

**Department of Computer Science and Information Technology**

**Lab Manual of Computer Networking**

**Subject Code: CSIT-501**

**B. Tech: III Year 5th Sem**

**Name of the Student :   Pratik Patidar**

**Branch & Section       :   B.Tech (CSIT)**

**Roll No.                         :   0827CI191046**

**Program 1:** Write a program to generate VRC for a given Character.

Input: Character (Any Letter, digit or symbol) (ASCII value to binary)

(Generate binary (8 bits) according to the ASCII value)

Output: VRC with even/odd parity in 9 bits

A= 65 🡪 binary 🡪 count ‘1’ bits 🡪 if even, parity bit = 0, else parity bit = 1

#include <iostream>

using namespace std;

void decToBinary(int n)

{

int binaryNum[32];

int i = 0,count=0,bit=0;

while (n > 0) {

binaryNum[i] = n % 2;

n = n / 2;

i++;

}

for (int j = i - 1; j >= 0; j--)

{

if(binaryNum[j]==1)

{

count++;

}

}

cout<<"count : "<<count<<endl;

if(count%2 !=0)

{

bit=1;

}

cout<<"bits : ";

for (int j = i - 1; j >= 0; j--)

{

cout << binaryNum[j];

}

cout<<bit;

}

int main()

{

int n = 18;

decToBinary(n);

return 0;

}

**Program 2:** Write a program to generate LRC for a given Character.

Input: Character (Any digit or symbol)

(Generate binary according to the ASCII value)

Output: LRC with even/odd parity in 8 bits

C: 67 : 10100110 //8 bits

E: 69: 10110010 //8 bits

LRC: 00010100

#include <iostream>

using namespace std;

void decToBinary(int n, int binaryNum[])

{

int i = 7, count = 0, bit = 0;

while (n > 0)

{

binaryNum[i] = n % 2;

n = n / 2;

i--;

}

for (int j = 0; j <= 7; j++)

{

if (binaryNum[j] == 1)

{

count++;

}

}

cout << endl

<< "count : " << count << endl;

if (count % 2 != 0)

{

bit = 1;

}

cout << "bits : ";

for (int j = 0; j <= 7; j++)

{

cout << binaryNum[j];

}

// cout<<bit;

}

int main()

{

char alfa1, alfa2;

int binaryNum1[8] = {0}, binaryNum2[8] = {0}, lrc[8] = {0};

cin >> alfa1 >> alfa2;

int asci1 = alfa1, asci2 = alfa2, i;

cout << asci1 << " " << asci2 << endl;

decToBinary(asci1, binaryNum1);

decToBinary(asci2, binaryNum2);

for (i = 0; i < 8; i++)

{

if (binaryNum1[i] == binaryNum2[i])

{

lrc[i] = 0;

}

else

{

lrc[i] = 1;

}

}

cout << endl

<< "LRC = ";

for (int j = 0; j < 8; j++)

{

cout << lrc[j];

}

return 0;

}

**Program 3:** Write a program to bit stuff the given binary data.

*Algorithm for Bit−Stuffing*

1. Start

2. Initialize the array for transmitted stream with the special bit pattern 0111 1110 which indicates the beginning of the frame.

3. Get the bit stream to be transmitted in to the array.

4. Check for five consecutive ones and if they occur, stuff a bit 0

5. Display the data transmitted as it appears on the data line after appending 0111 1110 at the End.

6. For de−stuffing, copy the transmitted data to another array after detecting the stuffed bits.

7. Display the received bit stream.

8. Stop

**INPUT/OUTPUT:**

**Enter the bit string (space for each byte) 11111111 01111110 00111110**

**After stuffing: 01111110 111110111011111010001111100 01111110**

#include <iostream>

using namespace std;

int main()

{

int n, count = 0;

cout << "enter no of bytes in data : ";

cin >> n;

int ar[n], a[8] = {0, 1, 1, 1, 1, 1, 1, 0};

cout << "enter bytes of data : ";

for (int i = 0; i < n; i++)

{

cin >> ar[i];

}

for (int i = 0; i < 8; i++)

{

cout << a[i];

}

for (int i = 0; i < n; i++)

{

cout << ar[i];

if (ar[i] == 1)

{

count++;

}

else

{

count = 0;

}

if (count == 5)

{

cout << "0";

}

}

for (int i = 0; i < 8; i++)

{

cout << a[i];

}

return 0;

}

**Program 4:** Write a program to convert Dotted Decimal Notation IP to Binary Notation IP

Sample input by user:

Enter the IP in Dotted Decimal Notation:

Enter first octet: 126

Enter second octet: 7

Enter third octet: 89

Enter fourth octet: 76

You entered in Dotted Decimal Notation: 126.7.89.76

Sample output:

IP in Binary Decimal Notation:

01111110 00000111 01011001 01001100

#include<iostream>

#include<stdlib.h>

using namespace std;

void decToBin( int binArr[][8] , int row )

{

int num = binArr[row][0];

int index = 7 ;

while (index >= 0 )

{

binArr[row][index] = num % 2 ;

num = num / 2 ;

index-- ;

