

frameCountが200になったときに現れるので200と名付けました。

どことなくチューリング・パターンに似ているのは気のせいだろうか。

この作品は配列を用いて作った作品（後ほど紹介する“炎”）をcopy関数で加工したものです。小さいrectを敷き詰めているので、fillとstrokeを変えると様々な色彩パターンを生み出すことが可能です。表紙の画像は白と黒以外の色を設定した例です。

```
float[][] box;
int si_box=600;
int box_r=0;
int ran_n=0;
float ang=0;
void setup() {
  background(255);
  box=new float[si_box][si_box];
  size(600,600);
  for (int y=0; y<si_box; y++) {
    for (int x=0; x<si_box; x++) {
      if (y==0) {
        box[x][y]=(int)random(0,255);
      } else if (y==1 && x>0 && x<si_box-1) {
        box[x][y]=(box[x-1][y-1]+box[x][y-1]+
          box[x+1][y-1])/3;
      } else if(y>1 && x>0 && x<si_box-1) {
        box[x][y]=(box[x-1][y-1]+box[x][y-1]+
          box[x+1][y-1]+box[x][y-2]+
          noise(x,y)*2)/4*noise(x,y,x*y)*2.16;
      }
    }
  }
  for (int stepx=0; stepx<si_box; stepx++) {
    for (int stepy=0; stepy<si_box; stepy++) {
      box_r=600/si_box;
      push();
      translate(stepx*box_r,600-stepx*box_r);
      noStroke();
      fill(255-box[stepx][stepy]);
      rect(0,0,1,1);
      pop();
    }
  }
}
void draw() {
  copy(0,0,width,height,
    -2,-2,width+2,height+2);
}
```

二次元配列を利用して各座標  
で利用する色にかかわる値を  
設定します。その後画面全体  
にrectを敷き詰めます。

色の情報をrectに設定し、  
copy関数で加工します。

