生命游戏 02 说明文档

李南星 2013013286

Github 部署位置：

http://fuyusyogun.github.io/Game_Of_Life/index.html

功能实现

本次实验要求对于前次实验的结果进行改动。

在上次实验中，我们完成了生命游戏基本的功能。通过对一个 map 数组（0 为死细胞，1 为活细胞）的操作，能够随机生成一个游戏棋盘。通过计算数组对应位置的细胞数量，程序能够实现细胞按照声明游戏规则进行生长。之后通过一个固定的刷新函数在一个 canvas 对象上进行重绘。

```
function getNeighbour(i, j){
}
function grow(){
}
function reset(){
}
function initializeMap(x, y, r) {
}
```

我们同时通过对 map 数组的操作和一个刷新标记 flag 实现了游戏的重新生成、暂停、启动和重置功能。同时我们实现了鼠标事件的绑定，能够通过在界面上点击的方式产生、杀死细胞。

我们同时能够时刻显示存活的细胞数。

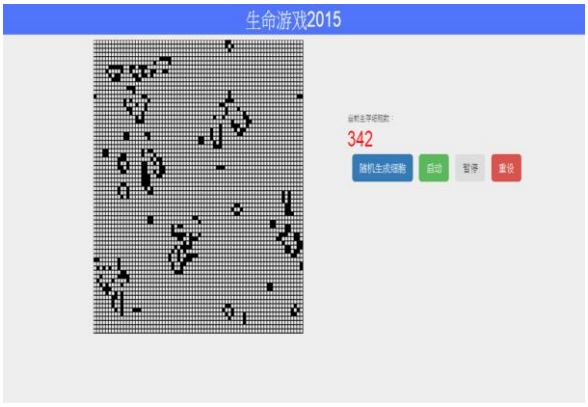
```
function refresh() {
}

function drawMap(x, y){
}

drawMap(width, height);

function continueGrow(){
}

function pauseGrow(){
    flag = 0;
}
```



本次实验要求：

1 更改生命游戏的规则，将相邻细胞的定义进行更改。

2 实现鼠标点击产生墙壁。

3 实现自定义地图尺寸、密度及刷新帧率。

1 对于这项要求，我对原版程序中获取相邻活细胞数的函数 getNeighbour 进行了改动。

```
function getNeighbour(i, j){
    var num = 0;
    for(y = j - 2; y <= j + 2; y++){
        if(y < 0){
            if(map[i][y + height] == 1){
                num += 1;
            }
        }
        else if(y >= height){
            if(map[i][y - height] == 1){
                num += 1;
            }
        }
        else if(map[i][y] == 1)
            num += 1;
    }
    for(x = i - 2; x <= i + 2; x++){
        if(x < 0){
            if(map[x + width][j] == 1){
                num += 1;
            }
        }
        else if(x >= width){
            if(map[x - width][j] == 1){
                num += 1;
            }
        }
        else if(map[x][j] == 1)
            num += 1;
    }
    num -= map[i][j];
    num -= map[i][j];
    return num;
```

2 由于在前次的程序中已经实现了鼠标事件的绑定，这次实验中，我仅仅增加了定义“-1 为墙壁”，并对鼠标事件进行了相应的修改。墙壁在地图上以蓝色显示。

```
canvas.onmousedown = function(e){
    var mouse = getPointOnCanvas(canvas, e.pageX, e.pageY);
    var x, y;
    x = parseInt(mouse.x / scale);
    y = parseInt(mouse.y / scale);
    if(map[x][y] != -1)
        map[x][y] = -1;
    else if(map[x][y] == -1)
        map[x][y] = 0;
    redraw();
}
```

3 由于前次程序的函数中有相应定制尺寸、密度、帧率的接口，这次程序仅仅添加了输入控件和相应的确认、生成函数，并进行了相应的错误处理。

代码改动

基于前次程序的基础上，本次实验改动了细胞的规则（getNeighbour 函数）。

添加了-1 表示墙壁的规则，进而修改了对应的计算规则和显示函数。将鼠标事件的点击改变为添加、消除墙壁。

添加了输入控件以及对应的确认、生成函数（generate 函数）。

实验结果

程序能够很好地完成相应的功能。

打开网页，棋盘进行正常显示。默认生成 100*100，密度为 0.5 的棋盘。帧率默认为 10fps。

上方的输入控件能够自定义游戏参数。

点击生成按钮可以根据参数重新生成（不会消除墙壁）。

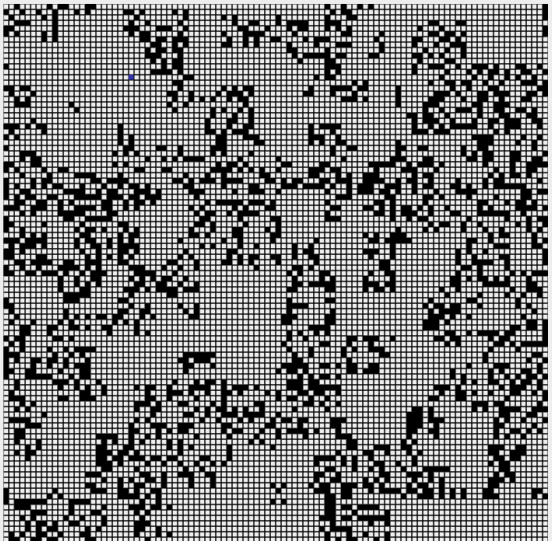
棋盘的尺寸能够智能缩放。

启动、暂停能够控制生长的暂停、继续。

重置能够清空棋盘上的所有方格。

在任一时刻点击棋盘都能添加墙壁。若点击墙壁则能够消除它。

生命游戏2



当前生存细胞数：

2237

宽度 100 px (5 ~ 100, 整数)

高度 100 px (5 ~ 100, 整数)

密度 0.5 (0 ~ 1, 小数)

刷新 10 fps (1 ~ 30, 整数)

生成 启动

暂停 重置