**数据结构课程设计**

**班级：1615403**

**学号：161540205**

**姓名：张佳钰**

**指导老师：孙涵**

1. 扫雷小游戏

采用的数据结构：

静态数组

算法思想：

先随机数生成雷阵

在点击第一下时 将随机将旁边的雷清除

点击雷区时 判断是否为雷

若为雷 弹出 lose

否则 判断是否为0 若不为零 将数字显示

否则 通过广度遍历的方法 将0区显示直到部 为0

源程序：

Html

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">

<title>Bomb</title>

<meta name="description" content="bomb">

<meta name="keywords" content="bomb">

<link href="css/bomb.css" rel="stylesheet">

<link rel="stylesheet" type="text/css" href="../stylesheets/normalize.css" media="screen">

<link rel="icon" type="image/x-icon" href="../ico/bunny.ico">

<link rel="stylesheet" type="text/css" href="../stylesheets/stylesheet.css" media="screen">

<link rel="stylesheet" type="text/css" href="../stylesheets/github-light.css" media="screen">

<script type="text/javascript" src="js/jquery.min.js"></script>

<link href="../public/bootstrap-3.3.6-dist/css/bootstrap.min.css" rel="stylesheet">

<script src="../public/bootstrap-3.3.6-dist/js/bootstrap.min.js"></script>

<script type="text/javascript" src="../public\_js/node\_modules/vue/dist/vue.js"></script>

</head>

<body>

<div class="body-bg-img">

</div>

<div class="container">

<div class="bs-example bs-example-form" style="padding: 20px;">

<div class="col-lg-12">

<h3>扫扫扫扫 雷</h3>

</div>

<div class="col-lg-12">

<h3>

啊哈 扫雷

</h3>

</div>

<div id="ask\_height\_width">

<div class="col-lg-8 col-md-8 col-xs-8"><input type="text" class="col-lg-8 col-md-8 col-xs-8" name="width" id="width" placeholder="width"></div>

<div class="col-lg-8 col-md-8 col-xs-8"><input type="text" class="col-lg-8 col-md-8 col-xs-8" name="height" id="height" placeholder="height"></div>

<div class="col-lg-8 col-md-8 col-xs-8"><input type="number" class="col-lg-8 col-md-8 col-xs-8" name="hard" id="hard" placeholder="难度" max="6" min="2"></div>

</div>

<div class="col-lg-8 col-md-8 col-xs-8"><input type="button" name="submit" value="let's begin" class="begin col-lg-8 col-md-8 col-xs-8" onclick="start()"></div>

</div>

<div class="col-lg-12 col-md-12 col-xs-12" id="record">

<div class="col-lg-12 col-md-12 col-xs-12">

<span>雷的数量</span>

<span>{{ bomb\_num }}</span>

</div>

<div class="col-lg-12 col-md-12 col-xs-12">

<span>

已标记雷的数量

</span>

<span>{{ bomb\_cur }}</span>

</div>

</div>

<div id="bg" style="padding: 30px" class="col-lg-12 col-xs-12 col-md-12">

</div>

<div id="re\_start" style="padding: 30px" class="col-lg-8 col-xs-8 col-md-8">

<input type="button" name="submit" id="re\_start" value="re\_start" class="begin col-lg-8 col-md-8 col-xs-8" onclick="re\_start()">

</div>

</div>

</body>

<script type="text/javascript">

$('#record').hide();

$('#re\_start').hide();

document.oncontextmenu=function(e){return false;}

</script>

<script type="text/javascript" src="js/bomb.js"></script>

</html>

bomb.js：

/// <reference path="jquery.min.js" />

var width=0;

var height=0;

var hard=0;

var flag=parseInt(0);

var a;

var bomb\_num=0;

var bomb\_cur=0;

var correct\_bomb=0;

var on\_going=0;

var visit=new Array();

console.log(a);

var app = new Vue({

el: '#record',

data: {

bomb\_num: bomb\_num,

bomb\_cur: bomb\_cur

}

});

function init\_visit(visit)

{

for(var i=0;i<=height+1;i++)

{

visit[i]=new Array();

for(var j=0;j<=width+1;j++)

{

visit[i][j]=0;

}

}

}

function Create\_bomb(width\_w,height\_w,hard\_w)

