create subnets in this manner:

Input: Take any class C ip address

Processing: Create 8 subnets and find the network address, first valid and last valid ip address in each subnet

Output: As shown below.

The below table is an example of a Class C ip address but your answer will not match it. It is just given for your reference.

Network ID	First Valid Host	Last Valid Host
204.17.5.0	204.17.5.1	204.17.5.30
204.17.5.32	204.17.5.33	204.17.5.62
204.17.5.64	204.17.5.65	204.17.5.94
204.17.5.96	204.17.5.97	204.17.5.126
204.17.5.128	204.17.5.129	204.17.5.158
204.17.5.160	204.17.5.161	204.17.5.190
204.17.5.192	204.17.5.193	204.17.5.222
204.17.5.224	204.17.5.225	204.17.5.254

Answer:

Code:

```
#include<stdio.h>

void main()
{
    printf("Enter the Class C IP Address with each block in different line without ' . ' \n");
    int a,b,c,d;
    scanf("%d",&a);
```

```
scanf("%d",&b);
scanf("%d",&c);
scanf("%d",&d);
printf("The given class C IP address is.....");
printf("%d.%d.%d.%d",a,b,c,d);
printf("\n\nDefault Class C Subnet Mask is 255.255.255.0\n");
int s1[8]={ 1,1,1,1,1,1,1,1};
int s2[8]={ 1,1,1,1,1,1,1,1};
int s3[8]={ 1,1,1,1,1,1,1,1};
int s4[8]={ 0,0,0,0,0,0,0,0};
```

```
int a1[8]; int b1[8]; int c1[8]; int d1[8];
```

```
int i;
```

```
for(i=7;a>0;i--)
```

```
{
```

```
    a1[i]=a%2;
```

```
    a=a/2;
```

```
}
```

```
for(i=7;b>0;i--)
```

```
{
```

```
    b1[i]=b%2;
```

```
    b=b/2;
```

```
}
```

```
for(i=7;c>0;i--)
```

```
{
```

```
    c1[i]=c%2;
```

```
    c=c/2;
```

```
}
```

```
for(i=7;d>0;i--)
```

```
{
```

```
    d1[i]=d%2;
```

```
    d=d/2;
```

```
}
```

```

int sn1[8]; int sn2[8]; int sn3[8]; int sn4[8];
for(i=0;i<8;i++)
{
    sn1[i]=a1[i]&s1[i];
    sn2[i]=b1[i]&s2[i];
    sn3[i]=c1[i]&s3[i];
    sn4[i]=d1[i]&s4[i];
}
printf("\nThe subnet mask for the given IP is....");
for(i=0;i<8;i++)
{
    printf("%d",sn1[i]);
}
printf(".");
for(i=0;i<8;i++)
{
    printf("%d",sn2[i]);
}
printf(".");
for(i=0;i<8;i++)
{
    printf("%d",sn3[i]);
}
printf(".");
for(i=0;i<8;i++)
{
    printf("%d",sn4[i]);
}
printf("\nThe Subnet Mask in decimal format is = ");
int dn1,dn2,dn3,dn4;
int base = 1;
for(i=7;i>=0;i--)
{
    dn1 = dn1 + sn1[i]* base;

```

```

    base = base * 2;
}
printf("%d.",dn1);
base = 1;
    for(i=7;i>=0;i--)
{
    dn2 = dn2 + sn2[i]* base;
    base = base * 2;
}
printf("%d.",dn2);
base = 1;
    for(i=7;i>=0;i--)
{
    dn3 = dn3 + sn3[i]* base;
    base = base * 2;
}
printf("%d.",dn3);
base = 1;
    for(i=7;i>=0;i--)
{
    dn4 = dn4 + sn4[i]* base;
    base = base * 2;
}
printf("%d",dn4);

printf("\n\nThe Network ID in binary format is....");

int n1[8]; int n2[8]; int n3[8]; int n4[8];
    for(i=0;i<8;i++)
    {
        n1[i]=a1[i]&sn1[i];
        n2[i]=b1[i]&sn2[i];
        n3[i]=c1[i]&sn3[i];
        n4[i]=d1[i]&sn4[i];
    }

