中山大學
SUN YAT-SEN UNIVERSITY

《操作系统实验》

实验报告

（实验一）

学 院 名 称 ：	数据科学与计算机学院
专业（班级）：	17 级计算机类教务 3 班
学 生 姓 名 ：	姚森舰
学　　号 ：	17341189
时　　间 ：	2019　年　3　月　15　日

一. 实验题目

1. 接管裸机的控制权。

二. 实验目的

安装虚拟机软件VMware，学会创建裸机模拟IMB-PC环境，同时生成一个大小为1.44MB软盘。学习使用汇编语言写一个程序，利用WinHex工具和NASM将程序转换成二进制数据存入生成的软盘第一个扇区中，并使用首扇区作为引导程序使裸机在开机的时候运行。

三. 实验要求

设计IBM PC的一个引导扇区程序，程序功能是:用字符从屏幕左边某行位置45度角下斜射出，保持一个可观察的适当速度直线运动，碰到屏幕的边后产生反射，改变方向运动，如此类推，不断运动；在此基础上，增加你的个性扩展，如同时控制两个运动的轨迹，或炫酷动态变色，个性画面，如此等等，自由不限。还要在屏幕某个区域特别的方式显示你的学号姓名等个人信息。将这个程序的机器码放进放进虚拟软盘的首扇区，并用此软盘引导你的XXXPC，直到成功。

四. 实验方案

实验环境：Windows10 +VMware

实验工具：NASM+Winhex+Notepad++

大概流程：用Notepad++编写汇编程序，通过nasm编译成二进制代码，通过VMware创建带软盘的虚拟机，利用winhex将编译好的二进制代码写到虚拟机的软盘的第一个扇区，然后在Vmware上运行测试虚拟机和程序。

在老师给的代码上加以修改，显示了自己的姓名与一些字符，并修改了弹球的跳动范围。

大概思路：先定义一些字符串组成图案，然后利用循环显示出来。然后写一个类似于C 语言中的 switch 语句，判断该进行左上，左下，右上，右下中的移动，并将移动后即将前进的方向赋值给 rdul 处的数值，用于下次判断。通过判断是否到达 0 行，24 行，0 列，79 列判断是否到达边界并转向。在过程中通过改变行号 x，列号 y 来改变字符位置，通过(80*x+y)算得字符在显存中的位置并显示。

关键部分代码:

```
    mov al,1
      cmp al,byte[rdul]
    jz  DnRt
      mov al,2
      cmp al,byte[rdul]
    jz  UpRt
      mov al,3
      cmp al,byte[rdul]
    jz  UpLt
      mov al,4
      cmp al,byte[rdul]
    jz  DnLt
      jmp 7C00H
```

类似 switch 语句，决定方向

```
DnRt:
    inc word[x]
    inc word[y]
    mov bx,word[x]
    mov ax,25
    sub ax,bx               ;到底了，也就是25行
      jz  dr2ur
    mov bx,word[y]
    mov ax,80
    sub ax,bx               ;到最右边，也就是80列
      jz  dr2dl             ;右下反弹左下
    jmp show
dr2ur:                      ;到底了，也就是25行，最低到24，即23 24 23
      mov word[x],23
      mov byte[rdul],Up_Rt  ;右下反弹右上
      jmp show
dr2dl:
      mov word[y],78        ;右下反弹左下，最右到79，即 78 79 78
      mov byte[rdul],Dn_Lt
      jmp show
```

到边界之后反弹，具体见图中注释

```
mov cx,6                    ;显示6行
mov ax, str0                ;首地址
mov si,ax
mov bp,860                  ;第一个字符在显存中的位置
row1:
    push cx                     ;保护
    mov cx,19                   ;一行的字符数

char1:
    mov ah,4Fh                  ;字符属性
    mov al,[si]                 ;  AL = 显示字符值（默认值为20h=空格符）
    mov word[gs:bp],ax          ;显存
    inc si                      ;字符串偏移量
    add bp,2                    ;一个字符需要2个字节才能在显存中显示
    loop char1
    add bp,122                  ;(80-字符数)*2 换行
    pop cx
    loop row1
```