{

a=[];

var count=0;

height=height\_w;

width=width\_w;

hard=hard\_w;

console.log(width,height,hard);

init\_visit(visit);

for(var i=1;i<=height\_w;i++)

{

a[i]=new Array();

for(var j=1;j<=width\_w;j++)

{

a[i][j]=Math.floor(Math.random()\*(hard+1));

if(a[i][j]==0)

{

a[i][j]=-1;

count++;

}

}

}

console.log(count);

a[0]=new Array();

a[height+1]=new Array();

for(var i=0;i<=height\_w;i++)

{

a[i][0]=0;

a[i][(width\_w+1)]=0;

}

for(var j=0;j<=width\_w+1;j++)

{

a[0][j]=0;

a[height+1][j]=0;

}

console.log(a);

app.bomb\_cur=0;

app.bomb\_num=0;

return a;

}

function Count\_bomb(a,x,y,x1,y1)

{

var count=0;

for(var i=0;i<8;i++)

{

if(a[x1+x[i]][y1+y[i]]<0)

count++;

}

return count;

}

function random\_start(a,x1,y1)

{

var y=[-1,0,1,1,1,0,-1,-1];//start form left above

var x=[1,1,1,0,-1,-1,-1,0];

for(var i=0;i<x.length;i++)

{

a[x1+x[i]][y1+y[i]]=1;

}

a[x1][y1]=1;

var max\_x=Math.floor(Math.random()\*(width/2));

var max\_y=Math.floor(Math.random()\*(height/2));

for(var i=Math.abs(x1-max\_x);i<(Math.abs(x1-max\_x)+x1);i++)

{

for(var j=Math.abs(y1-max\_y);j<(Math.abs(y1-max\_y)+x1);j++)

{

if(a[i][j]<0)

a[i][j]=Math.floor(Math.random()\*(hard+1))-(6-hard);

}

}

return a;

}

function init\_map(a,width,height)

{

var x=[-1,0,1,1,1,0,-1,-1];//start form left above

var y=[1,1,1,0,-1,-1,-1,0];

var tmp;

bomb\_num=0;

for(var i=1;i<=height;i++)

{

for(var j=1;j<=width;j++)

{

if(a[i][j]>=0)

{

a[i][j]=Count\_bomb(a,x,y,i,j);

}

else

{

bomb\_num++;

}

}

}

app.bomb\_num=bomb\_num;

app.bomb\_cur=0;

}

function get\_matrix()

{

var a=Create\_bomb(20,20);

init\_map(a,20,20);

console.log(a);

return a;

}

function show\_num(x,y,a)

{

var c;

c=(x-1)\*width+y;

console.log(c);

$(`input[id=${c}]`).removeClass("default");

$(`input[id=${c}]`).addClass("flag");

$(`input[id=${c}]`).val(`${a[x][y]}`);

}

function left\_click(a,p\_x,p\_y)

{

//check if it is a bomb

if(a[p\_x][p\_y]<0)

{

alert("bomb loser!");

}

else if(a[p\_x][p\_y]>0)

{

var c;

c=(p\_x-1)\*width+p\_y;

console.log(c);

show\_num(p\_x,p\_y,a);

visit[p\_x][p\_y]=1;

}

else

{

var q=new Array();

var ob;

var y=[-1,0,1,1,1,0,-1,-1];//start form left above//列

var x=[1,1,1,0,-1,-1,-1,0];//行

var next\_x,next\_y;

q.unshift({x:p\_x,y:p\_y});

// show\_num(p\_x,p\_y,a);

while(q.length)

{

ob=q.pop();

show\_num(ob.x,ob.y,a);

if(a[ob.x][ob.y]==0)

{

console.log("ori:",ob.x,ob.y);

for(var i=0;i<x.length;i++)

{

var new\_x,new\_y;

new\_x=ob.x+x[i];

new\_y=ob.y+y[i];

console.log("x:",ob.x,"y:",ob.y,"new\_x:",new\_x,"new\_y:",new\_y,"width:",width,"height:",height);

if(new\_x>0&&new\_x<=height&&new\_y>0&&new\_y<=width)

{

console.log("yes",new\_x,new\_y,'p');

if(a[new\_x][new\_y]>=0)

{

show\_num(new\_x,new\_y,a);

