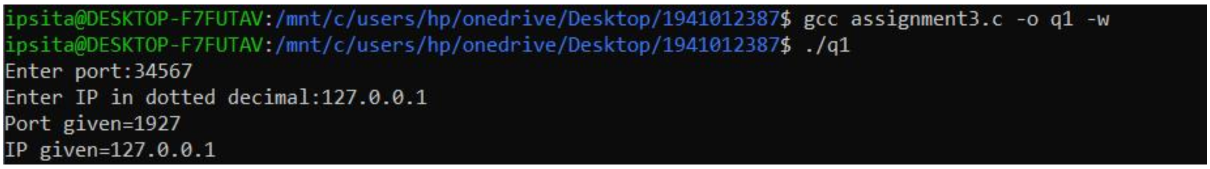
MINOR ASSIGNMENT-03
UNIX Network Programming (CSE 4042)

1. Write a program to create an IPV4 socket address structure and pack the structure with family
AF INET, port=34567 and IP address 127.0.0.0.1 respectively from the command line. Also, display the port and IP address

Code-

```
int main()
{
char ip[16];
in_port_t port;

struct sockaddr_in sa;
bzero(&sa,sizeof(sa));
printf("Enter port:");
scanf("%hu",&port);
printf("Enter IP in dotted decimal:");
scanf("%s",ip);

sa.sin_port=htons(port);
sa.sin_addr.s_addr=inet_addr(ip);

printf("Port given=%hu\n",sa.sin_port);
printf("IP given=%s\n",inet_ntoa(sa.sin_addr));
return 0;  }
```

Output snippet-



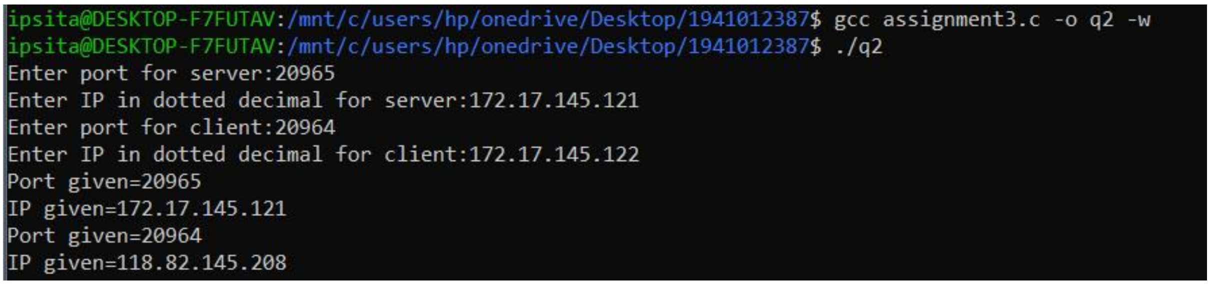
```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c -o q1 -w
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q1
Enter port:34567
Enter IP in dotted decimal:127.0.0.1
Port given=1927
IP given=127.0.0.1
```

2. Write a program to declare two Internet socket address structure, namely servaddr and cliaddr respectively. Read the port and IP address for the structures you have declared from the keyboard and display the port and IP address.

Code-

```
int main()
{
char ip_serv[16];
in_port_t port_serv;
printf("Enter port for server:");
scanf("%hu",&port_serv);
printf("Enter IP in dotted decimal for server:");
scanf("%s",ip_serv);

char ip_cli[16];
in_port_t port_cli;
printf("Enter port for client:");
scanf("%hu",&port_cli);
printf("Enter IP in dotted decimal for client:");
scanf("%s",ip_cli);

struct sockaddr_in servaddr;
servaddr.sin_port=htons(port_serv);
servaddr.sin_addr.s_addr=inet_addr(ip_serv);
printf("Port given=%hu\n",ntohs(servaddr.sin_port));
printf("IP given=%s\n",inet_ntoa(servaddr.sin_addr));
struct sockaddr_in cliaddr;
cliaddr.sin_port=htons(port_cli);
cliaddr.sin_addr.s_addr=htonl(ip_cli);
printf("Port given=%hu\n",ntohs(cliaddr.sin_port));
printf("IP given=%s\n",inet_ntoa(cliaddr.sin_addr));

return 0;
}
```

Output snippet-



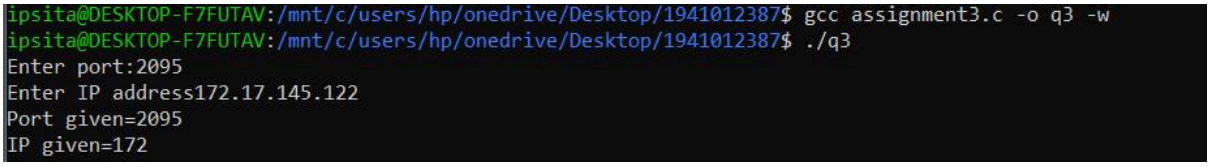
```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c -o q2 -w
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q2
Enter port for server:20965
Enter IP in dotted decimal for server:172.17.145.121
Enter port for client:20964
Enter IP in dotted decimal for client:172.17.145.122
Port given=20965
IP given=172.17.145.121
Port given=20964
IP given=118.82.145.208
```

3. Create a structure variable of the structure type struct sockaddr_in defined in the header <netinet/in.h>. Write a program to store the port and IP address in host byte order to network byte order from the command-line argument. Display the values of the structure variable, port and IP address, from network byte order to host byte order onto the monitor

Code-

```
int main()
{

in_addr_t ip;
in_port_t port;

struct sockaddr_in sa;
printf("Enter port:");
scanf("%hu",&port);
printf("Enter IP address");
scanf("%u",&ip);

sa.sin_port=htons(port);
sa.sin_addr.s_addr=htonl(ip);
printf("Port given=%hu\n",ntohs(sa.sin_port));
printf("IP given=%u\n",ntohl(sa.sin_addr.s_addr));
return 0;
}
```

Output snippet-



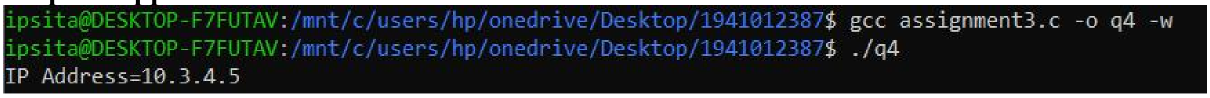
```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c -o q3 -w
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q3
Enter port:2095
Enter IP address172.17.145.122
Port given=2095
IP given=172
```

4. Assume that a line of code in your program is to read the IPV4 address in dotted decimal number and is stored in the network byte order as sa.sin_addr.s_addr=inet addr(''10.3.4.5''),where sa is the structure variable of the type struct sockaddr_in. You are required to rewrite the program to replace inet addr(...) with inet aton(...) to get the IP address and display that IP address using inet ntoa(..).

Code-

```
int main()
{
char *ip;

struct sockaddr_in sa;

inet_aton("10.3.4.5", &sa.sin_addr);
ip = inet_ntoa(sa.sin_addr);
printf("IP Address=%s\n",ip);

return 0;
}
```

Output snippet-



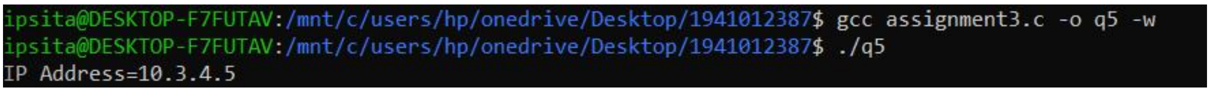
```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c -o q4 -w
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q4
IP Address=10.3.4.5
```

5. Assume that a line of code in your program is to read the IPV4 address in dotted decimal number and is stored in the network byte order as ca.sin_addr.s_addr=inet addr(''10.3.4.5''), where ca is the structure variable of the type struct sockaddr_in. You are required to rewrite the program to replace inet addr(...) with inet pton(...) to get the IP address and display that IP address using inet ntop(..).