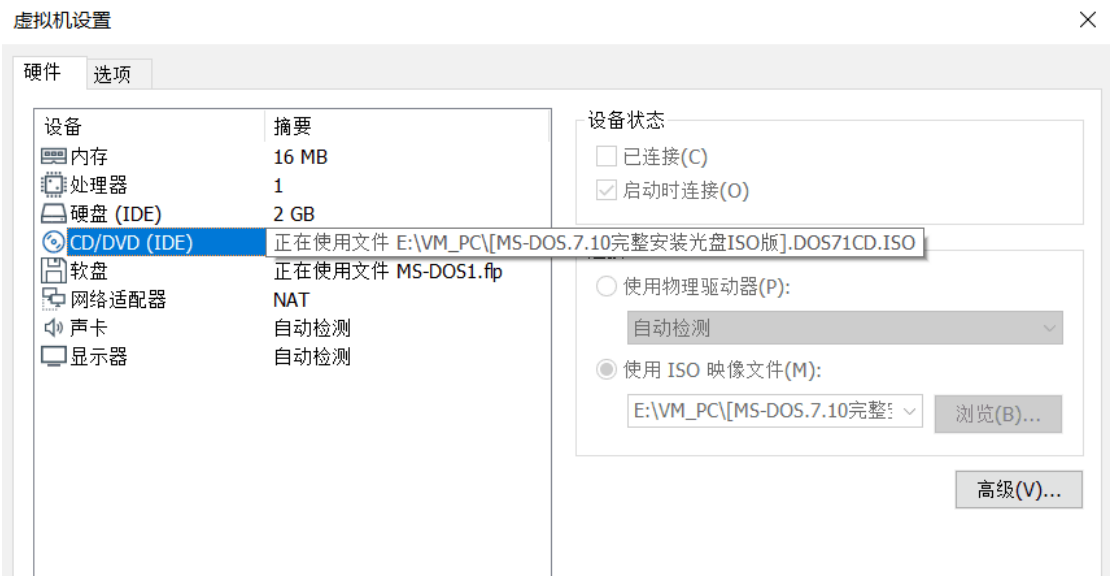
显示姓名等等字符串，具体见注释和代码文件

五. 实验过程与思想

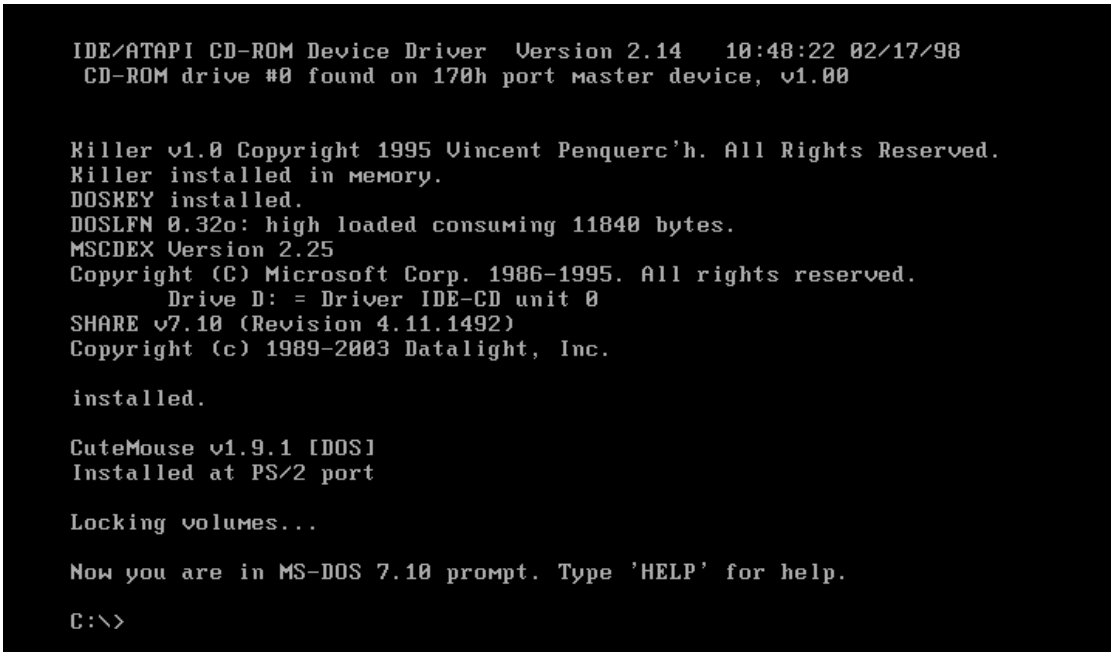
1.安装 VMware 并按要求创建一个无操作系统的裸机。

这个虚拟机就是用来做引导程序的，后面的汇编程序也是在这个虚拟机上进行的。由于开始没注意到要做 DOS 引导盘，所以又另外增加了一个虚拟机完成此部分任务：

下载 MS-DOS 7.0 的 ISO 映像文件，作为 CD/DVD 的映像文件并安装 MS-DOS：



安装完成：



然后跟着参考资料中安装 MS-DOS 7.0 的教程将一个软盘格式化为 DOS 引导盘：

```
System transferred

Volume label (11 characters, ENTER for none)?

    1,457,664 bytes total disk space
      295,936 bytes used by system
    1,161,728 bytes available on disk

          512 bytes in each allocation unit.
        2,269 allocation units available on disk.

Volume Serial Number is 172B-10F5

Format another (Y/N)?y


Insert new diskette for drive A:
and press ENTER when ready...

Checking existing disk format.
Verifying 1.44M
Format complete.
System transferred

Volume label (11 characters, ENTER for none)?
```

