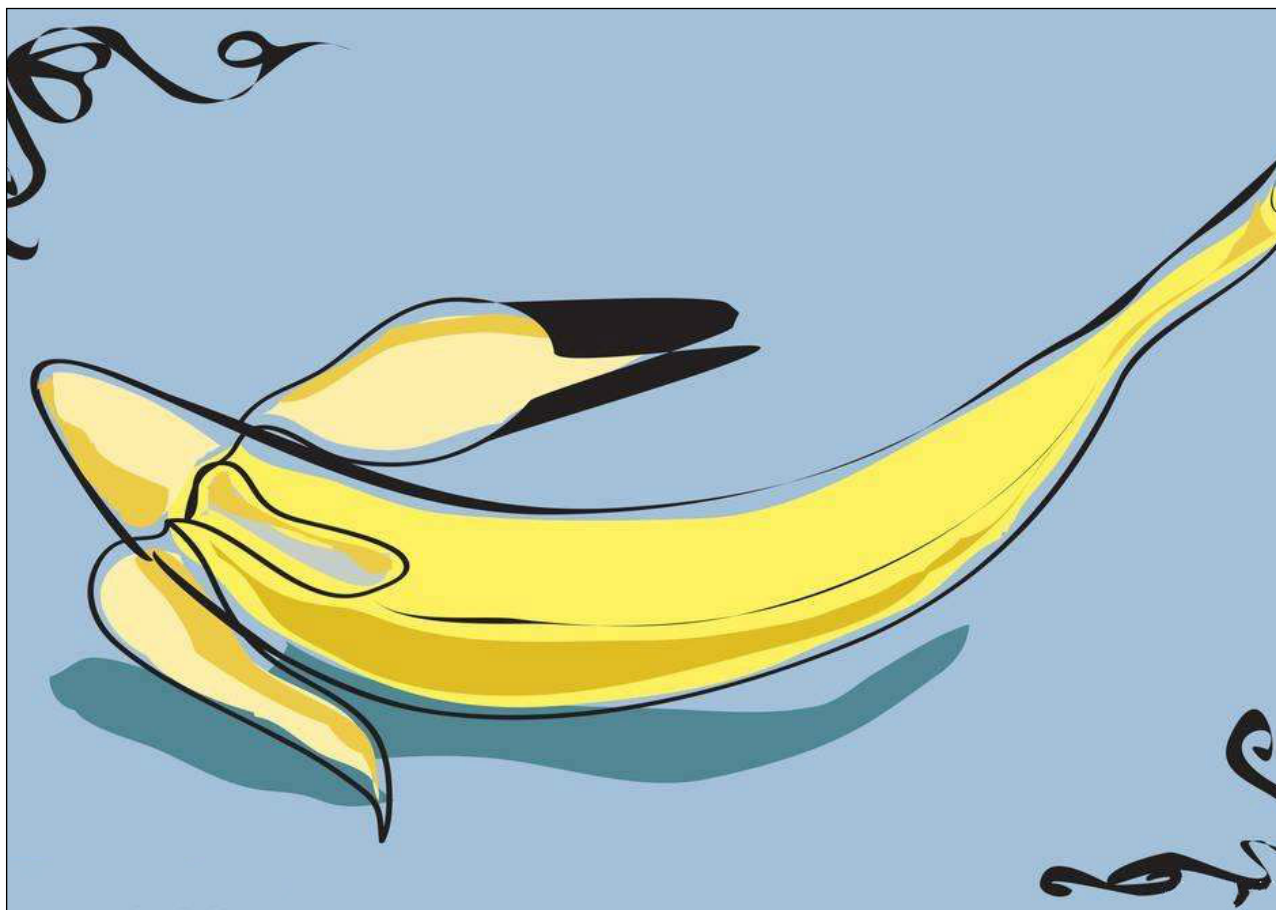

操作系统课程设计

Banana OS 项目设计和功能说明文档

孙晓青 1552683 2018年9月5日



项目开发环境

1. 编写语言

- C语言
- 汇编语言

2. 开发环境

- Ubuntu 16.04 (32位)
- Bochs 2.6

3. 运行环境

- Windows 8
- VMware Workstation

Banana OS 设计说明

为什么叫“Banana”

苹果（Apple）公司开发的Mac OS是全世界第一个基于FreeBSD系统、采用“面向对象操作系统”的全面的操作系统，整个设计处处体现着简洁的宗旨。为了致敬苹果，并且希望自己在学习Orange's的过程中开发的这款操作系统也能带来同样优美流畅的用户体验，因此将其命名为“Banana OS”，以鞭策自己不断前进。