}

}

int main()

{

int binary[4][8] ;

cout<<"Enter the IP in Dotted Decimal Notation:\nEnter first octet:";

cin>>binary[0][0];

cout<<"Enter second octet:";

cin>>binary[1][0];

cout<<"Enter third octet:";

cin>>binary[2][0];

cout<<"Enter fourth octet:";

cin>>binary[3][0];

for( int row = 0 ; row<4 ; row++ )

{

decToBin( binary , row ) ;

}

cout<<"IP in Binary Decimal Notation: ";

for( int row = 0 ; row < 4 ; row++ )

{

for( int column = 0 ; column < 8 ; column++ )

cout<<binary[row][column];

cout<<" ";

}

}

**Program 5:** Write a program to display the class of the IP Address entered by the user.

Sample Input:

Enter the IP Address in Dotted Decimal Notation: 193.90.74.125

Enter the first octet: 193

Enter the second octet: 90

Enter the third octet: 74

Enter the fourth octet: 125

Sample Output: It is Class B address

#include<iostream>

#include<stdlib.h>

#include<math.h>

using namespace std;

void IP\_Class()

{

int binary[4][8] ;

cout<<"Enter the IP in Dotted Decimal Notation:\nEnter first octet:";

cin>>binary[0][0];

cout<<"Enter second octet:";

cin>>binary[1][0];

cout<<"Enter third octet:";

cin>>binary[2][0];

cout<<"Enter fourth octet:";

cin>>binary[3][0];

if( binary[0][0] >= 1 && binary[0][0] <= 126 )

cout<<"It is Class A address.\n";

else if( binary[0][0] >= 128 && binary[0][0] <= 191 )

cout<<"It is Class B address.\n";

else if( binary[0][0] >= 192 && binary[0][0] <= 223 )

cout<<"It is Class C address.\n";

else if( binary[0][0] >= 224 && binary[0][0] <= 2399 )

cout<<"It is Class D address.\n";

else if( binary[0][0] >= 240 && binary[0][0] <= 255 )

cout<<"It is Class E address.\n";

else

cout<<"Invalid IP Addredd.\n";

}

int main()

{

IP\_Class() ;

}

**Program 6:** Write a program to convert IP address provided by the user in Binary notation to Dotted Decimal Notation entered by the user.

Sample Input:

Enter the IP Address in Dotted Decimal Notation:

Enter the Binary for first octet: 10001001

Enter the Binary for second octet: 01111011

Enter the Binary for third octet: 10000000

Enter the Binary for fourth octet: 11111011

Sample Output: The given IP address is: 137.123.128.251

#include<iostream>

#include<stdlib.h>

#include<math.h>

using namespace std;

int bin\_to\_dec(string add)

{

int num = 0;

int base = 0;

int i = add.length()-1;

while( i >= 0 )

{

if( add[i] == '1')

num = num + pow(2,base);

base++;

i--;

}

return num;

}

void Bin\_to\_DDN()

{

string first , second , third , fourth ;

cout<<"Enter the IP Address in Dotted Decimal Notation\nEnter the Binary for first octet:";

getline( cin>> ws, first);

cout<<"\nEnter the Binary for second octet:";

getline( cin>> ws, second);

cout<<"\nEnter the Binary for third octet:";

getline( cin>> ws, third);

cout<<"\nEnter the Binary for fourth octet:";

getline( cin>> ws, fourth);

if( first.length()== 8 && second.length()== 8 && third.length()== 8 && fourth.length()== 8)

{

int f = bin\_to\_dec(first) ;

int s = bin\_to\_dec(second) ;

int t = bin\_to\_dec(third) ;

int fth = bin\_to\_dec(fourth) ;

cout<<"\nDecimal address is :\n"<<f<<"."<<s<<"."<<t<<"."<<fth<<"\n";

}

else cout<<"Invalid Address\n";

}

int main()

{

Bin\_to\_DDN();

}

**Program 7:** Write a program to find out the Class, Network Address and Broadcast Address of the IP address provided by the user (in classful Addressing).

#include<iostream>

using namespace std;

char IP\_Class(int binary)

{

if( binary >= 1 && binary <= 126 )

return('A') ;//cout<<"It is Class A address.\n";

else if( binary >= 128 && binary <= 191 )

return('B') ;//cout<<"It is Class B address.\n";

else if( binary >= 192 && binary <= 223 )

return('C') ;//cout<<"It is Class C address.\n";

else if( binary >= 224 && binary <= 239 )

return('D') ;//cout<<"It is Class D address.\n";

else if( binary >= 240 && binary <= 255 )

return('E') ;//cout<<"It is Class E address.\n";

else

return('0');

}

int main()

{

int decimal[4][1] ;

cout<<"Enter the IP in Dotted Decimal Notation:\nEnter first octet:";

cin>>decimal[0][0];

cout<<"Enter second octet:";

cin>>decimal[1][0];

cout<<"Enter third octet:";

cin>>decimal[2][0];

cout<<"Enter fourth octet:";

cin>>decimal[3][0];

char ip\_class = IP\_Class(decimal[0][0]) ;

int network\_add[4][1] , broadcast\_add[4][1] ;

if(ip\_class == 'A')