```

```

    }
    for(i=0;i<8;i++)
    {
        printf("%d",n1[i]);
    }
    printf(".");
    for(i=0;i<8;i++)
    {
        printf("%d",n2[i]);
    }
    printf(".");
    for(i=0;i<8;i++)
    {
        printf("%d",n3[i]);
    }
    printf(".");
    for(i=0;i<8;i++)
    {
        printf("%d",n4[i]);
    }

    printf("\nThe Network ID in decimal format is = ");
dn1=0;dn2=0;dn3=0;dn4=0;
base=1;
    for(i=7;i>=0;i--)
    {
        dn1 = dn1 + n1[i]* base;
        base = base * 2;
    }
    printf("%d.",dn1);
    base = 1;
    for(i=7;i>=0;i--)
    {
        dn2 = dn2 + n2[i]* base;
        base = base * 2;
    }

```

```

    }
    printf("%d.",dn2);
    base = 1;
    for(i=7;i>=0;i--)
    {
        dn3 = dn3 + n3[i]* base;
        base = base * 2;
    }
    printf("%d.",dn3);
    base = 1;
    for(i=7;i>=0;i--)
    {
        dn4 = dn4 + n4[i]* base;
        base = base * 2;
    }
    printf("%d",dn4);

    printf("\n\nNetwork ID          First Valid ID          Last Valid ID\n");
    int cn=0;
    for(i=0;i<8;i++)
    {
        printf("\n%d.%d.%d.%d          %d.%d.%d.%d
        %d.%d.%d.%d",dn1,dn2,dn3,(dn4+cn),dn1,dn2,dn3,(dn4+1+cn),dn1,dn2,dn3,(dn4+30+cn));
        cn=cn+32;
    }
}

```

Snapshots of Code:

```
IP.c
1  #include<stdio.h>
2  void main()
3  {
4      printf("Enter the Class C IP Address with each block in different line without ' . ' \n");
5      int a,b,c,d;
6      scanf("%d",&a);
7      scanf("%d",&b);
8      scanf("%d",&c);
9      scanf("%d",&d);
10     printf("The given class C IP address is.....");
11     printf("%d.%d.%d.%d",a,b,c,d);
12     printf("\n\nDefault Class C Subnet Mask is 255.255.255.0\n");
13     int s1[8]={1,1,1,1,1,1,1,1};
14     int s2[8]={1,1,1,1,1,1,1,1};
15     int s3[8]={1,1,1,1,1,1,1,1};
16     int s4[8]={0,0,0,0,0,0,0,0};
17
18     int a1[8]; int b1[8]; int c1[8]; int d1[8];
19     int i;
20     for(i=7;a>0;i--)
21     {
22         a1[i]=a%2;
23         a=a/2;
24     }
25     for(i=7;b>0;i--)
26     {
27         b1[i]=b%2;
28         b=b/2;
29     }
30     for(i=7;c>0;i--)
31     {
32         c1[i]=c%2;
33         c=c/2;
34     }
35     for(i=7;d>0;i--)
36     {
37         d1[i]=d%2;
38         d=d/2;
39     }
40
41     int sn1[8]; int sn2[8]; int sn3[8]; int sn4[8];
42     for(i=0;i<8;i++)
43     {
44         sn1[i]=a1[i]&s1[i];
45         sn2[i]=b1[i]&s2[i];
46         sn3[i]=c1[i]&s3[i];
47         sn4[i]=d1[i]&s4[i];
48     }
49     printf("\nThe subnet mask for the given IP is.....");
50     for(i=0;i<8;i++)
51     {
52         printf("%d", sn1[i]);
53     }
```

Compiler Resources Compile Log Debug Find Results Close

Abort Compilation - Compilation Time: 1.25s

```
IP.c
25     for(i=7;b>0;i--)
26     {
27         b1[i]=b%2;
28         b=b/2;
29     }
30     for(i=7;c>0;i--)
31     {
32         c1[i]=c%2;
33         c=c/2;
34     }
35     for(i=7;d>0;i--)
36     {
37         d1[i]=d%2;
38         d=d/2;
39     }
40
41     int sn1[8]; int sn2[8]; int sn3[8]; int sn4[8];
42     for(i=0;i<8;i++)
43     {
44         sn1[i]=a1[i]&s1[i];
45         sn2[i]=b1[i]&s2[i];
46         sn3[i]=c1[i]&s3[i];
47         sn4[i]=d1[i]&s4[i];
48     }
49     printf("\nThe subnet mask for the given IP is.....");
50     for(i=0;i<8;i++)
51     {
52         printf("%d", sn1[i]);
53     }
```