Code-

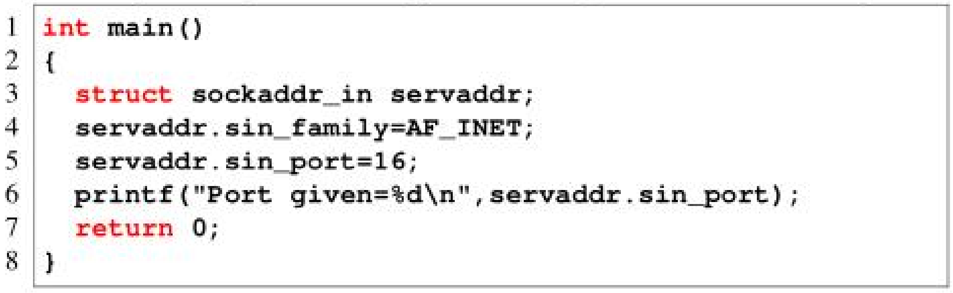
```
int main()
{char ip[16];

struct sockaddr_in ca;

inet_pton(AF_INET, "10.3.4.5", &ca.sin_addr);
inet_ntop(AF_INET, &ca.sin_addr, ip ,16);

printf("IP Address=%s\n",ip);

return 0;
}
```

Output snippet-



```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c -o q5 -w
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q5
IP Address=10.3.4.5
```

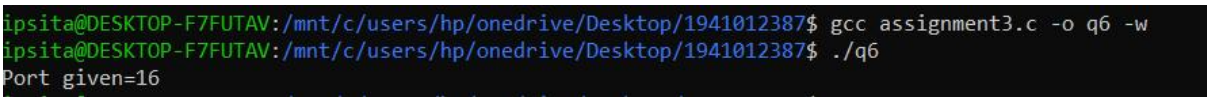
6. Find out the output of the given code snippet. Also state your answer for such output.



```
int main()
{
  struct sockaddr_in servaddr;
  servaddr.sin_family=AF_INET;
  servaddr.sin_port=16;
  printf("Port given=%d\n",servaddr.sin_port);
  return 0;
}
```

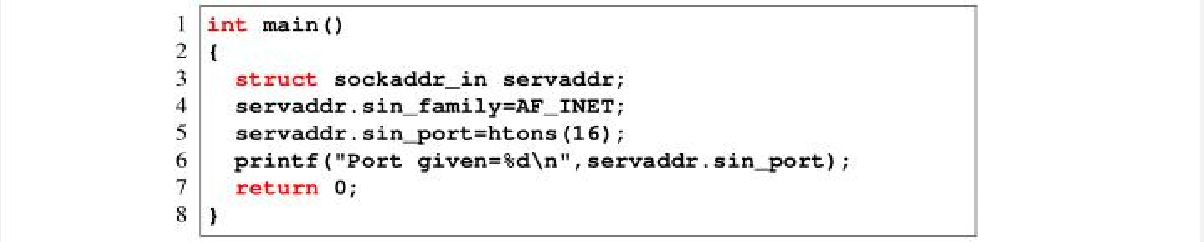
Output snippet-

Here, simply the servaddr.sin_port was assigned the value 16, hence the output displays 16 as port given.



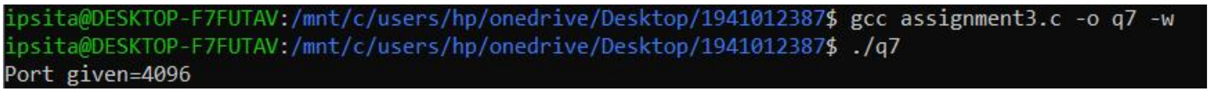
```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c -o q6 -w
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q6
Port given=16
```

7. Find out the output of the given code snippet and justify the reason of getting such output (Hint: look into Host byte order and Network byte order).



```
int main()
{
  struct sockaddr_in servaddr;
  servaddr.sin_family=AF_INET;
  servaddr.sin_port=htons(16);
  printf("Port given=%d\n",servaddr.sin_port);
  return 0;
}
```

Output snippet-



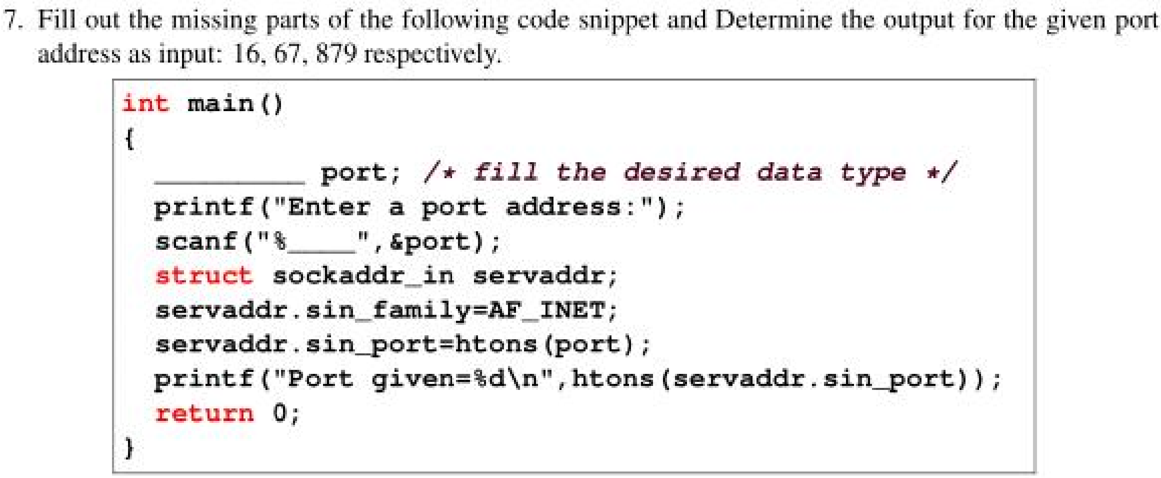
```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c -o q7 -w
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q7
Port given=4096
```

Explanation-

Here, servaddr.sin_port was assigned the value 16 over htons function which swaps its byte ordering and converts it into Big-endian notation (i.e., converts it to Network byte ordering from Host byte ordering).

16=00000000 00010000

After Swap=00010000 00000000=4096



7. Fill out the missing parts of the following code snippet and Determine the output for the given port address as input: 16, 67, 879 respectively.

```
int main()
{
  _________ port; /* fill the desired data type */
  printf("Enter a port address:");
  scanf("%____",&port);
  struct sockaddr_in servaddr;
  servaddr.sin_family=AF_INET;
  servaddr.sin_port=htons(port);
  printf("Port given=%d\n",htons(servaddr.sin_port));
  return 0;
}
```

Filled Code-

int main()

{

in_port_t port; // or unsigned short int port;

printf("Enter a port address:");

scanf(**"%hu"**,&port);

struct sockaddr_in servaddr;

servaddr.sin_family=AF_INET;

