**《Web程序设计》**

**开放实验报告**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **实验日期** | | **2020年4月10日** | | |
| **班级** | **姓名** | | **学号** | **学院** |
| **电气810** | **聂永欣** | | **2186113564** | **电气学院** |
|  |  | |  |  |
|  |  | |  |  |

**目录**

[一、实验目的 3](#_Toc37097018)

[二、实验概述 3](#_Toc37097019)

[三、实验环境 3](#_Toc37097020)

[四、实验内容 3](#_Toc37097021)

[(一) PLANTS VS ZOMBIES 3](#_Toc37097022)

[1. HTML内容 3](#_Toc37097023)

[1) HTML中所使用的标签 3](#_Toc37097024)

[2) HTML中所使用的响应 3](#_Toc37097025)

[3) HTML中所使用js的函数： 3](#_Toc37097026)

[4) HTML结构 4](#_Toc37097027)

[2. JavaScript内容 4](#_Toc37097028)

[1) Cfunction.js的解释 4](#_Toc37097029)

[3. 实际运行效果 6](#_Toc37097030)

[五、实验总结 7](#_Toc37097031)

[(一) 完成目标 7](#_Toc37097032)

[(二) 实验收获 8](#_Toc37097033)

# 一、实验目的

运用web程序设计有关知识详细解读游戏“植物大战僵尸”的实现过程，并将游戏改编、加工。

# 二、实验概述

小组先在网络上选择了一款“植物大战僵尸”的小游戏，运用所学知识对小游戏的实现过程进行了详细的解读，将其做了一定改编，并通过图像处理将游戏“定制”。

# 三、实验环境

Windows10 64位、Chrome浏览器

# 四、实验内容

## PLANTS VS ZOMBIES

### HTML内容

1. HTML中所使用的标签

<img />：绘制显示页面以及棋格

<div></div>：显示文字信息

<form></form>

<input />：显示单选框、按钮、文本框控件

1. HTML中所使用的响应

Onclick：提供按钮功能响应

ondragstart事件在用户开始拖动元素或选择的文本时触发

oncontextmenu='return false'取消鼠标右键的响应

onselectstart几乎可以用于所有对象，其触发时间为目标对象被开始选中时

onMouseOver鼠标移动到区域上时触发的响应

onMouseOut鼠标移开时触发的响应

onchange在域的内容改变时事件发生的响应

onMouseDown鼠标点下时触发的响应

onMouseUp鼠标抬起时触发的响应

1. HTML中所使用js的函数：
2. PauseGame（）暂停游戏的函数，onclick之和弹出一个返回游戏的窗口的函数
3. OnClickMenu（）通过onclick触发，在游戏中显示菜单的函数
4. OptionsMenuDown（）通过菜单中的返回游戏，由onMouseDown触发，实现返回游戏功能的函数
5. ShowLevel（）点击开始冒险，通过onclick触发，弹出选择关卡窗口界面的函数
6. HidenLevel（）关闭关卡选项，通过onclick触发，返回主界面
7. ShowMiniGame（）通过onclick触发，显示小游戏的选项
8. HiddenMiniGame（）通过onclick触发，关闭选择界面，返回主界面的函数
9. ShowHelp（）通过onclick触发，弹出帮助的函数
10. ShowOptions（）通过onclick触发，弹出游戏选项的函数
11. HTML结构
12. 进度条模块
13. 选项界面模块
14. 主界面
15. 背景图片
16. 植物卡牌系列
17. 出场的僵尸
18. 选择植物卡片
19. 阳光和铲子
20. 菜单
21. 植物图鉴
22. 活动新植物