Compiler Resources Compile Log Debug Find Results Close


- Compilation Time: 1.25s




(globals)

IP.c

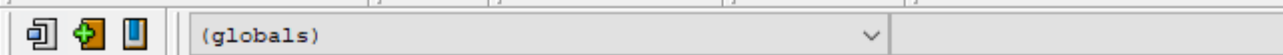
```
52     printf("%d",sn1[i]);
53 }
54 printf(".");
55 for(i=0;i<8;i++)
56 {
57     printf("%d",sn2[i]);
58 }
59 printf(".");
60 for(i=0;i<8;i++)
61 {
62     printf("%d",sn3[i]);
63 }
64 printf(".");
65 for(i=0;i<8;i++)
66 {
67     printf("%d",sn4[i]);
68 }
69 printf("\nThe Subnet Mask in decimal format is = ");
70 int dn1,dn2,dn3,dn4;
71 int base = 1;
72 for(i=7;i>=0;i--)
73 {
74     dn1 = dn1 + sn1[i]* base;
75     base = base * 2;
76 }
77 printf("%d.",dn1);
78 base = 1;
79 for(i=7;i>=0;i--)
80 {
81     dn2 = dn2 + sn2[i]* base;
82     base = base * 2;
83 }
```

Line: 161 Col: 14 Sel: 0 Lines: 167 Length: 3374 Insert





```
82     base = base * 2;
83 }
84 printf("%d.",dn2);
85 base = 1;
86 for(i=7;i>=0;i--)
87 {
88     dn3 = dn3 + sn3[i]* base;
89     base = base * 2;
90 }
91 printf("%d.",dn3);
92 base = 1;
93 for(i=7;i>=0;i--)
94 {
95     dn4 = dn4 + sn4[i]* base;
96     base = base * 2;
97 }
98 printf("%d",dn4);
```

(globals)

IP.c

```
97     }
98     printf("%d",dn4);
99
100    printf("\n\nThe Network ID in binary format is....");
101
102    int n1[8]; int n2[8]; int n3[8]; int n4[8];
103    for(i=0;i<8;i++)
104    {
105        n1[i]=a1[i]&sn1[i];
106        n2[i]=b1[i]&sn2[i];
107        n3[i]=c1[i]&sn3[i];
108        n4[i]=d1[i]&sn4[i];
109    }
110    for(i=0;i<8;i++)
111    {
112        printf("%d",n1[i]);
113    }
114    printf(".");
115    for(i=0;i<8;i++)
116    {
117        printf("%d",n2[i]);
118    }
119    printf(".");
120    for(i=0;i<8;i++)
121    {
122        printf("%d",n3[i]);
123    }
124    printf(".");
125    for(i=0;i<8;i++)
126    {
127        printf("%d",n4[i]);
128    }
```

Line: 161

Col: 14

Sel: 0

Lines: 167

Length: 3374

Insert

Done parsing ir

```
126    {
127        printf("%d",n4[i]);
128    }
129    printf("\n\nThe Network ID in decimal format is = ");
130    dn1=0;dn2=0;dn3=0;dn4=0;
131    base=1;
132    for(i=7;i>=0;i--)
133    {
134        dn1 = dn1 + n1[i]* base;
135        base = base * 2;
136    }
137    printf("%d.",dn1);
138    base = 1;
139    for(i=7;i>=0;i--)
140    {
141        dn2 = dn2 + n2[i]* base;
142        base = base * 2;
143    }
```

```

142     base = base * 2;
143 }
144 printf("%d.",dn2);
145 base = 1;
146 for(i=7;i>=0;i--)
147 {
148     dn3 = dn3 + n3[i]* base;
149     base = base * 2;
150 }
151 printf("%d.",dn3);
152 base = 1;
153 for(i=7;i>=0;i--)
154 {
155     dn4 = dn4 + n4[i]* base;
156     base = base * 2;
157 }
158 printf("%d",dn4);
159
160 printf("\n\nNetwork ID      First Valid ID      Last Valid ID\n");
161 int cn=0;
162 for(i=0;i<8;i++)
163 {
164     printf("\n%d.%d.%d.%d      %d.%d.%d.%d      %d.%d.%d.%d",dn1,dn2,dn3,(dn4+cn),dn1,dn2,dn3,(dn4+1+cn),dn1,dn2,dn3,(dn4+30+cn));
165     cn=cn+32;
166 }
167 }

