实验四报告

by 蔡燕芝 12级计科一班 12348006

一、实验目的：

1. 学习中断机制知识，掌握中断处理程序设计的要求。

2. 设计时钟中断处理程序

3. 扩展MyOS1，增加一个系统服务的实验。

4. 建立具有简单异步事件处理的原型操作系统。

二、实验内容：

设计满足下列要求的原型操作系统：

(1)操作系统工作期间，在屏幕24行79列位置轮流显示’|’、’/’和’\’，适当控制显示速度，以方便观察效果。

(2)用户程序运行时，键盘事件有事反应：当键盘有按键时，屏幕适当位置显示”OUCH! OUCH!”。

(3)在内核中，对33号、34号、35号和36号中断编写中断服务程序，程序服务是分别在屏幕1/4区域内显示一些个性化信息。再编写一个汇编语言的程序，利用int 33、int 34、int 35和int 36产生中断调用你这4个服务程序。

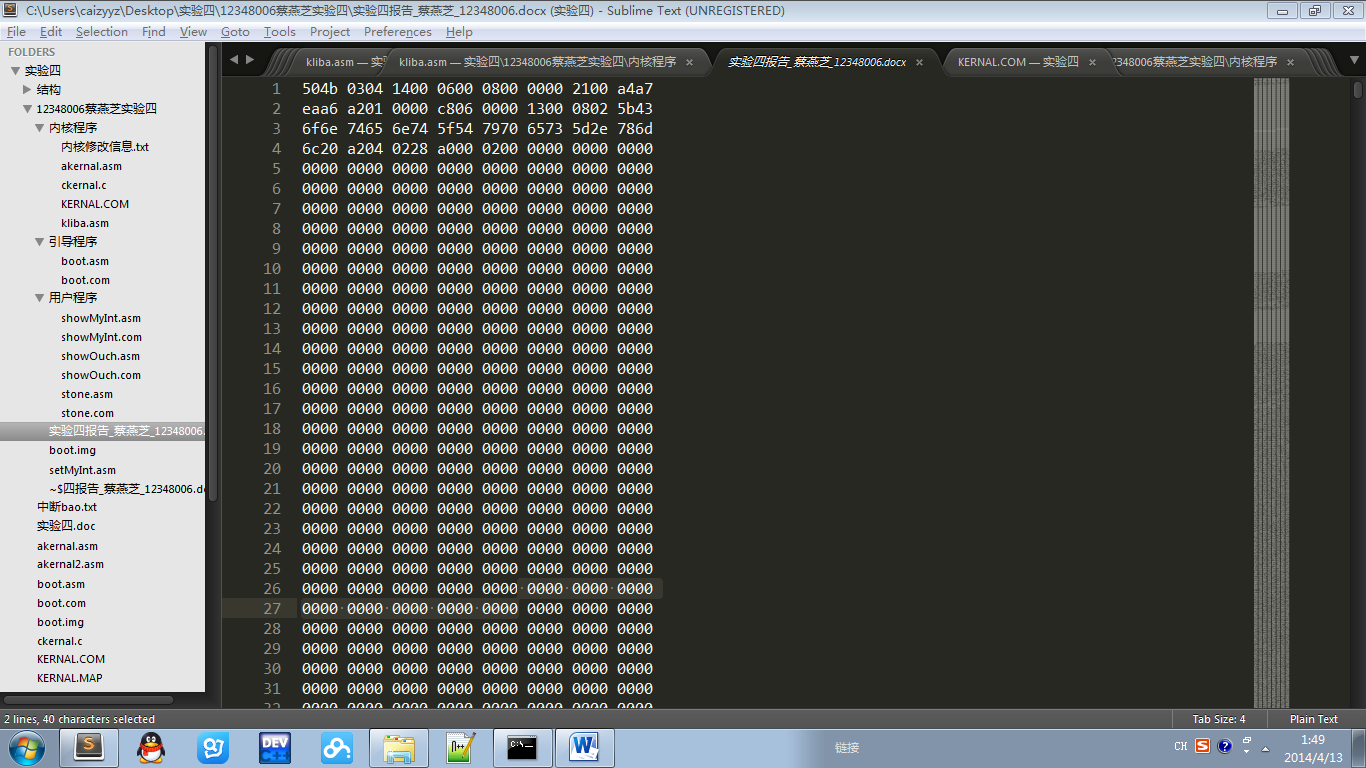
三、实验报告

此次实验在上一个实验的基础上，添加了中断例程，为用户程序提供服务，而这一次按照要求，33号、34号、35号和36号中断提供的服务是在屏幕1/4区域内显示一些个性化信息，8号中断是操作系统工作期间(包括没有用到的8号中断的用户程序运行期间)，在屏幕24行79列位置轮流显示’|’、’/’和’\’。

