#include <iostream>

#include <stdio.h>

#include <stdlib.h>

#include <algorithm>

#include <memory.h>

#include <TH2D.h>

using namespace std;

int main()

{

//Declare array, histogram

static const int A = 100;

double h= 1/A;

double Potentials[A][A];

double Rho[A][A];

double TempPotentials[A][A];

int maxtime = 1000;

int timecount = 0;

TCanvas \*c1 = new TCanvas("c1", "c1",900,900);

c1->cd();

TH2D \*ContourPlot = new TH2D("Contour Plot", "Relaxation Method", A, 0., 1., A, 0., 1.);

//Initialize

for (int i=0; i <= A-1; i++)

{for (int j=0; j <= A-1; j++)

if (i\*i + j\*j >= A)

{

if (i >= A/2 && j >= A/2)

{Potentials[i][j]= 1;

Rho[i][j]=0;}

if (i < A/2 && j > A/2)

{Potentials[i][j]= -1;

Rho[i][j]=0;}

if (i >= A/2 && j < A/2)

{Potentials[i][j]= -1;

Rho[i][j]=0;}

if (i < A/2 && j < A/2)

{Potentials[i][j]= 1;

Rho[i][j]=0;}

}

else

{Potentials[i][j]= 1;

Rho[i][j]=0;}

}

//Loop

while (timecount <= 1000)

{

for (int i=1; i <= A-2; i++)

{for (int j=1; j <= A-2; j++)

{TempPotentials[i][j] = ((Potentials[i][j-1] + Potentials[i-1][j] + Potentials[i][j+1] + Potentials[i+1][j])/4) + (h\*h)\*(Rho[i][j]/4);}

}

for (int j=0; j <= A-1; j++)

{

TempPotentials[0][j]= 0;

TempPotentials[A-1][j]= 0;

}

for (int i=0; i <= A-1; i++)

{

TempPotentials[i][0]= 0;

TempPotentials[i][A-1]= 0;

}

for (int i=0; i<=A-1; i++)

{for (int j=0; j <= A-1; j++)

{

if (i\*i + j\*j < A)

{Potentials[i][j] = TempPotentials[i][j];}

}

}

timecount++;

}

//Print

for (int i=0; i<=A-1; i++)

{for (int j=0; j <= A-1; j++)

{ cout << Potentials[i][j] << " ";

ContourPlot->SetBinContent(i+1,j+1, Potentials[i][j]);

}

cout << endl;

}

gStyle->SetPalette(53);

ContourPlot->Draw("COLZ");

return 0;

}