完成后软盘部分内容如下：

MS-DOS1.flp

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ANSI ASCII
00000000	EB	3C	90	4D	53	57	49	4E	34	2E	31	00	02	01	01	00	ë< MSWIN4.1
00000010	02	E0	00	40	0B	F0	09	00	12	00	02	00	00	00	00	00	à @ ð
00000020	00	00	00	00	00	00	29	F7	10	66	06	4E	4F	20	4E	41	)÷ f NO NA
00000030	4D	45	20	20	20	20	46	41	54	31	32	20	20	20	33	C9	ME FAT12 3É
00000040	8E	D1	BC	FC	7B	16	07	BD	78	00	C5	76	00	1E	56	16	ŽÑ¼ü{ ½x Åv V
00000050	55	BF	22	05	89	7E	00	89	4E	02	B1	0B	FC	F3	A4	06	U¿" ‰~ ‰N ± üó¤
00000060	1F	BD	00	7C	C6	45	FE	0F	38	4E	24	7D	20	8B	C1	99	½ |ÆEþ 8N$} ‹Á™
00000070	E8	7E	01	83	EB	3A	66	A1	1C	7C	66	3B	07	8A	57	FC	è~ ƒë:f¡ |f; ŠWü
00000080	75	06	80	CA	02	88	56	02	80	C3	10	73	ED	33	C9	FE	u €Ê ˆV €Ã sí3Éþ
00000090	06	D8	7D	8A	46	10	98	F7	66	16	03	46	1C	13	56	1E	Ø}ŠF ˜÷f F V
000000A0	03	46	0E	13	D1	8B	76	11	60	89	46	FC	89	56	FE	B8	F Ñ‹v `‰Fü‰Vþ¸
000000B0	20	00	F7	E6	8B	5E	0B	03	C3	48	F7	F3	01	46	FC	11	÷æ‹^ ÃH÷ó Fü
000000C0	4E	FE	61	BF	00	07	E8	28	01	72	3E	38	2D	74	17	60	Nþa¿ è(r>8-t `
000000D0	B1	0B	BE	D8	7D	F3	A6	61	74	3D	4E	74	09	83	C7	20	± ¾Ø}ó¦at=Nt ƒÇ
000000E0	3B	FB	72	E7	EB	DD	FE	0E	D8	7D	7B	A7	BE	7F	7D	AC	;ûrçëÝþ Ø}{§¾ }¬
000000F0	98	03	F0	AC	98	40	74	0C	48	74	13	B4	0E	BB	07	00	˜ ð¬˜@t Ht ´ »
00000100	CD	10	EB	EF	BE	82	7D	EB	E6	BE	80	7D	EB	E1	CD	16	Í ëï¾‚}ëæ¾€}ëáÍ
00000110	5E	1F	66	8F	04	CD	19	BE	81	7D	8B	7D	1A	8D	45	FE	^ f Í ¾ }‹} Eþ
00000120	8A	4E	0D	F7	E1	03	46	FC	13	56	FE	B1	04	E8	C2	00	ŠN ÷á Fü Vþ± èÂ
00000130	72	D7	EA	00	02	70	00	52	50	06	53	6A	01	6A	10	91	r×ê p RP Sj j '
00000140	8B	46	18	A2	26	05	96	92	33	D2	F7	F6	91	F7	F6	42	‹F ¢& –'3Ò÷ö'÷öB
00000150	87	CA	F7	76	1A	8A	F2	8A	E8	C0	CC	02	0A	CC	B8	01	‡Ê÷v ŠòŠèÀÌ Ì¸
00000160	02	80	7E	02	0E	75	04	B4	42	8B	F4	8A	56	24	CD	13	€~ u ´B‹ôŠV$Í
00000170	61	61	72	0A	40	75	01	42	03	5E	0B	49	75	77	C3	03	aar @u B ^ IuwÃ
00000180	18	01	27	0D	0A	49	6E	76	61	6C	69	64	20	73	79	73	' Invalid sys
00000190	74	65	6D	20	64	69	73	6B	FF	0D	0A	44	69	73	6B	20	tem diskÿ Disk
000001A0	49	2F	4F	20	65	72	72	6F	72	FF	0D	0A	52	65	70	6C	I/O errorÿ Repl
000001B0	61	63	65	20	74	68	65	20	64	69	73	6B	2C	20	61	6E	ace the disk, an
000001C0	64	20	74	68	65	6E	20	70	72	65	73	73	20	61	6E	79	d then press any
000001D0	20	6B	65	79	0D	0A	00	00	49	4F	20	20	20	20	20	20	key IO
000001E0	53	59	53	4D	53	44	4F	53	20	20	20	53	59	53	7F	01	SYSMSDOS SYS
000001F0	00	41	BB	00	07	60	66	6A	00	E9	3B	FF	00	00	55	AA	A» `fj é;ÿ Uª
00000200	F0	FF	FF	03	40	00	05	60	00	07	80	00	09	A0	00	0B	ðÿÿ @ ` €