操作系统设计思路

Banana OS 主要是在参考源码的基础上实现了文件系统，并对shell 做了改进，还实现了几个系统级指令和多个用户级应用。

在通读整本Orange's之后，我发现书上提供的源码并没有完善的文件管理，所以首先就考虑修改文件系统。我在Orange's原有文件系统的基础上建立了文件目录，实现了文件

系统的多级管理，还添加了文件的增删改查的指令，允许用户在当前目录下查看文件、进入某个文件夹或者返回上一级目录。除此之外，为了让 Banana OS 更加丰富有趣，我还加入了开机动画和扫雷、推箱子、计算器等应用，这样就形成了一个类似 linux 风格的shell。

操作系统组成

- boot（引导）
- kernal（内核）
- command（应用集）
- fs（文件系统）
- lib（可用代码库）
- include（头文件集）
- mm（内存调度系统）

系统工作流程

从软盘引导 —> 在软盘中查找 Loader.bin —> 加载 loader.bin
—> 跳转 loader.bin 中的代码开始执行 —> 在软盘中查找系统内核 kernel.bin
—> 进保护模式 —> 加载 kernel.bin —> 跳转 kernel.bin 中的代码开始执行 —> 更新 GDT
—> 初始化 IDT —> 初始化 TSS —> 跳入系统主函数 —> 启动系统进程 —> 开启时钟中断
—> 开始进程调度 —> (系统开始运转)

Banana OS 基本功能说明

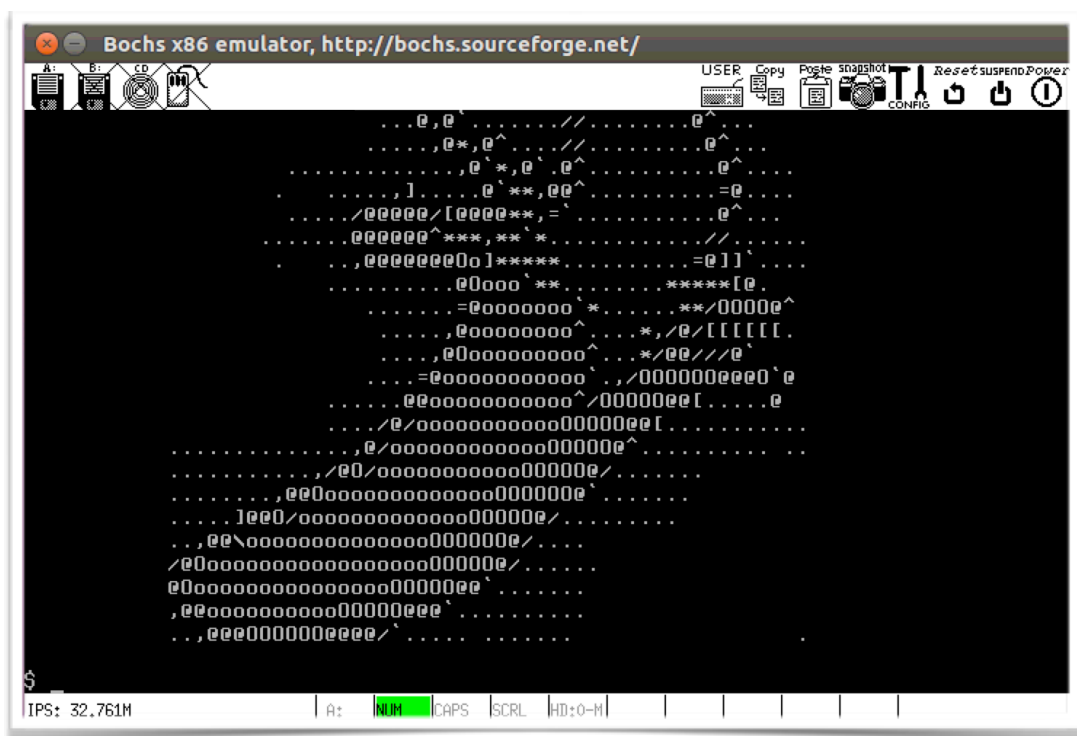
功能汇总

命令	概述
animation	播放开机动画
ls	输出当前目录下的文件列表

help	输出所有可用命令
mineSweeping	开始扫雷游戏
Boxman	开始推箱子游戏
getDate	进入获取天数应用程序
cdfolder	打开文件夹
calculator	浮点计算器
mkfile	新建文件
mkfolder	新建文件夹
catfile	读取文件内容
rmfile	删除文件
editfile	写文件（覆盖原有内容）
cdup	返回上一级目录