//show\_num(new\_x,new\_y,a);

console.log(new\_x,new\_y,a[new\_x][new\_y],visit[new\_x][new\_y]);

if((a[new\_x][new\_y]==0)&&(visit[new\_x][new\_y]==0))

q.unshift({x:new\_x,y:new\_y});

console.log(" q.length: "+q.length);

visit[new\_x][new\_y]=1;

}

}

}

}

}

}

}

function right\_click(a,x,y)

{

var c;

c=(x-1)\*width+y;

console.log(c);

if($(`input[id=${c}]`).hasClass("bomb\_flag"))

{

$(`input[id=${c}]`).removeClass("bomb\_flag");

$(`input[id=${c}]`).addClass("default");

bomb\_cur--;

if(a[x][y]<0)

{

correct\_bomb--;

}

}

else

{

$(`input[id=${c}]`).removeClass("default");

$(`input[id=${c}]`).addClass("bomb\_flag");

bomb\_cur++;

if(a[x][y]<0)

correct\_bomb++;

}

if(bomb\_cur==bomb\_num&&bomb\_num==correct\_bomb)

alert("Congradulation!!!!!");

}

function check\_bomb(id)

{

var e=window.event;

var j=id%width;

if(j==0)

j=width;

var i=(id-j)/width+1;//i行j列

if(e.which==3)

console.log("rigth");

console.log("click: "+e.which);

console.log(i+" "+j);

visit[i][j]=1;

if(flag==1)

{

console.log('bomb');

if(a[i][j]<0)

// alert("bomb");

console.log(a);

console.log(i+" "+j);

if(e.which==1)

left\_click(a,i,j);

else if(e.which==3)

right\_click(a,i,j);

}else{

console.log('init');

bomb\_cur=0;

a=random\_start(a,i,j);

init\_map(a,width,height);

left\_click(a,i,j);

flag=1;

}

app.bomb\_cur=bomb\_cur;

// $('#record').show();

}

function re\_start()

{

flag=0;

on\_going=0;

correct\_bomb=0;

$('#bg').empty();

$('#record').hide();

$('#re\_start').hide();

$('#ask\_height\_width').show();

console.log('p');

}

function start()

{

if(on\_going==0)

{

console.log("start");

var flag=0;

var text='';

var width=$('#width').val();

var height=$('#height').val();

var hard=$('#hard').val();

if($.trim(width)==''||$.trim(height)==''||$.trim(hard)=='')

{

alert('请输入雷阵的宽和高和难度');

}

else

{

height=parseInt(height);

width=parseInt(width);

hard=parseInt(hard);

var a=Create\_bomb(width,height,hard);

on\_going=1;

console.log(a);

var count =0;

for(var i=1;i<=height;i++)

{

for(var j=1;j<=width;j++)

{

count++;

text+=`<input type="button" class="default" value=" " name="${count}" id="${count}" onmousedown="check\_bomb(this.id)">`;

}

text+=`</br>`;

}

$('#bg').append(text);

$('#record').show();

$('#re\_start').show();

$('#ask\_height\_width').hide();