再新建一个软盘，并利用 Winhex 填满自己的信息：

MYDOSOS.flp | information

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ANSI ASCII
00000000	31	37	33	34	31	31	38	39	20	59	61	6F	53	65	6E	6A	17341189 YaoSenj
00000010	69	61	6E	31	37	33	34	31	31	38	39	20	59	61	6F	53	ian17341189 YaoS
00000020	65	6E	6A	69	61	6E	31	37	33	34	31	31	38	39	20	59	enjian17341189 Y
00000030	61	6F	53	65	6E	6A	69	61	6E	31	37	33	34	31	31	38	aoSenjian1734118
00000040	39	20	59	61	6F	53	65	6E	6A	69	61	6E	31	37	33	34	9 YaoSenjian1734
00000050	31	31	38	39	20	59	61	6F	53	65	6E	6A	69	61	6E	31	1189 YaoSenjian1
00000060	37	33	34	31	31	38	39	20	59	61	6F	53	65	6E	6A	69	7341189 YaoSenji
00000070	61	6E	31	37	33	34	31	31	38	39	20	59	61	6F	53	65	an17341189 YaoSe
00000080	6E	6A	69	61	6E	31	37	33	34	31	31	38	39	20	59	61	njian17341189 Ya
00000090	6F	53	65	6E	6A	69	61	6E	31	37	33	34	31	31	38	39	oSenjian17341189
000000A0	20	59	61	6F	53	65	6E	6A	69	61	6E	31	37	33	34	31	YaoSenjian17341
000000B0	31	38	39	20	59	61	6F	53	65	6E	6A	69	61	6E	31	37	189 YaoSenjian17
000000C0	33	34	31	31	38	39	20	59	61	6F	53	65	6E	6A	69	61	341189 YaoSenjia
000000D0	6E	31	37	33	34	31	31	38	39	20	59	61	6F	53	65	6E	n17341189 YaoSen
000000E0	6A	69	61	6E	31	37	33	34	31	31	38	39	20	59	61	6F	jian17341189 Yao
000000F0	53	65	6E	6A	69	61	6E	31	37	33	34	31	31	38	39	20	Senjian17341189
00000100	59	61	6F	53	65	6E	6A	69	61	6E	31	37	33	34	31	31	YaoSenjian173411
00000110	38	39	20	59	61	6F	53	65	6E	6A	69	61	6E	31	37	33	89 YaoSenjian173
00000120	34	31	31	38	39	20	59	61	6F	53	65	6E	6A	69	61	6E	41189 YaoSenjian
00000130	31	37	33	34	31	31	38	39	20	59	61	6F	53	65	6E	6A	17341189 YaoSenj
00000140	69	61	6E	31	37	33	34	31	31	38	39	20	59	61	6F	53	ian17341189 YaoS
00000150	65	6E	6A	69	61	6E	31	37	33	34	31	31	38	39	20	59	enjian17341189 Y
00000160	61	6F	53	65	6E	6A	69	61	6E	31	37	33	34	31	31	38	aoSenjian1734118
00000170	39	20	59	61	6F	53	65	6E	6A	69	61	6E	31	37	33	34	9 YaoSenjian1734
00000180	31	31	38	39	20	59	61	6F	53	65	6E	6A	69	61	6E	31	1189 YaoSenjian1
00000190	37	33	34	31	31	38	39	20	59	61	6F	53	65	6E	6A	69	7341189 YaoSenji
000001A0	61	6E	31	37	33	34	31	31	38	39	20	59	61	6F	53	65	an17341189 YaoSe
000001B0	6E	6A	69	61	6E	31	37	33	34	31	31	38	39	20	59	61	njian17341189 Ya
000001C0	6F	53	65	6E	6A	69	61	6E	31	37	33	34	31	31	38	39	oSenjian17341189
000001D0	20	59	61	6F	53	65	6E	6A	69	61	6E	31	37	33	34	31	YaoSenjian17341
000001E0	31	38	39	20	59	61	6F	53	65	6E	6A	69	61	6E	31	37	189 YaoSenjian17
000001F0	33	34	31	31	38	39	20	59	61	6F	53	65	6E	6A	55	AA	341189 YaoSenjU ª
00000200	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	