### JavaScript内容

总共有4个Js文件，以下是分别对它们的解释：

1. Cfunction.js的解释
2. **oSym 控制游戏加载与游戏超时的类**

1.Init（）对游戏系统初始化

2.Clear（）清空游戏缓存

3.Start（）若检测正确，则运行游戏，反之退出

4.Stop（）游戏异常则退出

1. **oS 初始化游戏界面的类**

1.Init（）初始化所使用的函数

2.LoadProgress（）载入游戏所使用的函数

3.InitPn（）LoadImage（）CheckImg（）共同使用以使游戏的图片正常显示

1. **oP 控制游戏过程、僵尸第几波进攻、场上植物的类**

1.Init（）对僵尸生成进行初始化

2.SelectFlagZombie（）与旗手僵尸有关的函数

3.SetTimeoutTomZombie（）SetTimeoutTomZombie（）最后一波僵尸到来有关的函数

4.AddZombiesFlag（）加入旗手僵尸的函数

5.FlagPrgs（）升起一大波僵尸即将到来的函数

6.MonPrgs（）Monitor（）玩家、操控者的反应的函数

1. **oGd 疯狂的戴夫的商店的类**

1.Init（）对商店物品和玩家金钱进行初始化的函数

2.add（）购买函数

3.del（）取消函数

4.MoveBullet（）结账函数

5.MB1（）MB2（）MB3（）购买物品及后续拓展添加可购买物品有关的函数

1. **oZ 控制僵尸攻击行为、攻击范围、移动、血量的类**

1.Init（）初始化僵尸属性的函数

2.add（）僵尸攻击造成伤害的函数

3.getZ（）得到僵尸的现有属性的函数

4.getArZ（）判断僵尸是否死亡的函数

5.moveTo（）僵尸移动的函数

6.traversalOf（）和障碍物有关的函数

1. **oT 在游戏构成中玩家通过鼠标操纵影响场上局势，如种植植物等的类**

1.Init（）初始化种植植物的信息的函数

2.add（）种植植物到战场上的函数

3.check（）判断植物是否死亡的函数

4.delP（）判断植物是否冷却时间刷新的函数

5.indexOf（）与植物图鉴有关的函数

1. **以下是一些其他的细碎的函数总集：**
2. $User（）返回用户操作系统和浏览器信息的函数
3. NewO（） 返回一个prototype对象，为以后的继承类做好准备的函数
4. InheritO = function(d, i, c, g, b, h, f, e, a) 继承d对象，为d对象添加新的属性和方法
5. asyncInnerHTML（）对应开始页面上的按钮功能的函数
6. WhichMouseButton（）检测用户鼠标类型及所指向区域的函数
7. GroundOnmousedown（）对用户鼠标点下后的响应做出反应的函数
8. GroundOnmousemove（）鼠标进入植物区域时触发的函数
9. DisplayZombie（）将僵尸放进战场的函数
10. AutoSelectCard（）自动选择卡片的函数
11. InitPCard（）初始化选择卡片的函数
12. ViewCardTitle（）观看卡片属性的函数
13. SelectCard（）在对战开始前选择卡片的函数
14. ResetSelectCard（）将已选择的卡片取消选择的函数
15. LetsGO（）开始本次对战的函数
16. ViewPlantTitle（）观看植物属性的函数
17. BeginCool（）植物种植后进入冷却的函数
18. DoCoolTimer（）冷却计时函数
19. ChosePlant（）拖拽植物卡片并放入战场的函数
20. CancelPlant（）铲掉植物的函数
21. ShovelPlant（）调用小铲子图标的函数
22. ChoseShovel（）选择小铲子的函数
23. GrowPlant（）默认种植植物的函数
24. AutoProduceSun（）从天上自动生成阳光的函数
25. AppearSun（）出现阳光并调用动态阳光图片的函数
26. DisappearSun（）阳光到时间自动消失的函数
27. ClickSun（）MoveClickSun（）手动收集阳光的函数
28. AutoClickSun（）自动收集阳光的函数
29. ShowLargeWave（）显示一大波僵尸即将到来.gif的函数
30. ShowFinalWave（）显示最后一波.gif的函数
31. LawnMowerKill（）与割草机，判断何时有僵尸到达家门口并发动割草机的函数
32. GameOver（）判断何时有僵尸进入屋子并调用游戏结束.gif的函数
33. **cPlants.js的解释**