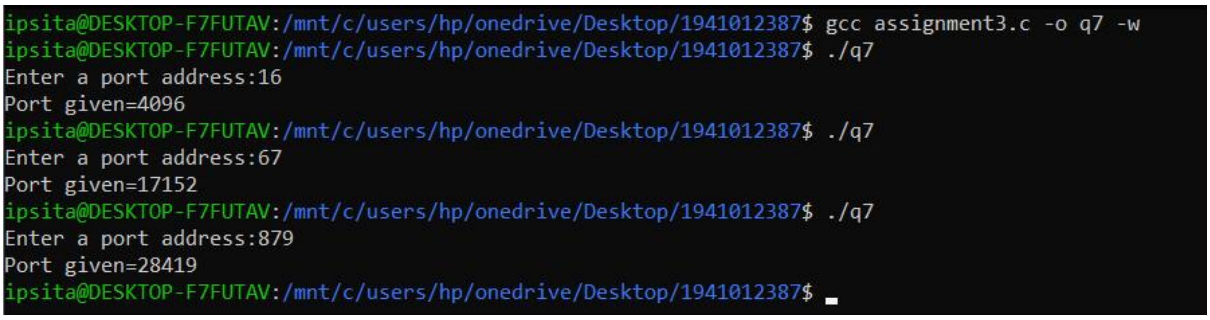
servaddr.sin_port=port;

printf("Port given=%d\n",htons(servaddr.sin_port));

return 0;

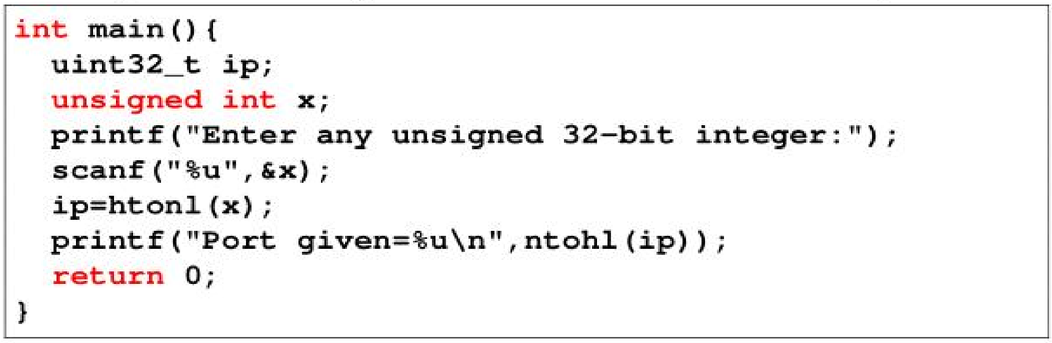
}

Output Snippet-



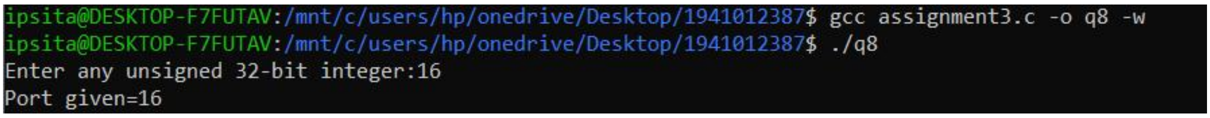
```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c -o q7 -w
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q7
Enter a port address:16
Port given=4096
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q7
Enter a port address:67
Port given=17152
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q7
Enter a port address:879
Port given=28419
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$
```

8. Find out the output of the code snippet and also state the reason for such output.



```
int main(){
  uint32_t ip;
  unsigned int x;
  printf("Enter any unsigned 32-bit integer:");
  scanf("%u",&x);
  ip=htonl(x);
  printf("Port given=%u\n",ntohl(ip));
  return 0;
}
```

Output snippet-



```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c -o q8 -w
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q8
Enter any unsigned 32-bit integer:16
Port given=16
```

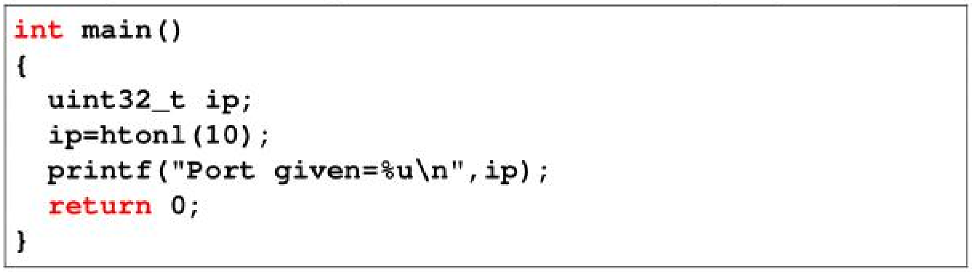
Explanation-
Here, a 32-bit long integer is taken as input from command line and is passed over a htonl(host to network) function which converts it into network byte order i.e. Big endian. Further it is passed on with a ntohl(network to host) function which again does byte swapping and converts it back to host byte order i.e. Little endian.

16=00000000 00010000

After 1st Swap=00010000 00000000=4096 (htonl)

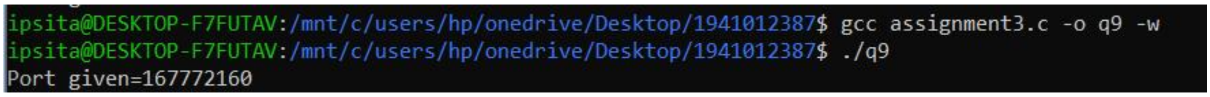
After 2nd Swap=00000000 00010000=16 (ntohl)

9. Find out the output of the code snippet and also trace the reason of getting such output.



```
int main()
{
  uint32_t ip;
  ip=htonl(10);
  printf("Port given=%u\n",ip);
  return 0;
}
```

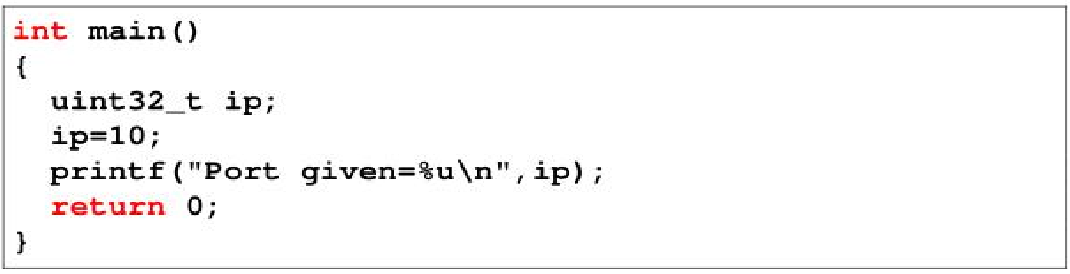
Output snippet-



```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c -o q9 -w
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q9
Port given=167772160
```

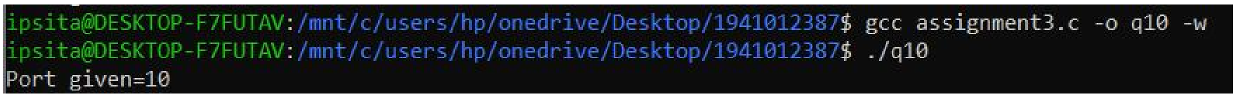
Explanation-
Here, ip is assigned a value htonl(10) which converts 10 to Big endian notation and displays it.
10=00000000 00000000 00000000 00001010
After Swap=00001010 00000000 00000000 00000000=167772160

10. Find out the output of the code snippet and also trace the reason of getting such output.



```
int main()
{
  uint32_t ip;
  ip=10;
  printf("Port given=%u\n",ip);
  return 0;
}
```