1. animation

输入 animation 命令后，播放开机动画（闪烁的香蕉）



2. ls

输入 ls 命令后，输出当前目录下的所有文件

```
$ ls
home:
$ mkfile
Please input the file NAME:
1.txt
New file created: 1.txt
$ mkfolder
Please input the folder NAME:
test_1
$ mkfolder
Please input the folder NAME:
test_2
$ ls
home:
1.txt
test_1
test_2
$ cdfolder
Please input the folder NAME:
test_1
$ ls
test_1:
$
```

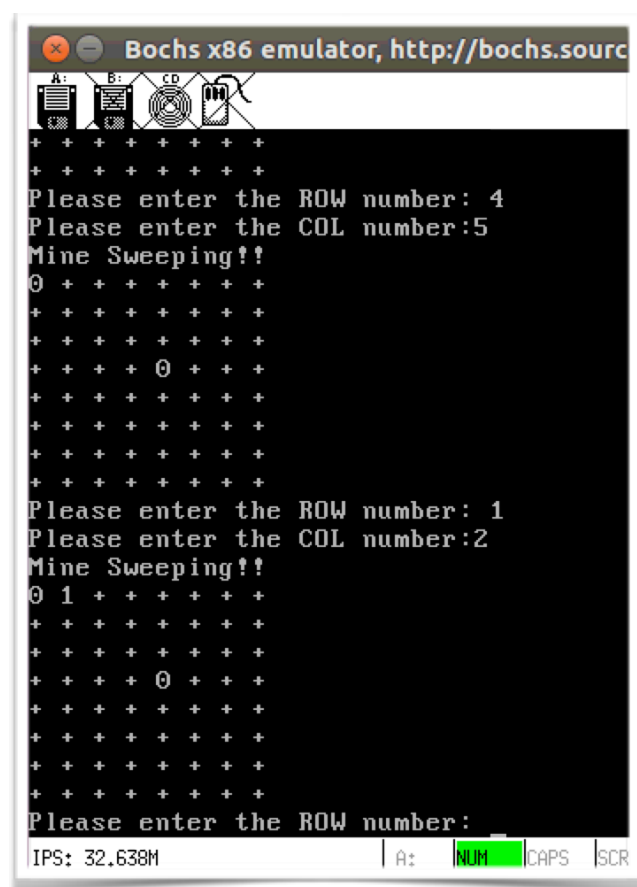
3. help

输入 help 后，输出操作系统控制台的所有可用命令

```
Bochs x86 emulator, http://bochs.sourceforge.net/
USER Copy Paste Snapshot CONFIG Reset suspend Power
$ help
*****
name      |      function
*****
animation |      Play the welcome anime
ls         |      List all files in current folder
help       |      List all commands
mineSweeping |      Start GAME Mine Sweeping
Boxman     |      Start GAME Boxman
getDate    |      Start APP Date Calculator
cdfolder   |      Open a folder
calculator |      Start APP Floating Point Calculator
mkfile     |      Create a new file
mkfolder   |      Create a new folder
catfile    |      Read a file
rmfile     |      Delete a file
editfile   |      Edit file, cover the content
cdup       |      return to the parent folder
*****
$
```

