LAB:13

NAME: DHRUV PATEL

ROLL NO: CE111

SUBJECT: NIS

Aim: Write a program to demonstrate Image Steganography operations: Embed and Extract Hide 2 bits per pixel. Hide 3 bits per pixel. Compute MSE (Mean Squared Error) and PSNR (Peak Signal to Noise Ratio) values.

- ## Source Code

```
#include <bits/stdc++.h>
using namespace std;
#define BITS 2
#define DIMS 4
#define MSG DIMS *DIMS *BITS
vector<vector<int>> lsb_embed(vector<vector<int>> img,bitset<MSG> num)
{
    for (int i = 0, k = 0; i < DIMS; i++)
    {
        for (int j = 0; j < DIMS; j++)
        {
            bitset<8> t(img[i][j]);
            for (int l = BITS - 1; l >= 0; l--, k++)
            {
                t[l] = num[MSG - 1 - k];
            }
            string t1 = t.to_string();
            img[i][j] = stoi(t1, 0, 2);
        }
    }
    return img;
}
float MSE(vector<vector<int>> img, vector<vector<int>> stegoImg)
{
    float sum = 0;
    for (int i = 0; i < DIMS; i++)
    {
        for (int j = 0; j < DIMS; j++)
        {
            sum += pow(img[i][j] - stegoImg[i][j], 2);
        }
    }
    return (sum / (DIMS * DIMS));
}
float PSNR(float mse)
{
```

```
    float temp = pow(255, 2) / mse;
    return (10 * log10(temp));
}
long long extract(vector<vector<int>> stegoImg)
{
    string msg = "";
    for (int i = 0; i < DIMS; i++)
    {
        for (int j = 0; j < DIMS; j++)
        {
            bitset<8> t(stegoImg[i][j]);
            for (int l = BITS - 1; l >= 0; l--)
                msg += to_string(t[l]);
        }
    }
    return stoll(msg, 0, 2);
}
int main()
{
    long long n;
    cout << "Enter number in decimal:";
    cin >> n;
    bitset<MSG> num(n);
    cout << "Enter cover Image:\n";
    auto img = vector<vector<int>>(DIMS, vector<int>(DIMS));
    for (int i = 0; i < DIMS; i++)
        for (int j = 0; j < DIMS; j++)
            cin >> img[i][j];
    vector<vector<int>> stegoImg = lsb_embed(img, num);
    cout << "\nStego Image:\n";
    for (vector<int> vect1D : stegoImg)
    {
        for (int pix : vect1D)
        {
            cout << pix << " ";
        }
        cout << endl;
    }
    cout << "\nMSE:";
    float mse = MSE(img, stegoImg);
    cout << mse;
    cout << "\nPSNR:";
    cout << PSNR(mse);
    cout << "\nextracted From Stego Image:";
    cout << extract(stegoImg);

    return 0;
}
```

♦ Input Output

Hide 2 bits per pixel

```
C:\Users\Dhruv j Patel\Desktop\NIS\github\NIS\Lab13>a
Enter number in decimal:95135746
Enter cover Image:
100 101 102 103
200 201 202 203
10 11 12 13
1 2 3 4

Stego Image:
100 100 101 101
202 202 202 203
10 10 14 12
0 0 0 6

MSE:2.1875
PSNR:44.7313
extracted From Stego Image:95135746
```

Hide 3 bits per pixel

Changed In Code

```
#define BITS 3
#define DIMS 4
```

```
C:\Users\Dhruv j Patel\Desktop\NIS\github\NIS\Lab13>a
Enter number in decimal:369258147159
Enter cover Image:
77 74 41 52
63 58 47 96
22 26 24 78
10 20 30 40

Stego Image:
72 72 40 53
58 63 47 97
20 24 24 74
14 21 26 47

MSE:11.75
PSNR:37.4304
extracted From Stego Image:369258147159
```