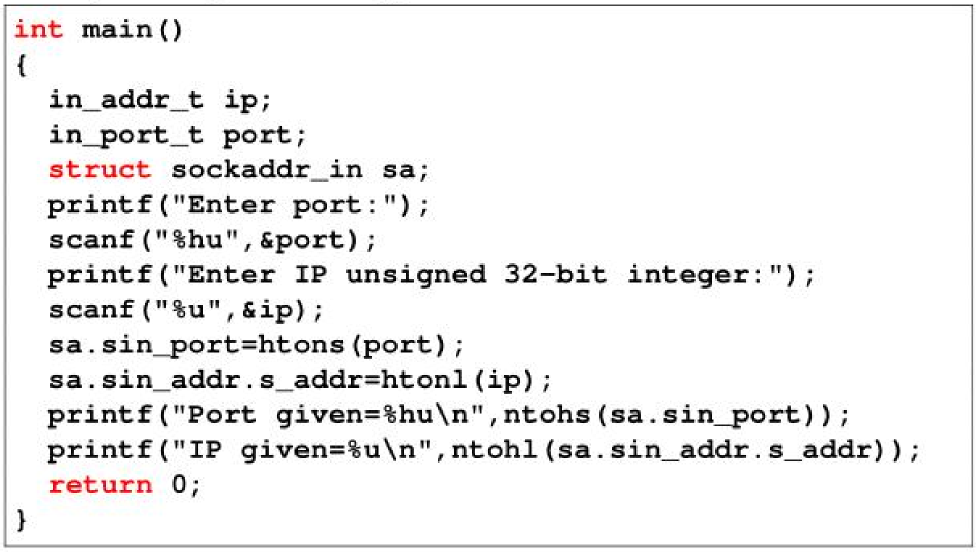
Output snippet-



```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c -o q10 -w
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q10
Port given=10
```

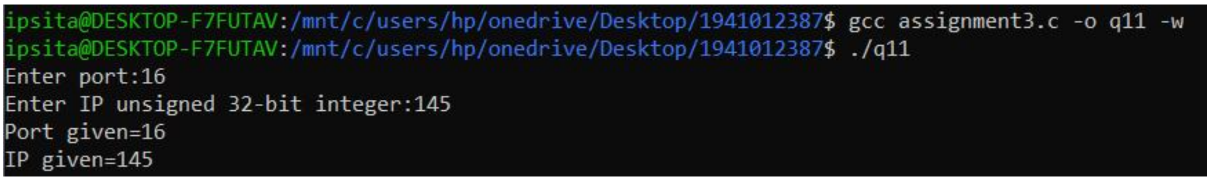
Here, “ip” which is declared as a unsigned 32 bit variable was assigned the value 10, hence the output displays 10 as port given.

11. Justify the output of the given code snippet:



```
int main()
{
  in_addr_t ip;
  in_port_t port;
  struct sockaddr_in sa;
  printf("Enter port:");
  scanf("%hu",&port);
  printf("Enter IP unsigned 32-bit integer:");
  scanf("%u",&ip);
  sa.sin_port=htons(port);
  sa.sin_addr.s_addr=htonl(ip);
  printf("Port given=%hu\n",ntohs(sa.sin_port));
  printf("IP given=%u\n",ntohl(sa.sin_addr.s_addr));
  return 0;
}
```

Output snippet-



```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c -o q11 -w
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q11
Enter port:16
Enter IP unsigned 32-bit integer:145
Port given=16
IP given=145
```

Explanation-Here, one 32-bit long integer and a 16 bit unsigned short integer are taken as input from command line and are passed over a htonl(host to network long) and htons(host to network short) function respectively which converts it into network byte order i.e. Big endian. Further they are passed on with a ntohl(network to host long) and ntohs(network to host short) function respectively which again does byte swapping and converts it back to the host byte order i.e. Little endian.

Id=145=00000000 00000000 00000000 10010001
After 1st Swap=10010001 00000000 00000000 00000000=2432696320 (htons)
After 2nd Swap=00000000 00000000 00000000 10010001=145 (ntohs)

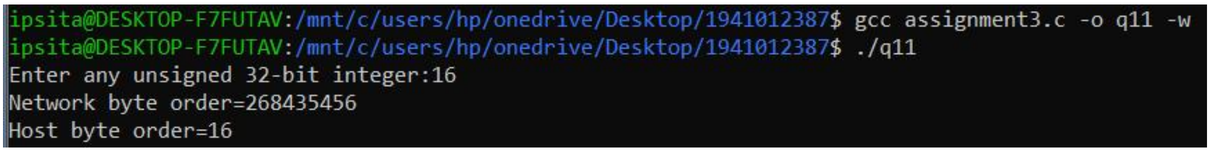
Port=16=00000000 00010000
After 1st Swap=00010000 00000000=4096 (htons)
After 2nd Swap=00000000 00010000=16 (ntohs)

11. Write a program to read an unsigned 32-bit integer x (i.e datatype uint32 t). Assign x to y in network byte order. Display the value of y in network byte order as well as in host byte order.

Code-

```
int main()

{

uint32_t y;

unsigned int x;

printf("Enter any unsigned 32-bit integer:");

scanf("%u",&x);

y=htonl(x);

printf("Network byte order=%u\n",y);

printf("Host byte order=%u\n",ntohl(y));

return 0;

}
```

Output snippet-



```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c -o q11 -w
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q11
Enter any unsigned 32-bit integer:16
Network byte order=268435456
Host byte order=16
```

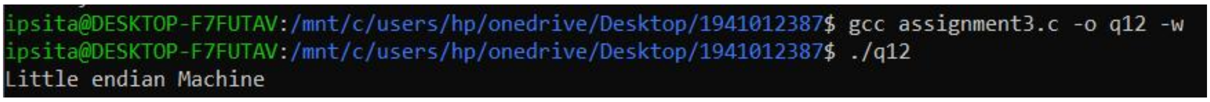
12. Develop a program to determine whether your working machine is in little-endian or in big-endian byte order.

Code-

```
#include <stdio.h>
int main()
{
        int x = 1;
        char *cp = &x;
        if(*cp == 1)
```

```
                printf("Little endian Machine\n");
        else
                printf("Big endian Machine\n");
        return 0;
}
```

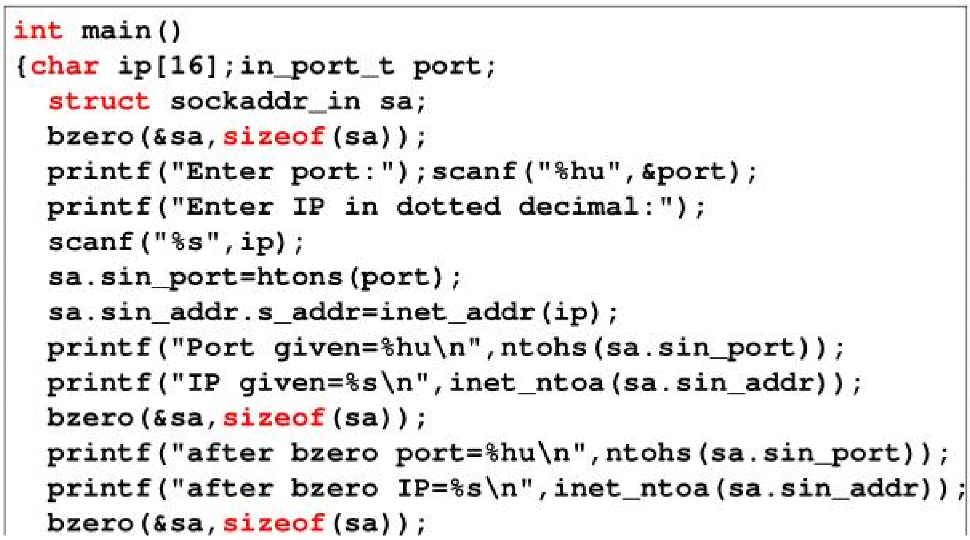
Output snippet-



```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c -o q12 -w
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q12
Little endian Machine
```

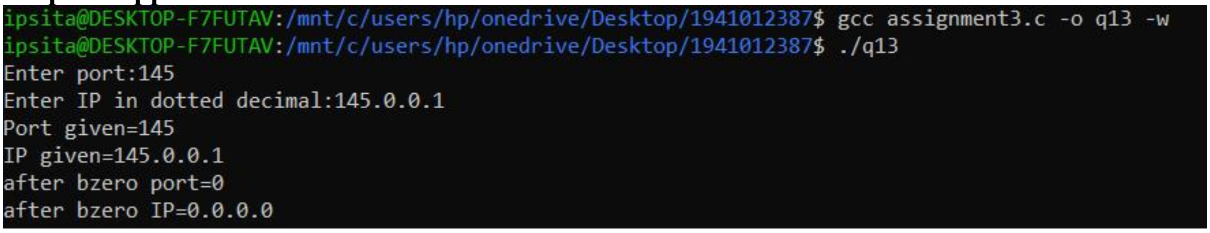
Since, I use a x86 intel processor my system is Little endian.