**以下是操作系统使用的规则如下：**



**以下是这一次实验的文件清单**

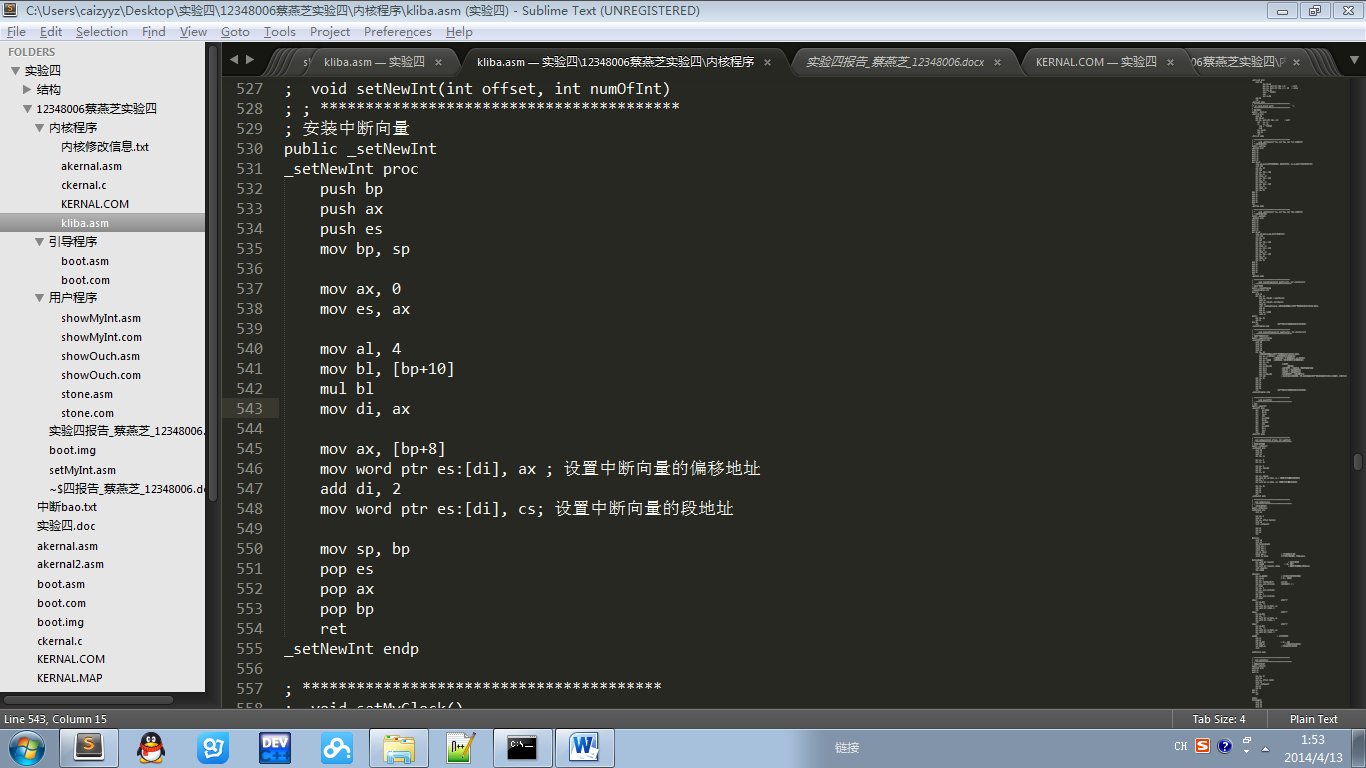


boot.img是软盘文件，其结构如下:

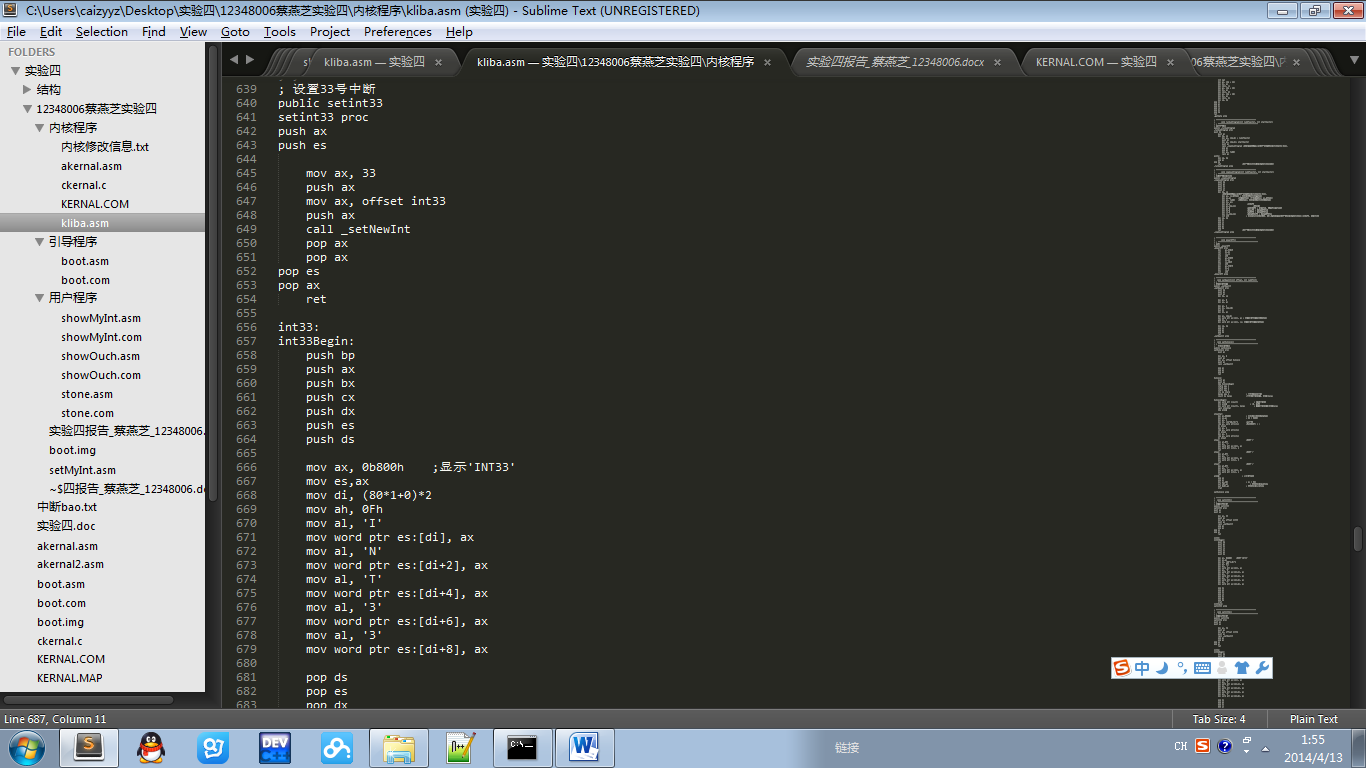
引导扇区程序512字节，存放在第一个扇区中，从第二个扇区开始的10个扇区存储操作系统的内核，之后的扇区存储用户程序showOuch, showMyInt, stone 依次分别占2 2 3个扇区

**下面是关于程序的实现：**

关于中断向量的安装:



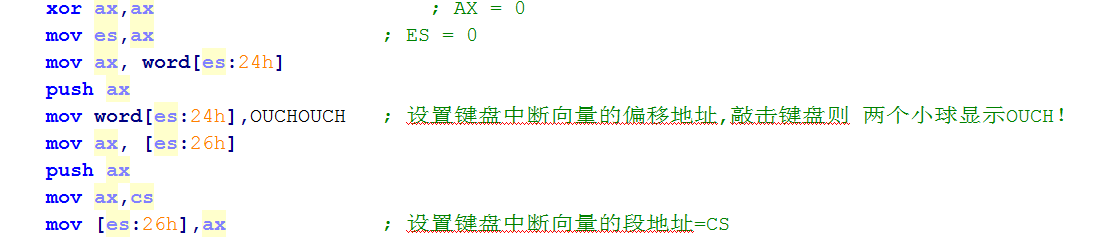
中断向量的定义：

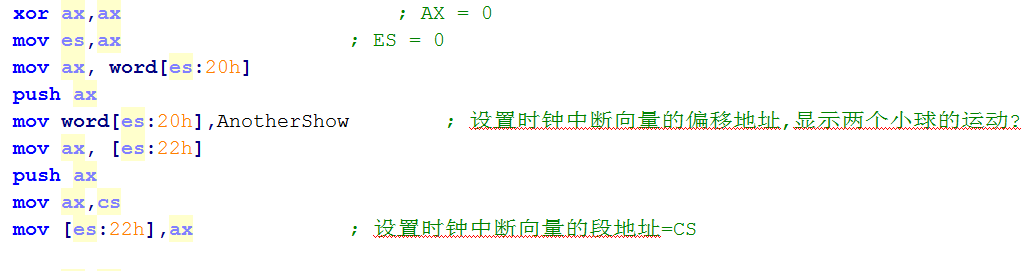


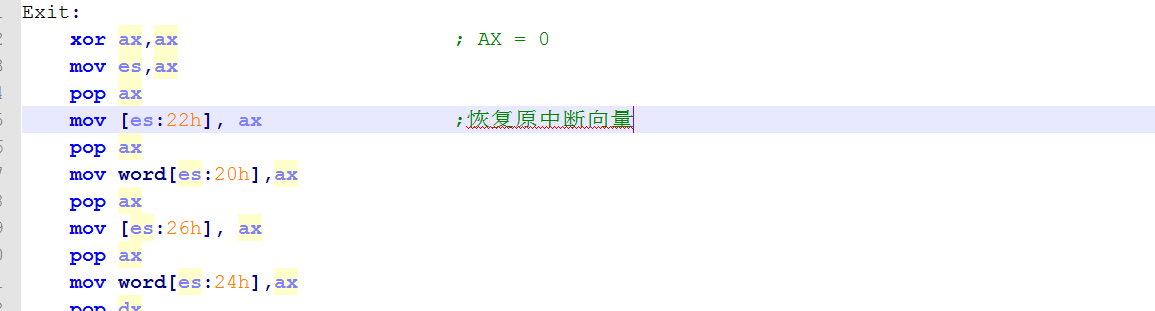
调用setInt33来安装33号中断向量，其他的自定义的中断的方法和这个类似。

在用户程序中：

定义新的中断必须要注意的是，将原中断向量保存起来，然后退出的时候要恢复所用的中断向量，否则这个中断号将无法被其他程序正确的使用！如stone.asm 中用到了8号和9号中断后，必须保存和恢复原来的中断向量，如下所示：



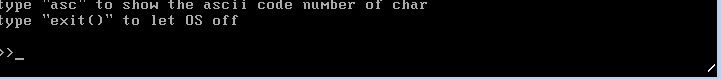


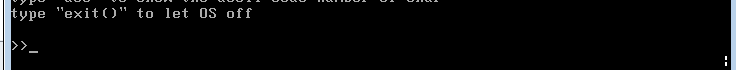


**以下是效果的展示：**

在操作系统运行期间：

屏幕右下角轮流显示 / | \：









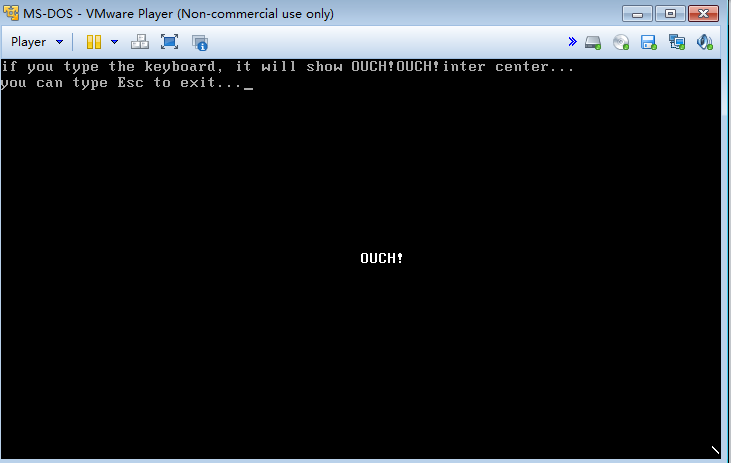
清单中三个用户程序是用于显示本次的中断服务，showMyInt是显示int 33 34 35 36的服务，按Ese键返回;

