**Program:**

#include<iostream.h>

#include <bits/stdc++.h>

using namespace std;

int i, j;

char filename[] = "Input\_Image.jpg";

int data = 0, offset, bpp = 0, width, height;

long jpgsize = 0, bmpdataoff = 0;

int\*\* image;

int temp = 0;

// Reading the jpg File

FILE\* image\_file;

image\_file = fopen(filename, "rb");

if (image\_file == NULL) {

printf("Lossy image-Opening File!!");

exit(1);

}

else

{

// Set file position of the

// stream to the beginning

// Contains file signature

// or ID "BM"

offset = 0;

// Set offset to 2, which

// contains size of jpg File

offset = 2;

fseek(image\_file, offset, SEEK\_SET);

// Getting size of jpg File

fread(&bmpsize, 4, 1, image\_file);

// Getting offset where the

// pixel array starts

// Since the information

// is at offset 10 from

// the start, as given

// in jpg Header

offset = 10;

fseek(image\_file, offset, SEEK\_SET);

// jpg data offset

fread(&jpgdataoff, 4, 1, image\_file);

// Getting height and width of the image

// Width is stored at offset 18 and height

// at offset 22, each of 4 bytes

fseek(image\_file, 18, SEEK\_SET);

fread(&width, 4, 1, image\_file);

fread(&height, 4, 1, image\_file);

// Number of bits per pixel

fseek(image\_file, 2, SEEK\_CUR);

fread(&bpp, 2, 1, image\_file);

// Setting offset to start of pixel data

fseek(image\_file, jpgdataoff, SEEK\_SET);

// Creating Image array

image = (int\*)malloc(height \* sizeof(int));

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

{

image[i] = (int\*)malloc(width \* sizeof(int));

}

// int image[height][width]

// can also be done

// Number of bytes in the image

// Image pixel array

int numbytes = (jpgsize - jpgdataoff) / 3;

// Reading the jpg File

// into Image Array

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

{

for (j = 0; j < width; j++)

{

fread(&temp, 3, 1, image\_file);

// the Image is a

// 24-bit jpg Image

temp = temp & 0x0000FF;

image[i][j] = temp;

}

}

}: