NAME: VAIBHAV BANKA

REG NO: 21BCE1955

Parity checker

```cpp
#include<iostream>
using namespace std;
int main(){
    int n;
    cout<<"Enter the number of bits:";
    cin>>n;
    cout<<endl;
    int arr[n];
    for(int i=0;i<n;i++){
        cin>>arr[i];
    }
    int res=0;
    for(int i=0;i<n;i++){
        if (arr[i]==1){
            res=res^arr[i];
        }
    }
    if (res==0){
        cout<<"Even parity"<<" ";
        int arre[n+1];
        for(int i=0;i<n;i++){
            arre[i]=arr[i];
        }
        arre[n]=0;
        for(int i=0;i<n+1;i++){
            cout<<arre[i];
        }
        cout<<endl;
        cout<<"Odd parity"<<" ";
        int arre1[n+1];
        for(int i=0;i<n;i++){
            arre1[i]=arr[i];
        }
        arre1[n]=1;
        for(int i=0;i<n+1;i++){
            cout<<arre1[i];
        }

    }
    else {
        cout<<"Odd parity"<<" ";
```

```cpp
        int arro[n+1];
        for(int i=0;i<n;i++){
            arro[i]=arr[i];
        }
        arro[n]=0;
        for(int i=0;i<n+1;i++){
            cout<<arro[i];
        }
        cout<<endl;
        cout<<"Even parity"<<" ";
        int arro1[n+1];
        for(int i=0;i<n;i++){
            arro1[i]=arr[i];
        }
        arro1[n]=1;
        for(int i=0;i<n+1;i++){
            cout<<arro1[i];
        }
    }
}
```

CRC

```cpp
#include<bits/stdc++.h>
using namespace std;
string xor1(string a, string b){
    string result = "";
    int n = b.length();
    for(int i = 1; i < n; i++){
        if (a[i] == b[i])
            result += "0";
        else
            result += "1";
    }
    return result;
```

```cpp
}
string mod2div(string dividend, string divisor){
    int pick = divisor.length();
    string tmp = dividend.substr(0, pick);
    int n = dividend.length();
    while (pick < n){
        if (tmp[0] == '1')
            tmp = xor1(divisor, tmp) + dividend[pick];
        else
            tmp = xor1(std::string(pick, '0'), tmp) +dividend[pick];
        pick += 1;
    }
    if (tmp[0] == '1')
        tmp = xor1(divisor, tmp);
    else
        tmp = xor1(std::string(pick, '0'), tmp);
    return tmp;
}
void encodeData(string data, string key){
    int l_key = key.length();
    string appended_data = (data +std::string(l_key - 1, '0'));
    string remainder = mod2div(appended_data, key);
    string codeword = data + remainder;
    cout << "Remainder : "<< remainder << "\n";
    cout << "Encoded Data (Data + Remainder) :"<< codeword << "\n";
}
void receiver(string data, string key){
    string currxor = mod2div(data.substr(0, key.size()), key);
    int curr = key.size();
    while (curr != data.size()){
        if (currxor.size() != key.size())
            currxor.push_back(data[curr++]);
        else
            currxor = mod2div(currxor, key);
    }
    if (currxor.size() == key.size())
        currxor = mod2div(currxor, key);
    if (currxor.find('1') != string::npos)
        cout << "there is some error in data" << endl;
    else
        cout << "correct message received" << endl;
}
int main(){
    string data;
    string key;
    cin>>data;
    cin>>key;
    encodeData(data, key);
```

```
    return 0;
}
```

```
PS C:\Users\vaibh\Downloads\final> cd "c:\Users\vaibh\OneDrive\Desktop\" ; if ($?) { g++ crc.cpp -o crc } ; if ($?) { .\crc }
100100
1101
Remainder : 001
Encoded Data (Data + Remainder) :100100001
PS C:\Users\vaibh\OneDrive\Desktop> cd "c:\Users\vaibh\OneDrive\Desktop\" ; if ($?) { g++ crc.cpp -o crc } ; if ($?) { .\crc }
10010
101
Remainder : 11
Encoded Data (Data + Remainder) :1001011
PS C:\Users\vaibh\OneDrive\Desktop> []
```