13. Find and Justify the output of the given code snippet:



```
int main()
{char ip[16];in_port_t port;
  struct sockaddr_in sa;
  bzero(&sa,sizeof(sa));
  printf("Enter port:");scanf("%hu",&port);
  printf("Enter IP in dotted decimal:");
  scanf("%s",ip);
  sa.sin_port=htons(port);
  sa.sin_addr.s_addr=inet_addr(ip);
  printf("Port given=%hu\n",ntohs(sa.sin_port));
  printf("IP given=%s\n",inet_ntoa(sa.sin_addr));
  bzero(&sa,sizeof(sa));
  printf("after bzero port=%hu\n",ntohs(sa.sin_port));
  printf("after bzero IP=%s\n",inet_ntoa(sa.sin_addr));
  bzero(&sa,sizeof(sa));
```

Output snippet-



```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c -o q13 -w
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q13
Enter port:145
Enter IP in dotted decimal:145.0.0.1
Port given=145
IP given=145.0.0.1
after bzero port=0
after bzero IP=0.0.0.0
```

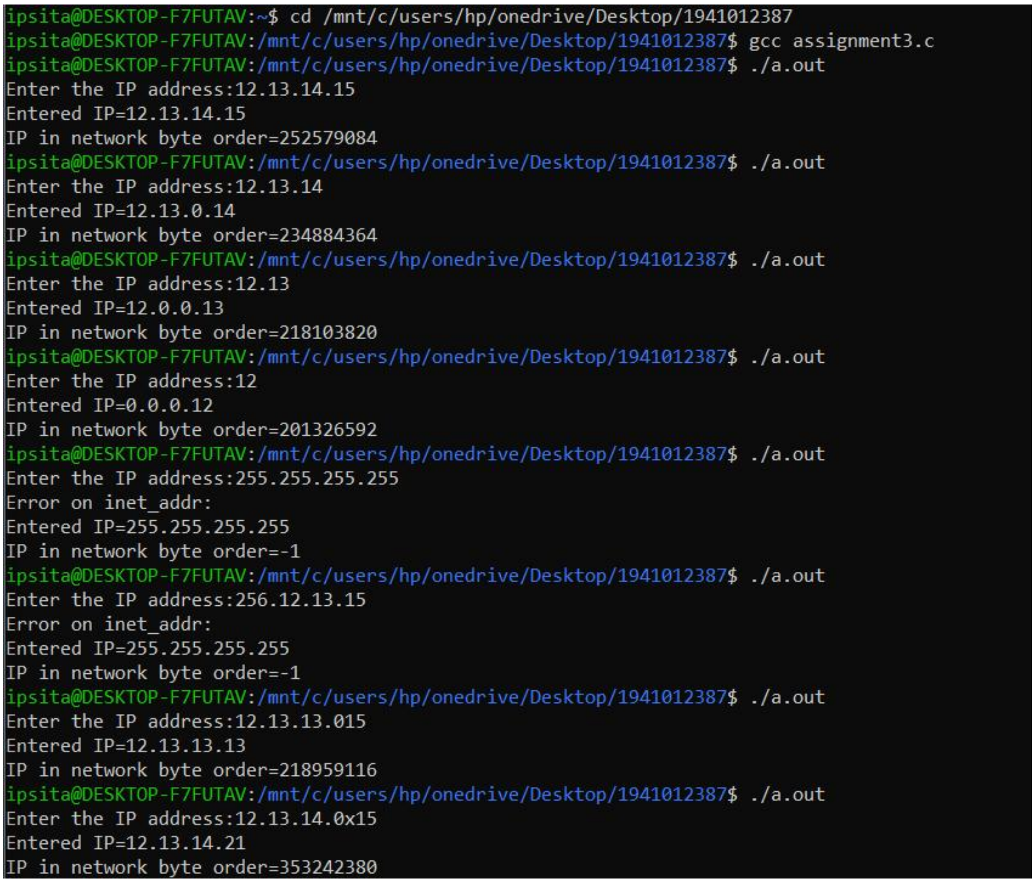
Explanation- Since, we know that bzero sets the specified number of bytes to 0 in the destination and here it is done for the size of the whole structure variable for which all the bytes are set to 0.

14. Consider the following code snippet to work with IP address conversion function **inet_addr()**:

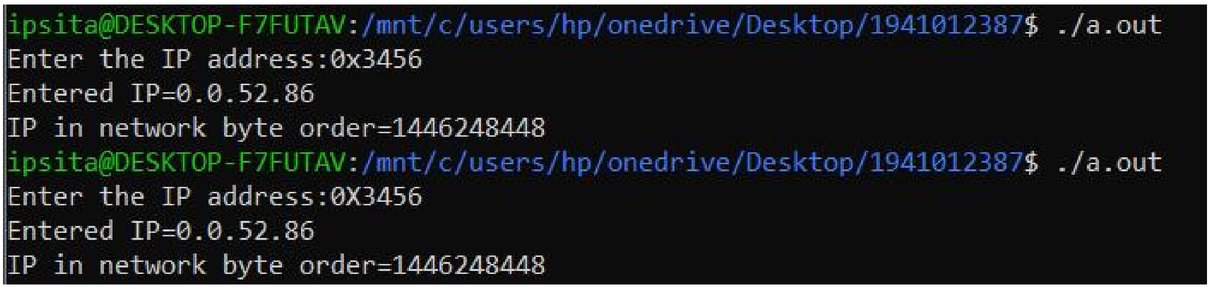
```
int main(){
  struct sockaddr_in serv;
  char ipaddr[16];
  printf("Enter the IP address:");
  scanf("%s",ipaddr);
  serv.sin_addr.s_addr=inet_addr(ipaddr);
  if(serv.sin_addr.s_addr==-1){
    printf("Error on inet_addr:\n");
  }
  printf("Entered IP=%s\n",inet_ntoa(serv.sin_addr));
  printf("IP in network byte order=%d\n",serv.sin_addr.s_addr);
  return 0;
}
```