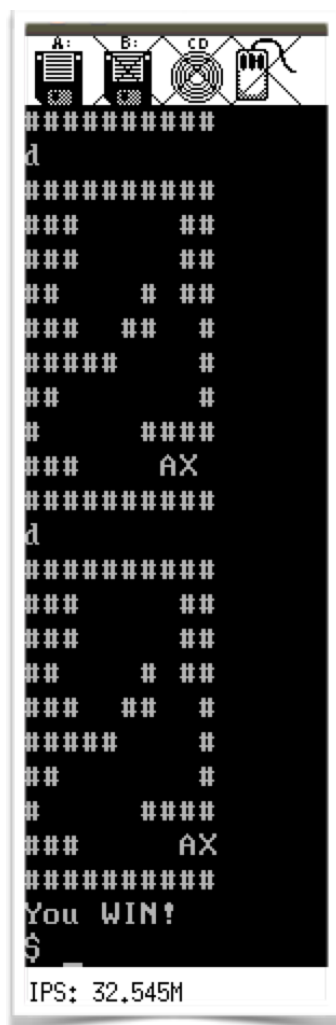
4. mineSweeping

输入 mineSweeping 后，开始扫雷游戏



5. Boxman

输入Boxman 后，开始推箱子游戏



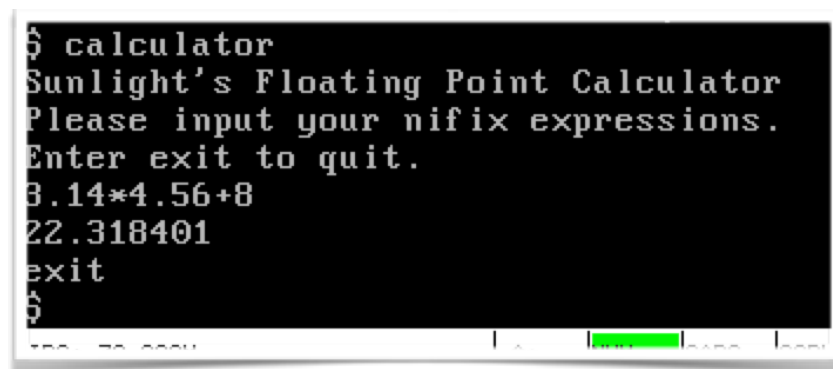
6. getDate

获取天数应用程序：输入年月日，输出这一日期是该年的第几天

A terminal window showing the execution of the 'getDate' command. The prompt is '\$'. The user enters '20180905'. The output is 'It is the 248 DAY of YEAR 2018\$'. The status bar at the bottom shows 'IPS: 31.445M', 'A:', 'NUM' (highlighted in green), 'CAPS', and 'SCRL'.

7. calculator

打开一个支持浮点运算的计算器，输入中缀表达式即可得到相应的计算结果

A terminal window showing the execution of the 'calculator' command. The prompt is '\$'. The user enters 'calculator'. The output is 'Sunlight's Floating Point Calculator', 'Please input your nifix expressions.', 'Enter exit to quit.', '8.14*4.56+8', '22.318401', 'exit', '\$'. The status bar at the bottom shows 'IPS: 31.445M', 'A:', 'NUM' (highlighted in green), 'CAPS', and 'SCRL'.

8. cdfolder

打开文件夹，进入该文件夹的目录下

A terminal window showing the execution of the 'cdfolder' command. The prompt is '\$'. The user enters 'cdfolder'. The output is 'Please input the folder NAME:', 'test', '\$ ls', 'test:', '\$'. The status bar at the bottom shows 'IPS: 31.530M', 'A:', 'NUM' (highlighted in green), 'CAPS', and 'SCRL'.

9. mkfile

在当前目录下新建一个文件

```
$ mkfile
Please input the file NAME:
1.txt
New file created: 1.txt
```

10. mkfolder

新建文件夹，需要输入文件夹名称

```
$ mkfolder
Please input the folder NAME:
test
$ ls
home:
test
$
```

11. catfile

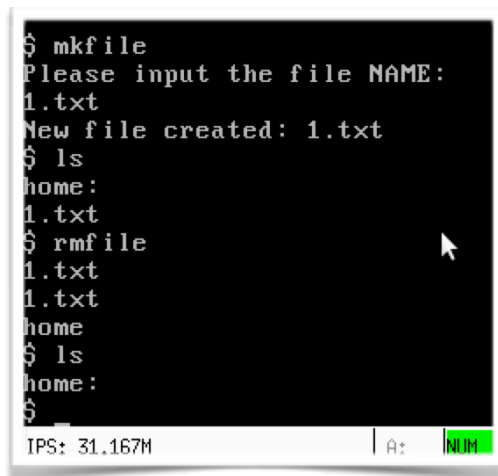
读取一个文件的内容，需要给出文件名

```
$ catfile
Please input the file NAME:
1.txt
File Content: hello
```

12. rmfile

删除文件，需要输入文件名

```
$ mkfile
Please input the file NAME:
1.txt
New file created: 1.txt
$ ls
home:
1.txt
$ rmfile
1.txt
1.txt
home
$ ls
home:
$
```



13. editfile

写文件（覆盖原有内容），需要输入文件名和新的文件内容

```
$ editfile
Please input the file NAME:
1.txt
hello
```



14. cdup

返回上一级文件目录

```
$ ls
test:
$ cdup
$ ls
home:
test
$
```



参考文献

书名	作者	出版社
《Orange S:一个操作系统的实现》	于渊	电子工业出版社
《Linux内核源代码情景分析》	毛德操 / 胡希明	浙江大学出版社
《Linux命令行大全》	William E.Shotts	人民邮电出版社