





CODE

mainwindow.cpp

#include "mainwindow.h"

#include "ui\_mainwindow.h"

#include "iostream"

static QImage img(500, 500,QImage::Format\_RGB888);

MainWindow::MainWindow(QWidget \*parent)

: QMainWindow(parent)

, ui(new Ui::MainWindow)

{

ui->setupUi(this);

dist0 = 256; // 2^7 = 256, just lesser than label size

x = 0, y = 0;

dist = dist0;

}

MainWindow::~MainWindow()

{

delete ui;

}

void MainWindow::on\_pushButton\_clicked()

{

int degree;

degree = ui->textEdit->toPlainText().toInt(); // degree me text edit store kiya

for (int i=degree;i>0;i--)

dist /= 2; // agar 1 dala toh dist ho jayga 128

SetXY(dist/2, dist/2); // yaha pe x, y allot karenge

HilbertA(degree); // calling Hilbert curve

ui->label->setPixmap(QPixmap::fromImage(img));

}

void MainWindow::SetXY(int xArg, int yArg)

{

this->x = xArg; // x = y = 64

this->y = yArg;

}

void MainWindow::HilbertA(int degree)

{

if(degree>0){

HilbertB(degree-1); // fixed order hota hai ye, nothing to see isme

LineDraw(0, dist); // line draw will draw from x,y to x + ye wale paramters

HilbertA(degree-1);

LineDraw(dist, 0);

HilbertA(degree-1);

LineDraw(0, -dist);

HilbertC(degree-1);

}

}

void MainWindow::HilbertB(int degree)

{

if(degree>0){ // same here

HilbertA(degree-1);

LineDraw(dist, 0);

HilbertB(degree-1);

LineDraw(0, dist);

HilbertB(degree-1);

LineDraw(-dist, 0);

HilbertD(degree-1);

}

}

void MainWindow::HilbertC(int degree)

{

if(degree>0){

HilbertD(degree-1);

LineDraw(-dist, 0);

HilbertC(degree-1);

LineDraw(0, -dist);

HilbertC(degree-1);

LineDraw(dist, 0);

HilbertA(degree-1);

}

}

void MainWindow::HilbertD(int degree)

{

if(degree>0){

HilbertC(degree-1);

LineDraw(0, -dist);

HilbertD(degree-1);

LineDraw(-dist, 0);

HilbertD(degree-1);

LineDraw(0, dist);

HilbertB(degree-1);

}

}

void MainWindow::LineDraw(int changeX, int changeY)

{

DDA(x, y, x + changeX, y + changeY); // calling dda from x,y to x + , y + parameter

x += changeX; // yaha x increment karenge taaki next line bhi bana sake

y += changeY;

}

void MainWindow::DDA(int x1, int y1, int x2, int y2)

{ // standard dda function banaya hai

float x0, y0, xInc, yInc, dx, dy;

QRgb val = qRgb(255,255,0);

dx = x2-x1;

dy = y2-y1;

int step = abs(dx) > abs(dy) ? abs(dx) : abs(dy);

xInc = (x2-x1)/step;

yInc = (y2-y1)/step;

x0 = x1;

y0 = y1;

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

img.setPixel(x0, y0, val);

x0 += xInc;

y0 += yInc;

}

ui->label->setPixmap(QPixmap::fromImage(img));

}

mainwindow.h

#ifndef MAINWINDOW\_H

#define MAINWINDOW\_H

#include <QMainWindow>

QT\_BEGIN\_NAMESPACE

namespace Ui { class MainWindow; }

QT\_END\_NAMESPACE

class MainWindow : public QMainWindow

{

Q\_OBJECT

public:

MainWindow(QWidget \*parent = nullptr);

~MainWindow();

private slots:

void on\_pushButton\_clicked();

void SetXY(int, int);

void HilbertA(int);

void HilbertB(int);

void HilbertC(int);

void HilbertD(int);

void LineDraw(int, int);

void DDA(int ,int ,int ,int);

private:

Ui::MainWindow \*ui;

int dist, dist0, x, y;

};

#endif // MAINWINDOW\_H

main.cpp

#include "mainwindow.h"

#include <QApplication>

int main(int argc, char \*argv[])

{

QApplication a(argc, argv);

MainWindow w;

w.show();

return a.exec();

}

OUTPUT