{

network\_add[0][0] = decimal[0][0] ;

network\_add[1][0] = 0 ;

network\_add[2][0] = 0 ;

network\_add[3][0] = 0 ;

broadcast\_add[0][0] = decimal[0][0] ;

broadcast\_add[1][0] = broadcast\_add[2][0] = broadcast\_add[3][0] = 255;

}

else if(ip\_class == 'B')

{

network\_add[0][0] = decimal[0][0] ;

network\_add[1][0] = decimal[1][0] ;

network\_add[2][0] = 0 ;

network\_add[3][0] = 0 ;

broadcast\_add[0][0] = decimal[0][0] ;

broadcast\_add[1][0] = decimal[1][0] ;

broadcast\_add[2][0] = broadcast\_add[3][0] = 255 ;

}

else if(ip\_class == 'C')

{

network\_add[0][0] = decimal[0][0] ;

network\_add[1][0] = decimal[1][0] ;

network\_add[2][0] = decimal[2][0] ;

network\_add[3][0] = 0 ;

broadcast\_add[0][0] = decimal[0][0] ;

broadcast\_add[1][0] = decimal[1][0] ;

broadcast\_add[2][0] = decimal[2][0] ;

broadcast\_add[3][0] = 255 ;

}

cout<<"\nClass of IP Address "<<decimal[0][0]<<"."<<decimal[1][0]<<"."<<decimal[2][0]<<"."<<decimal[3][0]<<": "<<ip\_class;

cout<<"\n Address of IP Address "<<network\_add[0][0]<<"."<<network\_add[1][0]<<"."<<network\_add[2][0]<<"."<<network\_add[3][0];

cout<<"\nBroadcast Address of IP Address "<<broadcast\_add[0][0]<<"."<<broadcast\_add[1][0]<<"."<<broadcast\_add[2][0]<<"."<<broadcast\_add[3][0];

}

**Program 8:** Write a program to suggest the user that which class IP address he/she can adopt in order to perform networking (using Classful Addressing)

Input: Number of Hosts

Output: Suggested IP address Class/ Classes.

#include<iostream>

#include<math.h>

using namespace std;

char sugg\_class(int host)

{

if( host <= pow(2,24)-2 && host > pow(2,16)-2 )

return('A');

else if( host <= pow(2,16)-2 && host > pow(2,8)-2 )

return('B');

else if( host <= pow(2,8)-2 )

return('C');

}

int main()

{

int host;

cout<<"Enter the number of Hosts.\n";

cin>>host;

cout<<"Class of IP address you prefer: "<<sugg\_class(host) ;

return 0;

}

**Program 9:** Write a Program to calculate total number of subnets and total number of hosts given subnet mask (dotted decimal notation) as input by the user for classful addressing.

Input: 255.255.240.0

Output: No. of Subnets: 14

No. of Hosts: 4094

#include<bits/stdc++.h>

#include <cmath>

using namespace std;

string d\_to\_b(int d)

{

int binaryNum[8]={0,0,0,0,0,0,0,0};

int i = 0;

string b;

while (d > 0) {

binaryNum[i] = d % 2;

d = d / 2;

i++;

}

if(i==0)

{

i=8;

for (int j = i - 1; j >= 0; j--)

{

b=b+to\_string(binaryNum[j]);

}

}

else

{

for (int j = i - 1; j >= 0; j--)

{

b=b+to\_string(binaryNum[j]);

}

}

return b;

}

string dot\_dec\_to\_dot\_bin(string sb)

{

int octet;

string s1="",bsb="",bin="";

for(int i=0;i<sb.length();i++)

{

if(sb[i]=='.')

{

bin+=d\_to\_b(stoi(s1));

s1="";

continue;

}

s1=s1+sb[i];

}

return bin;

}

int main()

{

int h\_id=0,N\_id\_s\_id=0, s\_id=0;

string sb;

cout<<"Enter the Subnet mask:\n";

cin>>sb;

sb+=".";

string sb\_binary=dot\_dec\_to\_dot\_bin(sb);

for(int i=0;i<sb\_binary.length();i++)

{

if(sb\_binary[i]=='0')

{

h\_id++;

}

}

N\_id\_s\_id=sb\_binary.length()-h\_id;

cout<<"\nFor class A : \n";

cout<<"Total No. of hosts per each Network: "<<pow(2,h\_id)<<endl;

s\_id=N\_id\_s\_id-8;

cout<<"Total no. of subnets: "<<pow(2,s\_id)<<endl;

cout<<"\nFor class B : \n";

cout<<"Total No. of hosts per each Network: "<<pow(2,h\_id)<<endl;

s\_id=N\_id\_s\_id-16;

cout<<"Total no. of subnets: "<<pow(2,s\_id)<<endl;

cout<<"\nFor class C : \n";

cout<<"Total No. of hosts per each Network: "<<pow(2,h\_id)-2<<endl;

s\_id=N\_id\_s\_id-24;

cout<<"Total no. of subnets: "<<pow(2,s\_id)<<endl;