U18CO018

Shubham Shekhaliya

Topic : Hamming code

Sub : CN

1-> Implement c++ program to detect and correct error using Hamming code.

Code :-

#include<bits/stdc++.h>

using namespace std;

int calculateRedundantBits(int n) {

    int p= 1;

    while( (1<<p) < n + p +1) p++;

    return p;

}

bool get(char c){

    return c=='1';

}

string generate(string s,char type) {

    int n = s.size();

    int redundantBits = calculateRedundantBits(n);

    bool code[n + redundantBits];

    int k = 0;

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

        if( ((i+1)&i) != 0) {

            code[i] = get(s[k++]);

        }

    }

    k = 1;

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

        if( ((i+1)&i) != 0 ) continue;

        bool parity = false;

        for(int j = i + 2;j<=n+redundantBits;j++) {

            if((j&(i+1)) != 0 ) {

                parity^= code[j-1];

            }

        }

        code[i] = parity^(type != 'E');

        cout<<"P "<< k << ": " << (code[i]?'1':'0') <<endl;

        k++;

    }

    string sb;

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

        sb+=(code[i]?'1':'0');

    }

    return sb;

}

int errorBit(string s,char type) {

    int redundantBits = ceil(log2(s.size()));

    int n = s.size() - redundantBits;

    bool code[n + redundantBits];

    for(int i = 0;i<s.size();i++) {

        code[i] = s[i] == '1';

    }

    int k = 0,c = 0;

    for(int i = redundantBits - 1 ;i >=0 ;i--) {

        k = (1<<(i));

        bool parity = code[k-1];

        for(int j = k + 1;j<= n + redundantBits;j++) {

            if((k&(j)) != 0 ) {

                parity^= code[j-1];

            }

        }

        c<<=1;

        c |= (parity == (type=='E'))?1:0;

    }

    return c;

}

void correction(int p,string s) {

    if(p==0) {

        cout<<"No error Present\n";

        cout<<"Received Code: " << s <<endl;;

        cout<<"Message: ";

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

            if(((i+1)&i) == 0) continue;

            cout<<s[i];

        }

    } else {

        cout<<"Error at: " << p << " 'bit in received Message" <<endl;;

        s[p - 1] = (s[p-1] == '1') ? '0':'1';

        cout<<"Corrected Code: "  + s <<endl;;

        cout<<"Message: ";

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

            if(((i+1)&i) == 0) continue;

                cout<<s[i];

        }

    }

}

int main() {

    cout<<"Enter the Data ";

    string s;

    cin>>s;

    cout<<"Odd or Even Parity:(O/E)";

    char ss;

    cin>>ss;

    string encoded = generate(s,ss);

    cout<<"Encoded data: "<< encoded << endl;;

    cout<<"Enter Recieved Message ";

    cin>>s;

    int p = errorBit(s, ss);

    correction(p, s);

    return 0;

}

Output :-

Example-1 :-