Run the test cases and justify the program generated output

(a) Enter the IP address: 12.13.14.15
(b) Enter the IP address: 12.13.14
(c) Enter the IP address: 12.13
(d) Enter the IP address: 12
(e) Enter the IP address: 255.255.255.255
(f) Enter the IP address: 256.12.13.15
(g) Enter the IP address: 12.13.14.015
(h) Enter the IP address: 12.13.14.0x15
(i) Enter the IP address: 0x3456
(j) Enter the IP address: 0X3456



```
ipsita@DESKTOP-F7FUTAV:~$ cd /mnt/c/users/hp/onedrive/Desktop/1941012387
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./a.out
Enter the IP address:12.13.14.15
Entered IP=12.13.14.15
IP in network byte order=252579084
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./a.out
Enter the IP address:12.13.14
Entered IP=12.13.0.14
IP in network byte order=234884364
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./a.out
Enter the IP address:12.13
Entered IP=12.0.0.13
IP in network byte order=218103820
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./a.out
Enter the IP address:12
Entered IP=0.0.0.12
IP in network byte order=201326592
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./a.out
Enter the IP address:255.255.255.255
Error on inet_addr:
Entered IP=255.255.255.255
IP in network byte order=-1
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./a.out
Enter the IP address:256.12.13.15
Error on inet_addr:
Entered IP=255.255.255.255
IP in network byte order=-1
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./a.out
Enter the IP address:12.13.13.015
Entered IP=12.13.13.13
IP in network byte order=218959116
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./a.out
Enter the IP address:12.13.14.0x15
Entered IP=12.13.14.21
IP in network byte order=353242380
```



```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./a.out
Enter the IP address:0x3456
Entered IP=0.0.52.86
IP in network byte order=1446248448
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./a.out
Enter the IP address:0X3456
Entered IP=0.0.52.86
IP in network byte order=1446248448
```

Explanation- Here, the input string is passed over inet_addr to check if it’s a valid ip address, if not valid i.e., exceeds or equals 255.255.255.255 then it throws -1 else the dotted decimal notation.

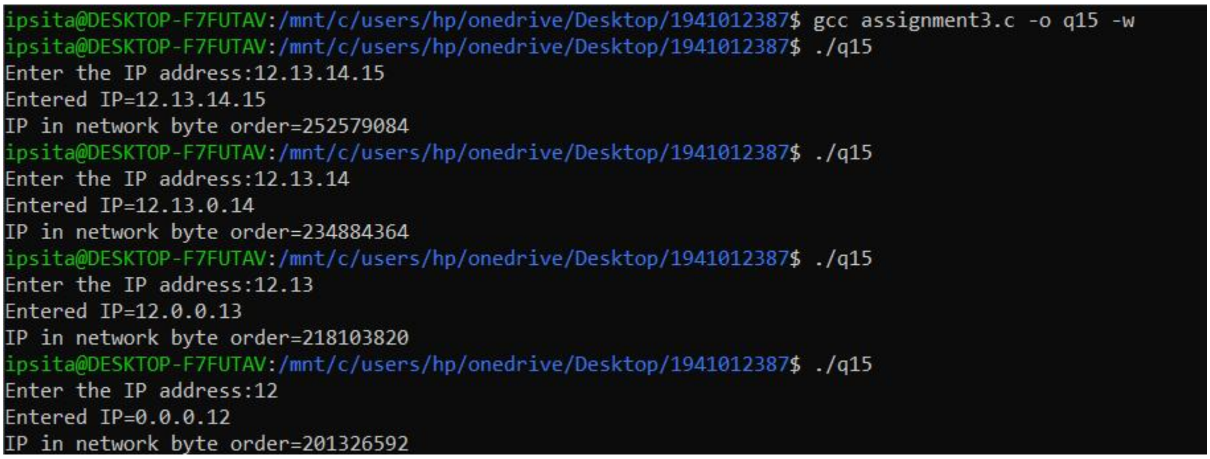
That is why (e) and (f) throw an error. In (i) and (j) the inputs are given as hexadecimal values and output show their respective dotted decimal notation.

15. Rewrite the question-14 code to replace inet addr() function with inet aton() and verify the above test cases in your modified program.

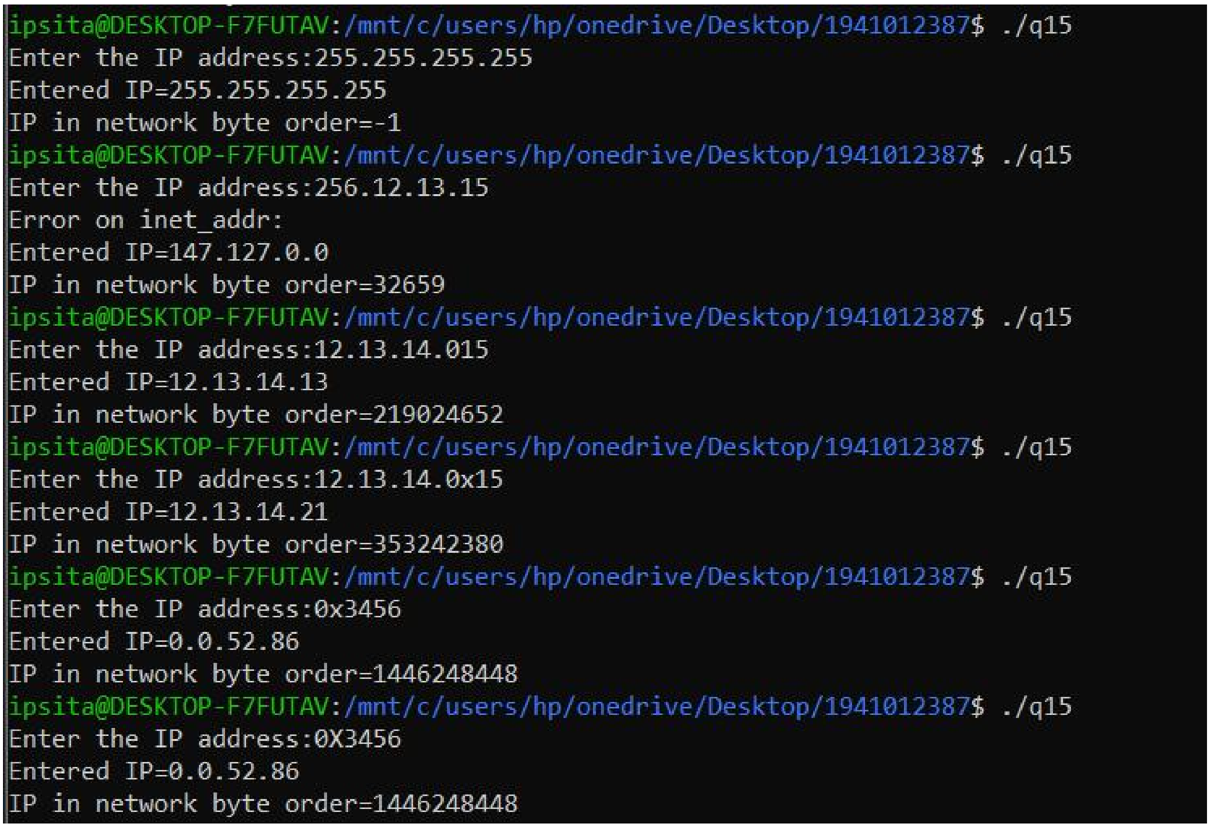
Code-

```
int main(){
   struct sockaddr_in serv;
   char ipaddr[16];
   printf("Enter the IP address:");
   scanf("%s",ipaddr);
   inet_aton(ipaddr,&serv.sin_addr.s_addr);
   if(inet_aton(ipaddr,&serv.sin_addr.s_addr)==0)
{
      printf("Error on inet_addr:\n");

}
printf("Entered IP=%s\n",inet_ntoa(serv.sin_addr));
printf("IP in network byte order=%d\n",serv.sin_addr.s_addr);
return 0;
}
```

Output snippet-



```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c -o q15 -w
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q15
Enter the IP address:12.13.14.15
Entered IP=12.13.14.15
IP in network byte order=252579084
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q15
Enter the IP address:12.13.14
Entered IP=12.13.0.14
IP in network byte order=234884364
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q15
Enter the IP address:12.13
Entered IP=12.0.0.13
IP in network byte order=218103820
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q15
Enter the IP address:12
Entered IP=0.0.0.12
IP in network byte order=201326592
```