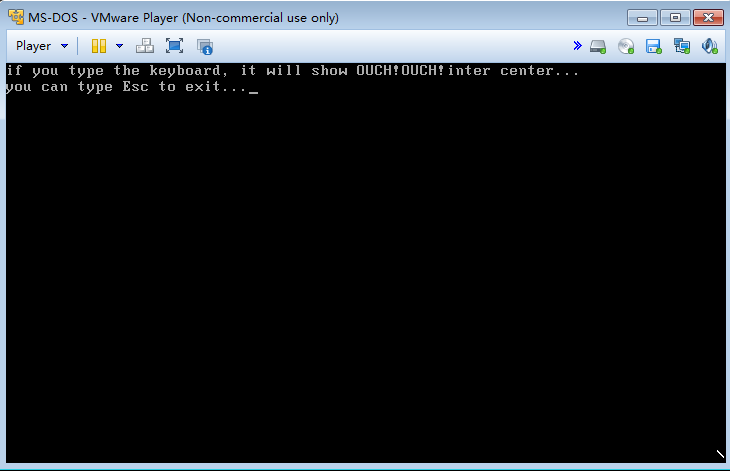
showOuch是敲击键盘就让屏幕的OUCH！闪动，按Ese键返回;

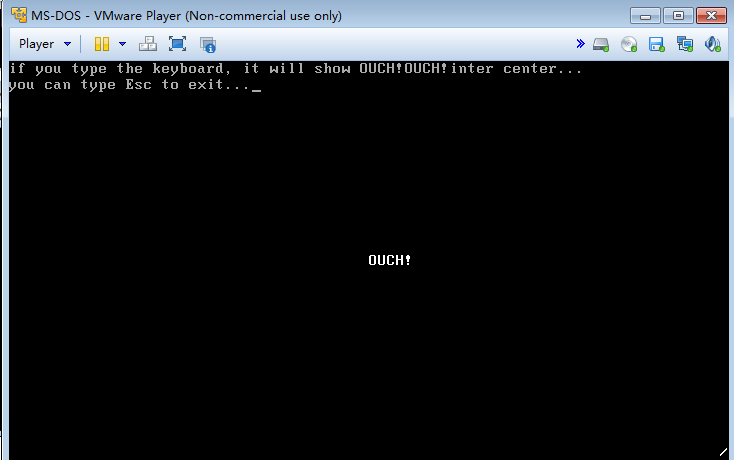
stone文件是之前第一个实验的石头加上了时钟和键盘中断后显示的结果

**效果分别如下：**

showOUCH：

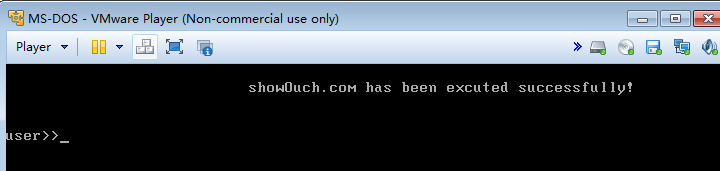




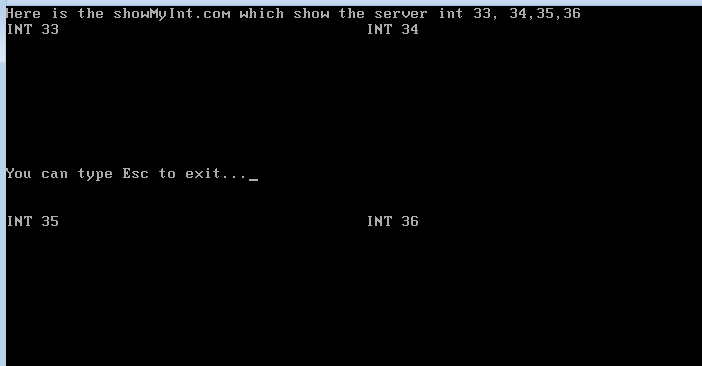


看到OUCH！在闪动。还可以看到在屏幕24行79列位置轮流显示’|’、’/’和’\’