}

}

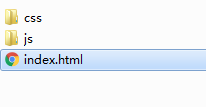
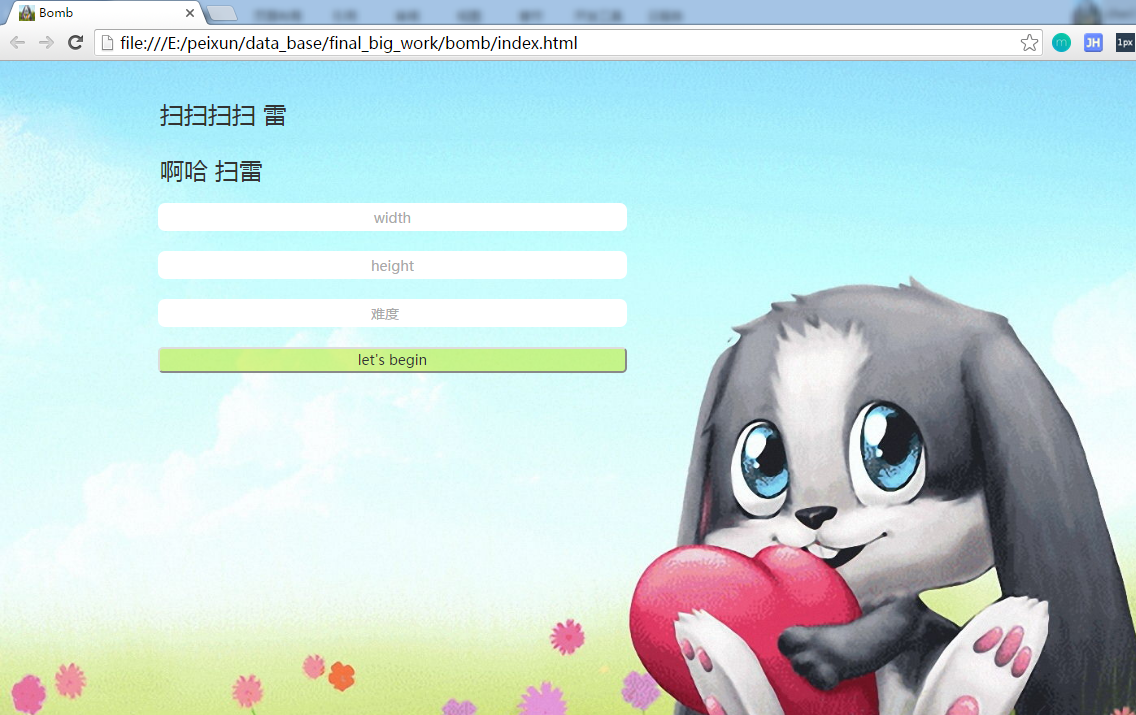
}

函数主要在 bomb.js 中

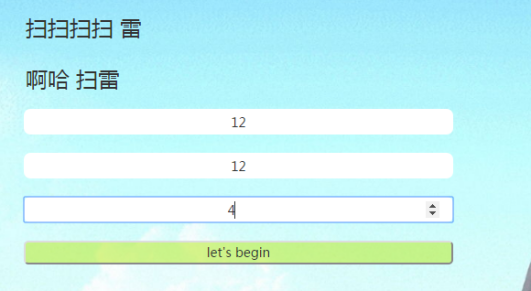
代码行数：

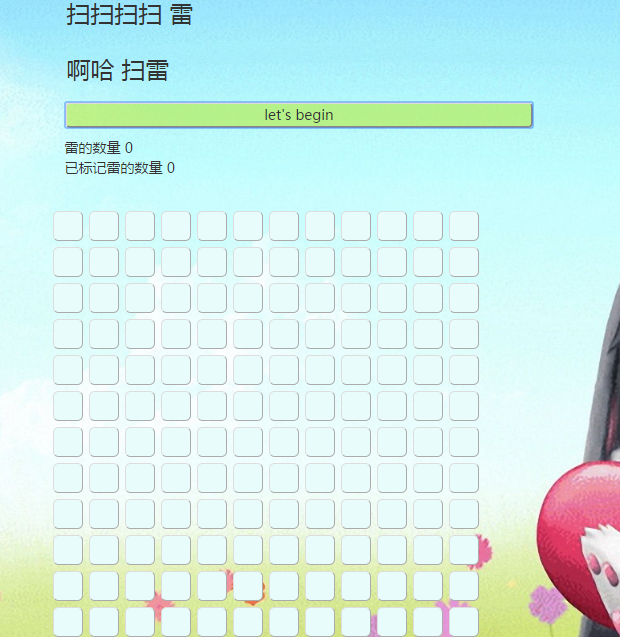
321行

//打开方式

1. 
2. 双击 index.html(建议用chrome打开)
3. 
4. 输入 width 和 height 还有 难度

如图



1. 点击 let’s begin
2. 

然后就可以扫雷了



//css样式在css/bomb.css中

//若查看代码可用sublime text 查看 很方便

写扫雷的感想：

几个星期前 意外发现 原来我的电脑上也有扫雷 就开始玩起来了 然后就停不下来了 本来打算看书的 因为没有一次成功过 好失败 就这样 连续扫了好几天 在失败了无数次之后 想想 恩 我应该也可以写个扫雷 对 写个扫雷 然后 课设题目 可以自己定 好开心 那就写扫雷好了 要动态显示雷的数量就又学了Vue组件 初步写完之后 自己玩了几局 发现不少bug 改好之后 就放一边了 然后 有同学玩 又发现了问题 原来是重新开始的初始化没完全初始化 然后改着改着到现在这样了 。