```

Compiler Resources Compile Log Debug Find Results Close

Abort Compilation

- Output Size: 131.9423828125 KiB
- Compilation Time: 1.25s

Snapshots of Output:

Output-1:

```

C:\Users\Shaunak_Sensarma\Desktop\IP.exe
Enter the Class C IP Address with each block in different line without ' . '
212
100
196
12
The given class C IP address is.....212.100.196.12

Default Class C Subnet Mask is 255.255.255.0

The subnet mask for the given IP is....11010100.01100100.11000100.00000000
The Subnet Mask in decimal format is = 213.100.237.0

The Network ID in binary format is....11010100.01100100.11000100.00000000
The Network ID in decimal format is = 212.100.196.0

Network ID      First Valid ID      Last Valid ID

212.100.196.0      212.100.196.1      212.100.196.30
212.100.196.32      212.100.196.33      212.100.196.62
212.100.196.64      212.100.196.65      212.100.196.94
212.100.196.96      212.100.196.97      212.100.196.126
212.100.196.128      212.100.196.129      212.100.196.158
212.100.196.160      212.100.196.161      212.100.196.190
212.100.196.192      212.100.196.193      212.100.196.222
212.100.196.224      212.100.196.225      212.100.196.254
-----
Process exited after 9.129 seconds with return value 50
Press any key to continue . . .

```

Output- 2:

```
C:\Users\Shaunak_Sensarma\Desktop\IP.exe
Enter the Class C IP Address with each block in different line without ' . '
196
189
248
0
The given class C IP address is.....196.189.248.0

Default Class C Subnet Mask is 255.255.255.0

The subnet mask for the given IP is....11000100.10111101.11111000.00000000
The Subnet Mask in decimal format is = 197.189.289.0

The Network ID in binary format is....11000100.10111101.11111000.00000000
The Network ID in decimal format is = 196.189.248.0

Network ID                First Valid ID            Last Valid ID
196.189.248.0             196.189.248.1            196.189.248.30
196.189.248.32            196.189.248.33            196.189.248.62
196.189.248.64            196.189.248.65            196.189.248.94
196.189.248.96            196.189.248.97            196.189.248.126
196.189.248.128           196.189.248.129           196.189.248.158
196.189.248.160           196.189.248.161           196.189.248.190
196.189.248.192           196.189.248.193           196.189.248.222
196.189.248.224           196.189.248.225           196.189.248.254
-----
Process exited after 8.937 seconds with return value 50
Press any key to continue . . .
```

Output3:

```
C:\Users\Shaunak_Sensarma\Desktop\IP.exe
Enter the Class C IP Address with each block in different line without ' . '
201
0
12
0
The given class C IP address is.....201.0.12.0

Default Class C Subnet Mask is 255.255.255.0

The subnet mask for the given IP is....11001001.00000001.00001100.00000000
The Subnet Mask in decimal format is = 202.1.53.0

The Network ID in binary format is....11001001.00000001.00001100.00000000
The Network ID in decimal format is = 201.1.12.0

Network ID                First Valid ID            Last Valid ID
201.1.12.0               201.1.12.1               201.1.12.30
201.1.12.32              201.1.12.33              201.1.12.62
201.1.12.64              201.1.12.65              201.1.12.94
201.1.12.96              201.1.12.97              201.1.12.126
201.1.12.128             201.1.12.129             201.1.12.158
201.1.12.160             201.1.12.161             201.1.12.190
201.1.12.192             201.1.12.193             201.1.12.222
201.1.12.224             201.1.12.225             201.1.12.254
-----
Process exited after 9.357 seconds with return value 41
Press any key to continue . . .
```