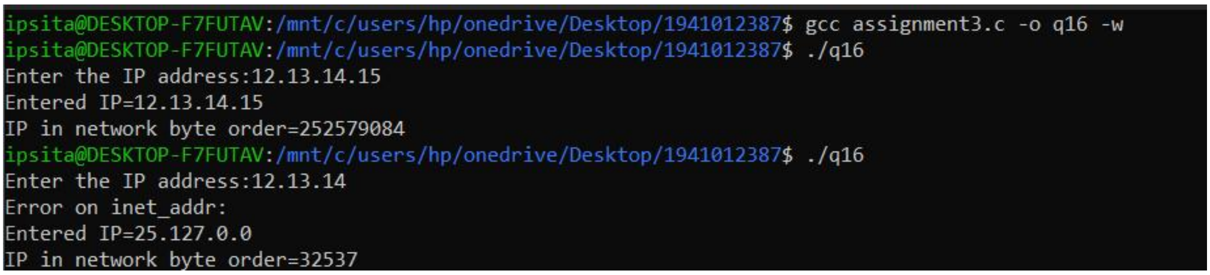


```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q15
Enter the IP address:255.255.255.255
Entered IP=255.255.255.255
IP in network byte order=-1
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q15
Enter the IP address:256.12.13.15
Error on inet_addr:
Entered IP=147.127.0.0
IP in network byte order=32659
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q15
Enter the IP address:12.13.14.015
Entered IP=12.13.14.13
IP in network byte order=219024652
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q15
Enter the IP address:12.13.14.0x15
Entered IP=12.13.14.21
IP in network byte order=353242380
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q15
Enter the IP address:0x3456
Entered IP=0.0.52.86
IP in network byte order=1446248448
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q15
Enter the IP address:0X3456
Entered IP=0.0.52.86
IP in network byte order=1446248448
```

16. Rewrite the question-14 code to replace inet addr() function with inet pton() and verify the above test cases in your modified program.

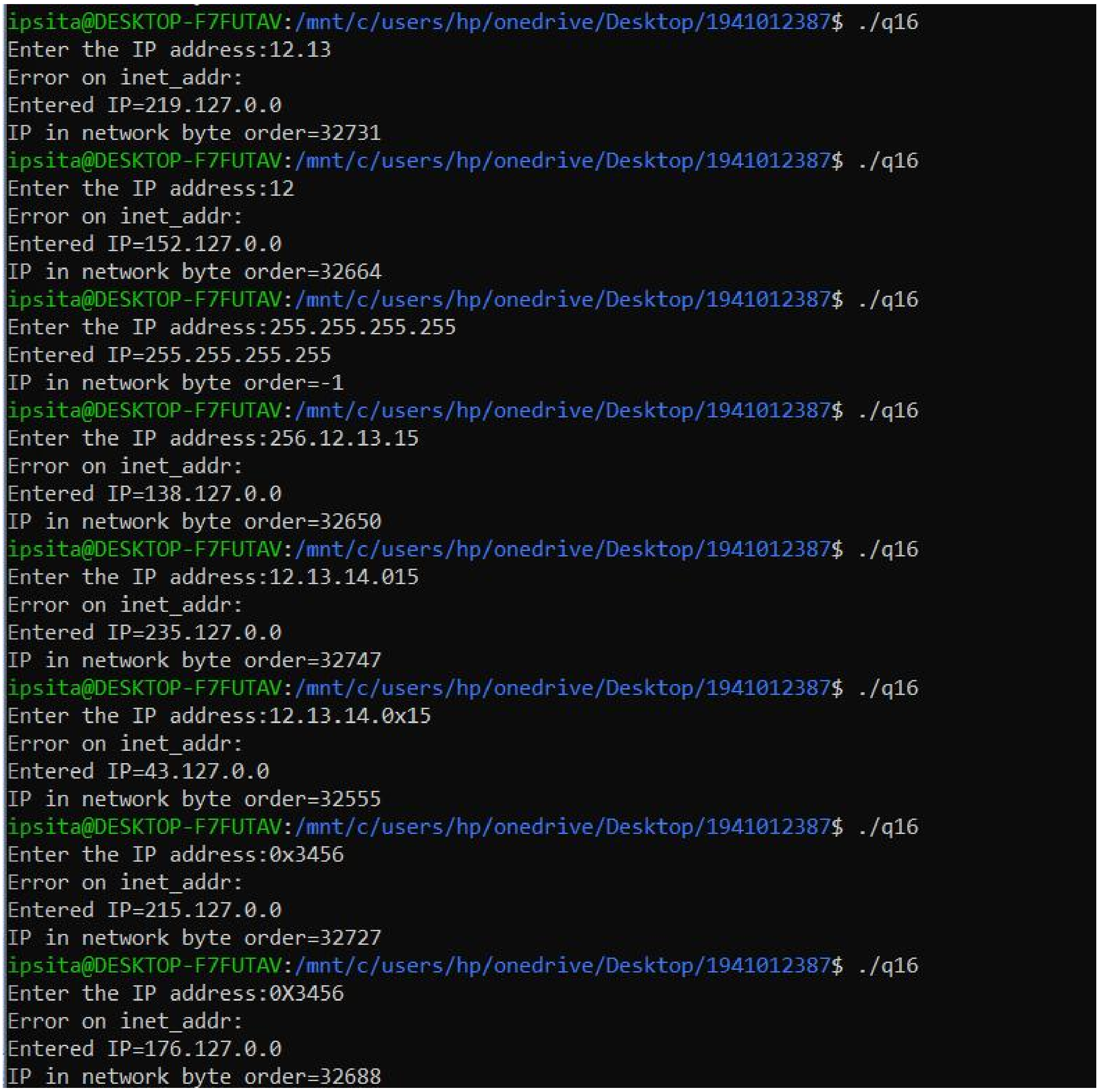
Code-

```
int main(){
   struct sockaddr_in serv;
   char ipaddr[16];
   printf("Enter the IP address:");
   scanf("%s",ipaddr);
   inet_pton(AF_INET,ipaddr,&serv.sin_addr.s_addr);
   if(inet_pton(AF_INET,ipaddr,&serv.sin_addr.s_addr)==0){
      printf("Error on inet_addr:\n");
   }
printf("Entered IP=%s\n",inet_ntoa(serv.sin_addr));
printf("IP in network byte order=%d\n",serv.sin_addr.s_addr);
return 0;
}
```



```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c -o q16 -w
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q16
Enter the IP address:12.13.14.15
Entered IP=12.13.14.15
IP in network byte order=252579084
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q16
Enter the IP address:12.13.14
Error on inet_addr:
Entered IP=25.127.0.0
IP in network byte order=32537
```

Output snippet-



```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q16
Enter the IP address:12.13
Error on inet_addr:
Entered IP=219.127.0.0
IP in network byte order=32731
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q16
Enter the IP address:12
Error on inet_addr:
Entered IP=152.127.0.0
IP in network byte order=32664
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q16
Enter the IP address:255.255.255.255
Entered IP=255.255.255.255
IP in network byte order=-1
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q16
Enter the IP address:256.12.13.15
Error on inet_addr:
Entered IP=138.127.0.0
IP in network byte order=32650
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q16
Enter the IP address:12.13.14.015
Error on inet_addr:
Entered IP=235.127.0.0
IP in network byte order=32747
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q16
Enter the IP address:12.13.14.0x15
Error on inet_addr:
Entered IP=43.127.0.0
IP in network byte order=32555
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q16
Enter the IP address:0x3456
Error on inet_addr:
Entered IP=215.127.0.0
IP in network byte order=32727
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q16
Enter the IP address:0X3456
Error on inet_addr:
Entered IP=176.127.0.0
IP in network byte order=32688
```

17. Rewrite the question-14 code to replace inet addr() function with inet pton() and verify the above test cases in your modified program.

