**Acropolis Institute of Technology and Research**

****

**Department of Computer Science and Information Technology**

**Name : Aditya Polekar**

**Enroll. No : 0827CI191003**

**Branch: C.S.I.T (CI)**

**Subject : Computer Network**

**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)**

**Code :**

#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 : "<<bit;

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

{

cout << binaryNum[j];

}

}

int main()

{

char ch;

cout<<"Enter any character : ";

cin>>ch;

int asci=ch;

cout<<asci<<endl;

decToBinary(asci);

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)**

**Code:-**

#include <iostream>

#include <string>

#include<algorithm>

using namespace std;

string DecToBin(int x)

{

string s;

while (x > 0)

{

if (x % 2 == 1)

{

s += '1';

}

else

{

s += '0';

}

x = x / 2;

}

int n = s.length();

for (int i = 0; i < n / 2; i++)

swap(s[i], s[n - i - 1]);

return s;

}

int main()

{

char ch1,ch2;

cout<<"Enter first character: ";

cin>>ch1;

cout<<"Enter second character: ";

cin>>ch2;

string c1,c2;

c1=DecToBin(int(ch1));

c2=DecToBin(int(ch2));

if(c1.length()<8)

{

reverse(c1.begin(), c1.end());

int x =(8-c1.length());

while (x--)

{

c1+='0';

}

reverse(c1.begin(), c1.end());

}

if(c2.length()<8)

{

reverse(c2.begin(), c2.end());

int x =(8-c2.length());

while (x--)

{

c2+='0';

}

reverse(c2.begin(), c2.end());

}

string lrc;

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

{

if (c1[i]==c2[i])

lrc+='0';

else

lrc+='1';

}

cout<<lrc<<endl;

return 0;

}

**Program 3: Write a program to bit stuff the given binary data. Algorithm for Bit−Stuffing**

**Code:-**

#include <iostream>

using namespace std;

int main()

{

int n;

int count=0;

cout<<"Enter no of bits : ";

cin>>n;

int ar[n];

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

cout<<"Enter bits of data : ";

//Taking input of bites

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**

**Code:**

#include <iostream>

using namespace std;

void decToBinary(int n)

{

// array to store binary number

int binaryNum[32];

// counter for binary array

int i = 0;

while (n > 0) {

// storing remainder in binary array

binaryNum[i] = n % 2;

n = n / 2;

i++;

}

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

cout << binaryNum[j];

}

int main()

{

int first,second,third,fourth;

cin>> first>>second>>third>>fourth;

cout<< decToBinary(first)<<”.”<< decToBinary(second)<<”.”<< decToBinary(third)<<”.”<< decToBinary(fourth);

}

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

**Code –**

#include<iostream>

using namespace std;

char findClass(char str[])

{

// storing first octet in arr[] variable

char arr[4];

int i = 0;

while (str[i] != '.')

{

arr[i] = str[i];

i++;

}

i--;

// converting str[] variable into number for

// comparison

int ip = 0, j = 1;

while (i >= 0)

{

ip = ip + (str[i] - '0') \* j;

j = j \* 10;

i--;

}

// Class A

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

return 'A';

// Class B

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

return 'B';

// Class C

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

return 'C';

// Class D

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

return 'D';

// Class E

else

return 'E';

}

int main(){

char str[] = "192.226.12.11";

char ipClass = findClass(str);

cout<<"Given IP address belongs to Class "<<" : "<<ipClass<<endl;

return 0;

}

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

**Code :**

#include<iostream>

using namespace std;

void con\_dec\_to\_Bin(int num)

{

int binaryArr[8] = {0};

int idx = 7;

while (idx >= 0 )

{

binaryArr[idx] = num % 2 ;

num = num / 2 ;

idx-- ;

}

while( idx++ < 7 )

{

cout<<binaryArr[idx]<<" ";

}

}

int main()

{

int dec\_number;

cout<<"Enter a decimal number: ";

cin>>dec\_number;

con\_dec\_to\_Bin(dec\_number);

return 0;

}

**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)**

**Code :**

#include <iostream>

#include <string.h>

#include <stack>

#include <vector>

#include <math.h>

using namespace std;

// Converts IP address to the binary form

vector<int> bina(vector<string> str)

{

vector<int> re(32,0);

int a, b, c, d, i, rem;

a = b = c = d = 1;

stack<int> st;

// Separate each number of the IP address

a = stoi(str[0]);

b = stoi(str[1]);

c = stoi(str[2]);

d = stoi(str[3]);

// convert first number to binary

for (i = 0; i <= 7; i++)

{

rem = a % 2;

st.push(rem);

a = a / 2;

}

// Obtain First octet

for (i = 0; i <= 7; i++) {

re[i] = st.top();

st.pop();