通过NewO（）定义一个cPlant父类，再通过InheritO = function(d, i, c, g, b, h, f, e, a)继承得到各种功能不同、形态不同的植物类，基本的功能均是通过调用cfunction的函数实现的，以下对出现的几个方法进行一定的解释：

1. Birth（）定义一个植物的初始状态
2. Die（）定义如何判断一株植物死亡的函数
3. PrivateBirth（）定义一个虚函数，为以后的多态性提供接口
4. PrivateDie（）定义一个虚函数，为以后的多态性提供接口
5. **cZombie.js的解释**

类似于NewO对植物父类初始化的操作，得到所有的僵尸父类，依次继承得到具体的各个僵尸类。

1. **Process.js的解释**

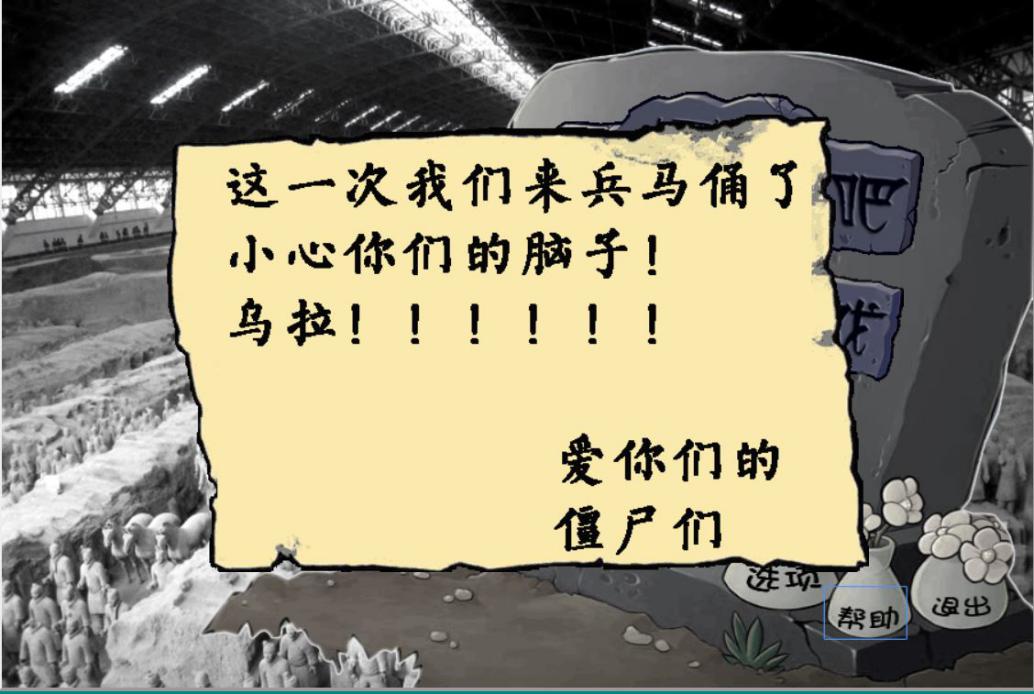
与显示Html中的页面有关的js文件

### 实际运行效果

1. 打开界面：删除了原来的版权文字部分



1. 二级界面：重写了“帮助”，将其中背景图片更换



1. 关卡选择界面：



1. 游戏初始界面：背景更换；另外，作为测试，将草地更换为了瓷砖



1. 游戏界面：可以看到植物所需的阳光值以及冷却时间都已经被修改为0，实现了无敌版的设计目的



# 五、实验总结

## 完成目标

* 完成了对游戏“植物大战僵尸”的解读
* 实现了对游戏“植物大战僵尸”的修改

## 实验收获

* 加深掌握了html各标签及各属性的作用及运用
* 加深掌握了JavaScript语言结构及设计
* 加深了对CSS的理解和使用
* 能初步完成简单的html页面、程序设计
* 增强了HTML、JavaScript文件的理解能力