按Ese键返回：



showMyInt的效果：

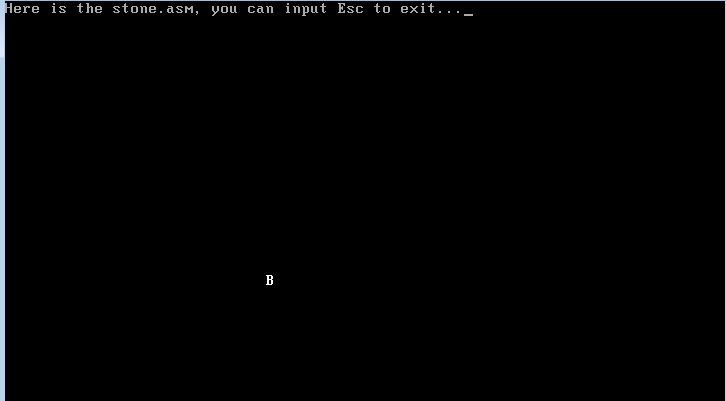


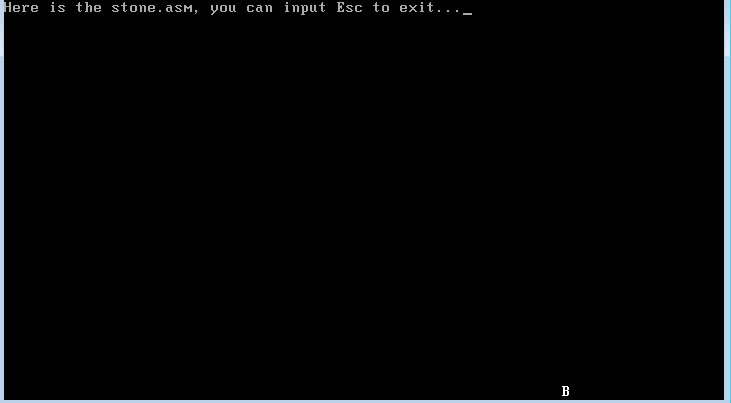
可以看到在1/4区域出现了字符串！（其中的提示信息是该用户程序打印出来的，int 33 等则是调用了中断程序）

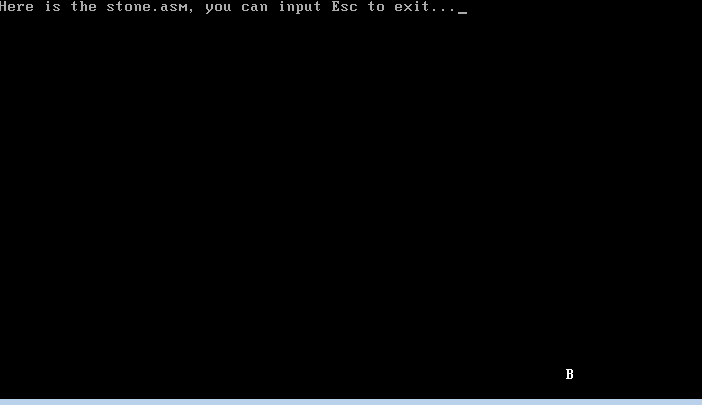
按Ese键返回：

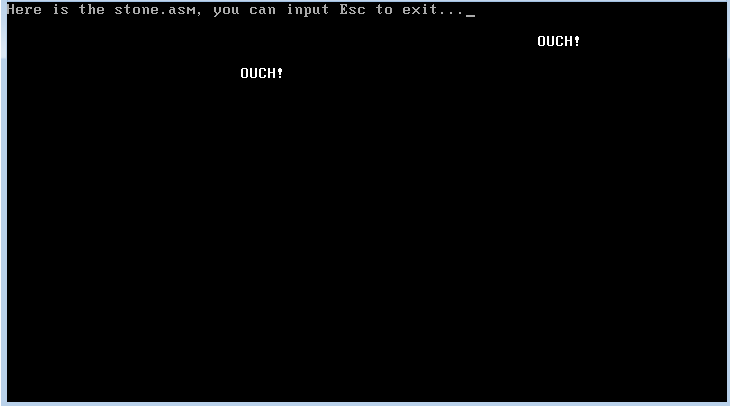


stone：









由于速度太快截屏截取不到效果，上面可以看到B在运动，但是实际上是有两个小球A B在一起运动，碰到边界就会被反弹，而按键盘的时候，两个小球运动的位置会出现OUCH！来响应键盘操作

按Ese返回：

