### 1.1

#include <iostream>

#include <cmath>

#include <cstdlib>

#define DIM 1024

#define DM1 (DIM-1)

#define \_sq(x) ((x)\*(x)) //square

#define \_cb(x) abs((x)\*(x)\*(x)) //absolute value of cube

#define \_cr(x) (unsigned char)(pow((x),1.0/3.0)) //cube root

unsigned char RD(int, int);

unsigned char GR(int, int);

unsigned char BL(int, int);

unsigned char RD(int i, int j){

return (char)(\_sq(cos(atan2(j-512,i-512)/2))\*255);

}

unsigned char GR(int i, int j){

return (char)(\_sq(cos(atan2(j-512,i-512)/2 - 2\*acos(-1)/3))\*255);

}

unsigned char BL(int i, int j){

return (char)(\_sq(cos(atan2(j-512,i-512)/2 + 2\*acos(-1)/3))\*255);

}

void pixel\_write(int, int);

FILE \*fp;

int main()

{

fp = fopen("MathPic1"".ppm","wb");

fprintf(fp, "P6\n%d %d\n255\n",DIM ,DIM);

for(int j = 0; j < DIM; j++)

for(int i = 0; i < DIM; i++)

pixel\_write(i, j);

fclose(fp);

return 0;

}

void pixel\_write(int i, int j)

{

static unsigned char color[3];

color[0] = RD(i, j)&255;

color[1] = GR(i, j)&255;

color[2] = BL(i, j)&255;

fwrite(color, 1, 3, fp);

}

### 1.2

unsigned char RD(int i,int j){

#define r(n)(rand()%n)

static char c[1024][1024];return!c[i][j]?c[i][j]=!r(999)?r(256):RD((i+r(2))%1024,(j+r(2))%1024):c[i][j];

}

unsigned char GR(int i,int j){

static char c[1024][1024];return!c[i][j]?c[i][j]=!r(999)?r(256):GR((i+r(2))%1024,(j+r(2))%1024):c[i][j];

}

unsigned char BL(int i,int j){

static char c[1024][1024];return!c[i][j]?c[i][j]=!r(999)?r(256):BL((i+r(2))%1024,(j+r(2))%1024):c[i][j];

}

### 1.3

unsigned char RD(int i,int j){

float x=0,y=0;int k;for(k=0;k++<256;){float a=x\*x-y\*y+(i-768.0)/512;y=2\*x\*y+(j-512.0)/512;x=a;if(x\*x+y\*y>4)break;}return log(k)\*47;

}

unsigned char GR(int i,int j){

float x=0,y=0;int k;for(k=0;k++<256;){float a=x\*x-y\*y+(i-768.0)/512;y=2\*x\*y+(j-512.0)/512;x=a;if(x\*x+y\*y>4)break;}return log(k)\*47;

}

unsigned char BL(int i,int j){

float x=0,y=0;int k;for(k=0;k++<256;){float a=x\*x-y\*y+(i-768.0)/512;y=2\*x\*y+(j-512.0)/512;x=a;if(x\*x+y\*y>4)break;}return 128-log(k)\*23;

}

### 1.4

unsigned char RD(int i,int j){

double a=0,b=0,c,d,n=0;

while((c=a\*a)+(d=b\*b)<4&&n++<880)

{b=2\*a\*b+j\*8e-9-.645411;a=c-d+i\*8e-9+.356888;}

return 255\*pow((n-80)/800,3.);

}

unsigned char GR(int i,int j){

double a=0,b=0,c,d,n=0;

while((c=a\*a)+(d=b\*b)<4&&n++<880)

{b=2\*a\*b+j\*8e-9-.645411;a=c-d+i\*8e-9+.356888;}

return 255\*pow((n-80)/800,.7);

}

unsigned char BL(int i,int j){

double a=0,b=0,c,d,n=0;

while((c=a\*a)+(d=b\*b)<4&&n++<880)

{b=2\*a\*b+j\*8e-9-.645411;a=c-d+i\*8e-9+.356888;}

return 255\*pow((n-80)/800,.5);

}

### 1.5

unsigned char RD(int i,int j){

static double k;k+=rand()/1./RAND\_MAX;int l=k;l%=512;return l>255?511-l:l;

}

unsigned char GR(int i,int j){

static double k;k+=rand()/1./RAND\_MAX;int l=k;l%=512;return l>255?511-l:l;

}

unsigned char BL(int i,int j){

static double k;k+=rand()/1./RAND\_MAX;int l=k;l%=512;return l>255?511-l:l;

}

### 1.6

unsigned char RD(int i,int j){

float s=3./(j+99);

float y=(j+sin((i\*i+\_sq(j-700)\*5)/100./DIM)\*35)\*s;

return (int((i+DIM)\*s+y)%2+int((DIM\*2-i)\*s+y)%2)\*127;

}

unsigned char GR(int i,int j){

float s=3./(j+99);

float y=(j+sin((i\*i+\_sq(j-700)\*5)/100./DIM)\*35)\*s;

return (int(5\*((i+DIM)\*s+y))%2+int(5\*((DIM\*2-i)\*s+y))%2)\*127;

}

unsigned char BL(int i,int j){

float s=3./(j+99);

float y=(j+sin((i\*i+\_sq(j-700)\*5)/100./DIM)\*35)\*s;

return (int(29\*((i+DIM)\*s+y))%2+int(29\*((DIM\*2-i)\*s+y))%2)\*127;

}

### 1.7

unsigned char RD(int i,int j){

#define D DIM

#define M m[(x+D+(d==0)-(d==2))%D][(y+D+(d==1)-(d==3))%D]

#define R rand()%D

#define B m[x][y]

return(i+j)?256-(BL(i,j))/2:0;

}

unsigned char GR(int i,int j){

#define A static int m[D][D],e,x,y,d,c[4],f,n;if(i+j<1){for(d=D\*D;d;d--){m[d%D][d/D]=d%6?0:rand()%2000?1:255;}for(n=1

return RD(i,j);

}

unsigned char BL(int i,int j){

A;n;n++){x=R;y=R;if(B==1){f=1;for(d=0;d<4;d++){c[d]=M;f=f<c[d]?c[d]:f;}if(f>2){B=f-1;}else{++e%=4;d=e;if(!c[e]){B=0;M=1;}}}}}return m[i][j];

}

### 1.8

unsigned char RD(int i,int j){

#define A float a=0,b,k,r,x

#define B int e,o

#define C(x) x>255?255:x

#define R return

#define D DIM

R BL(i,j)\*(D-i)/D;

}

unsigned char GR(int i,int j){

#define E DM1

#define F static float

#define G for(

#define H r=a\*1.6/D+2.4;x=1.0001\*b/D

R BL(i,j)\*(D-j/2)/D;

}

unsigned char BL(int i,int j){

F c[D][D];if(i+j<1){A;B;G;a<D;a+=0.1){G b=0;b<D;b++){H;G k=0;k<D;k++){x=r\*x\*(1-x);if(k>D/2){e=a;o=(E\*x);c[e][o]+=0.01;}}}}}R C(c[j][i])\*i/D;

}