}

// convert second number to binary

for (i = 8; i <= 15; i++) {

rem = b % 2;

st.push(rem);

b = b / 2;

}

// Obtain Second octet

for (i = 8; i <= 15; i++) {

re[i] = st.top();

st.pop();

}

// convert Third number to binary

for (i = 16; i <= 23; i++) {

rem = c % 2;

st.push(rem);

c = c / 2;

}

// Obtain Third octet

for (i = 16; i <= 23; i++) {

re[i] = st.top();

st.pop();

}

// convert fourth number to binary

for (i = 24; i <= 31; i++) {

rem = d % 2;

st.push(rem);

d = d / 2;

}

// Obtain Fourth octet

for (i = 24; i <= 31; i++) {

re[i] = st.top();

st.pop();

}

return (re);

}

// cls returns class of given IP address

char cls(vector<string> str)

{

int a = stoi(str[0]);

if (a >= 0 && a <= 127)

return ('A');

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

return ('B');

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

return ('C');

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

return ('D');

else

return ('E');

}

// Converts IP address

// from binary to decimal form

vector<int> deci(vector<int> bi)

{

vector<int> arr(4,0);

int a, b, c, d, i, j;

a = b = c = d = 0;

j = 7;

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

a = a + (int)(pow(2, j)) \* bi[i];

j--;

}

j = 7;

for (i = 8; i < 16; i++) {

b = b + bi[i] \* (int)(pow(2, j));

j--;

}

j = 7;

for (i = 16; i < 24; i++) {

c = c + bi[i] \* (int)(pow(2, j));

j--;

}

j = 7;

for (i = 24; i < 32; i++) {

d = d + bi[i] \* (int)(pow(2, j));

j--;

}

arr[0] = a;

arr[1] = b;

arr[2] = c;

arr[3] = d;

return arr;

}

int main()

{

string ipr = "192.168.1.1/24";

// You can take user input here

// instead of using default address

// Ask user to enter IP address of form(x.y.z.t/n)

cout<<"IP address CIDR format is:"<< ipr;

// Separate IP address and n

string str1 = "";

int idx = 0;

int len = ipr.size();

len -= 3;

while(len--){

str1 += ipr[idx];

idx++;

}

cout<<endl;

cout<<"IP Address : " <<str1<<endl;

string str2 = "";

idx++;

str2 += ipr[idx];

idx++;

str2 += ipr[idx];

cout<<"Value of n : "<< str2<<endl;

// IP address

string tr = str1;

// Split IP address into 4 subparts x, y, z, t

//str = tr.split("\\.");

vector<string> str;

string temp;

int n = tr.size();

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

if(tr[i] >= 48 && tr[i] <= 57)

temp +=tr[i];

else{

str.push\_back(temp);

temp = "";

}

}

str.push\_back(temp);

//cout<<str[0]<<endl<<str[1]<<endl<<str[2]<<endl<<str[3]<<endl;

vector<int> b;

cout<<endl;

// Convert IP address to binary form

b = bina(str);

n = stoi(str2);

vector<int> ntwk(32,0);

vector<int> brd(32,0);

int t = 32 - n;

// Obtanining network address

for (int i = 0; i <= (31 - t); i++) {

ntwk[i] = b[i];

brd[i] = b[i];

}

// Set 32-n bits to 0

for (int i = 31; i > (31 - t); i--) {

ntwk[i] = 0;

}

// Obtaining Broadcast address

// by setting 32-n bits to 1

for (int i = 31; i > (31 - t); i--) {

brd[i] = 1;

}

cout<<endl;

// Obtaining class of Address

char c = cls(str);

cout<<"Class : " << c << endl;

// Converting network address to decimal

vector<int> nt = deci(ntwk);

// Converting broadcast address to decimal

vector<int> br = deci(brd);

// Printing in dotted decimal format

cout<<"First Address : " << nt[0] << "." <<nt[1] <<"." << nt[2] <<"." << nt[3]<<endl;

// Printing in dotted decimal format

cout<<"Last Address : " <<br[0] << "." <<br[1] << "." << br[2] <<"." << br[3] << endl;

//Printing Number of Addresses in Block

cout<<"Total Number of Addresses :" <<br[3]-nt[3]+1<<endl;

return 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)**

**Code:**

#include <iostream>

using namespace std;

int main()

{

int hostno;

cin>>hostno;

if(hostno == 16777214){

cout<<"CLASS A"<<endl;

}else if(hostno == 65534){

cout<<"CLASS B"<<endl;

}else if(hostno == 254){

cout<<"CLASS C"<<endl;

}else{

cout<<"CLASS D / CLASS E"<<endl;

}

return 0;

}