--Same as 16---

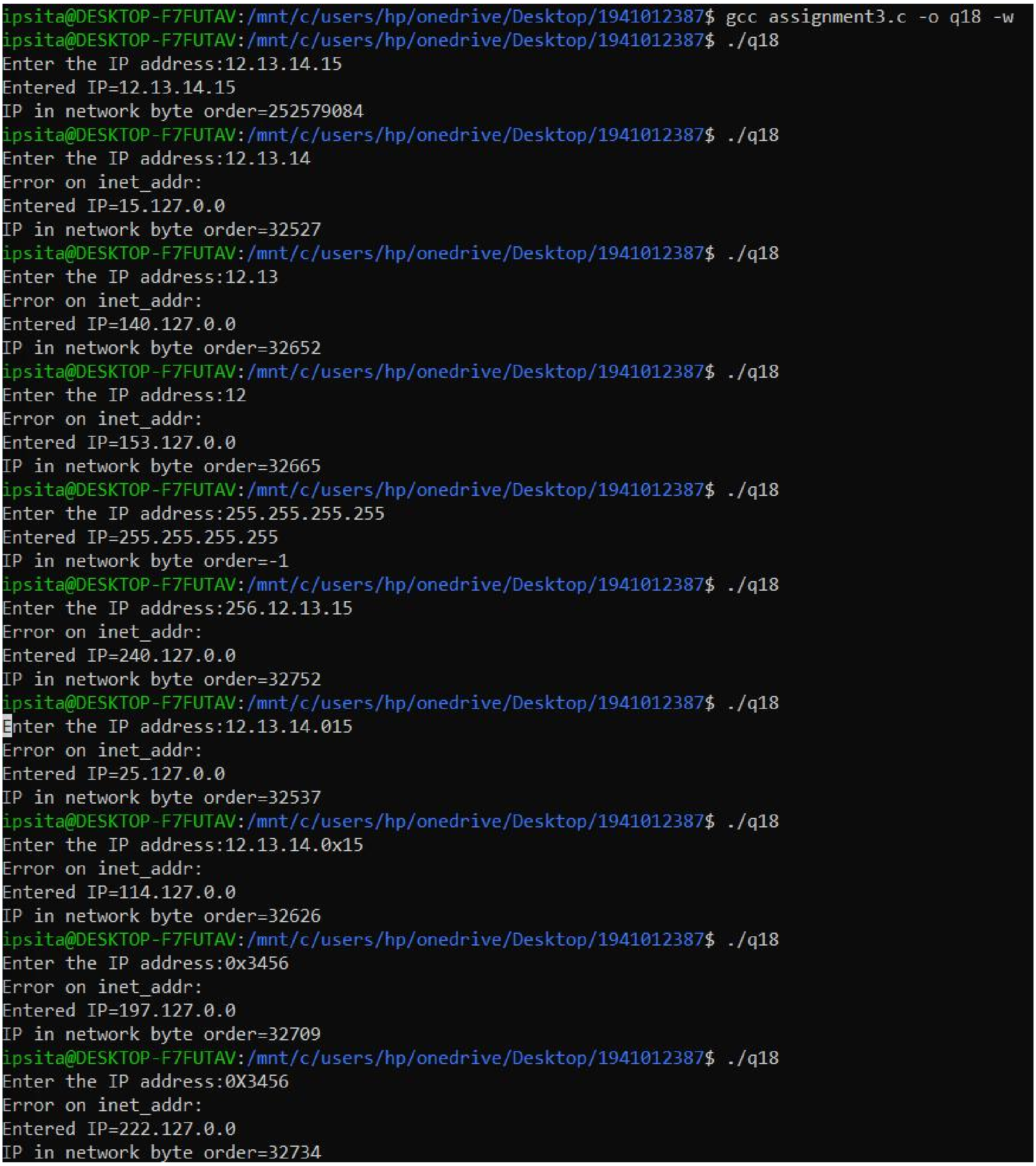
18. Rewrite the question-14 code to replace inet addr() function with inet pton() and inet ntoa() to inet ntop(). Also, verify the above test cases in your modified program.

Code-

```
int main(){
  struct sockaddr_in serv;

  char ipaddr[16];

  printf("Enter the IP address:");
  scanf("%s",ipaddr);
```

```
 inet_pton(AF_INET,ipaddr,&serv.sin_addr.s_addr);
   if(inet_pton(AF_INET,ipaddr,&(serv.sin_addr.s_addr))==0){
      printf("Error on inet_addr:\n");
   }
printf("Entered IP=%s\n",inet_ntop(AF_INET,&(serv.sin_addr),ipaddr,16));
printf("IP in network byte order=%d\n",serv.sin_addr.s_addr);
return 0;
}
```

Output snippet-



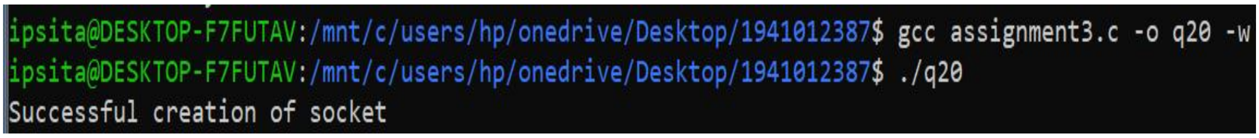
```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c -o q18 -w
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q18
Enter the IP address:12.13.14.15
Entered IP=12.13.14.15
IP in network byte order=252579084
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q18
Enter the IP address:12.13.14
Error on inet_addr:
Entered IP=15.127.0.0
IP in network byte order=32527
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q18
Enter the IP address:12.13
Error on inet_addr:
Entered IP=140.127.0.0
IP in network byte order=32652
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q18
Enter the IP address:12
Error on inet_addr:
Entered IP=153.127.0.0
IP in network byte order=32665
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q18
Enter the IP address:255.255.255.255
Entered IP=255.255.255.255
IP in network byte order=-1
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q18
Enter the IP address:256.12.13.15
Error on inet_addr:
Entered IP=240.127.0.0
IP in network byte order=32752
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q18
Enter the IP address:12.13.14.015
Error on inet_addr:
Entered IP=25.127.0.0
IP in network byte order=32537
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q18
Enter the IP address:12.13.14.0x15
Error on inet_addr:
Entered IP=114.127.0.0
IP in network byte order=32626
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q18
Enter the IP address:0x3456
Error on inet_addr:
Entered IP=197.127.0.0
IP in network byte order=32709
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q18
Enter the IP address:0X3456
Error on inet_addr:
Entered IP=222.127.0.0
IP in network byte order=32734
```

20. Write a program to create a socket (i.e. end-point of a connection) and display the whether end-point is successfully created or not.

Code-

```
int main(int argc , char *argv[])
{
        int socket_fd;
        struct sockaddr_in server;

        //Socket (i.e. end-point of a connection )created successfully
        socket_fd = socket(AF_INET , SOCK_STREAM , IPPROTO_TCP);//The protocol type
mentioned supports the family

        if (socket_fd < 0)
        {
                printf("Could not create socket\n");
        }
        else {
        printf("Successful creation of socket\n");
        }

}
```

Output snippet-



```
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ gcc assignment3.c -o q20 -w
ipsita@DESKTOP-F7FUTAV:/mnt/c/users/hp/onedrive/Desktop/1941012387$ ./q20
Successful creation of socket
```