2. 编写汇编程序：

关键代码见实验方案，具体代码见代码文件

3. 编译：

nasm

```
Microsoft Windows [版本 10.0.17134.648]
(c) 2018 Microsoft Corporation。保留所有权利。

C:\Users\user\AppData\Local\bin\NASM>nasm show1.asm

C:\Users\user\AppData\Local\bin\NASM>
```

show1	2019/3/18 22:57	文件	1 KB
show1.asm	2019/3/10 21:51	ASM 文件	5 KB

利用 Winhex 将生成的二进制文件粘贴到所用软盘的第一个扇区，值得注意的是不要改变软盘大小，同时引导程序也不要大于 512B，否则部分程序指令无法加载，使得程序无法正确运行。

show1

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	ANSI ASCII
00000000	31	C0	8C	C8	8E	C0	8E	D8	8E	C0	B8	00	B8	8E	E8	C6	1ÀŒÈŽÀŽØŽÀ¸ ¸ŽèÆ
00000010	06	D7	7D	2A	FF	0E	CE	7D	75	FA	C7	06	CE	7D	88	13	×}*ÿ Î}uúÇ Î}ˆ
00000020	FF	0E	D0	7D	75	EE	C7	06	CE	7D	88	13	C7	06	D0	7D	ÿ Ð}uîÇ Î}ˆ Ç Ð}
00000030	44	02	B0	01	3A	06	D2	7D	74	1E	B0	02	3A	06	D2	7D	D ° : Ò}t ° : Ò}
00000040	74	53	B0	03	3A	06	D2	7D	0F	84	86	00	B0	04	3A	06	tS° : Ò} „† ° :
00000050	D2	7D	0F	84	B8	00	EB	FE	FF	06	D3	7D	FF	06	D5	7D	Ò} „¸ ëþÿ Ó}ÿ Õ}
00000060	8B	1E	D3	7D	B8	19	00	29	D8	74	0E	8B	1E	D5	7D	B8	‹ Ó}¸)Øt ‹ Õ}¸
00000070	50	00	29	D8	74	11	E9	F0	00	C7	06	D3	7D	17	00	C6	P)Øt éð Ç Ó} Æ
00000080	06	D2	7D	02	E9	03	01	C7	06	D5	7D	4E	00	C6	06	D2	Ò} é Ç Õ}N Æ Ò
00000090	7D	04	E9	F5	00	FF	0E	D3	7D	FF	06	D5	7D	8B	1E	D5	} éõ ÿ Ó}ÿ Õ}‹ Õ
000000A0	7D	B8	50	00	29	D8	74	0E	8B	1E	D3	7D	B8	FF	FF	29	},P)Øt ‹ Ó},ÿÿ)
000000B0	D8	74	11	E9	B3	00	C7	06	D5	7D	4E	00	C6	06	D2	7D	Øt é³ Ç Õ}N Æ Ò}
000000C0	03	E9	E7	00	C7	06	D3	7D	01	00	C6	06	D2	7D	01	E9	éç Ç Ó} Æ Ò} é
000000D0	D9	00	FF	0E	D3	7D	FF	0E	D5	7D	8B	1E	D3	7D	B8	FF	Ù ÿ Ó}ÿ Õ}‹ Ó},ÿ
000000E0	FF	29	D8	74	0D	8B	1E	D5	7D	B8	FF	FF	29	D8	74	10	ÿ)Øt ‹ Õ},ÿÿ)Øt
000000F0	EB	77	C7	06	D3	7D	01	00	C6	06	D2	7D	04	E9	AB	00	ëwÇ Ó} Æ Ò} é«
00000100	C7	06	D5	7D	01	00	C6	06	D2	7D	02	E9	9D	00	FF	06	Ç Õ} Æ Ò} é ÿ
00000110	D3	7D	FF	0E	D5	7D	8B	1E	D5	7D	B8	FF	FF	29	D8	74	Ó}ÿ Õ}‹ Õ},ÿÿ)Øt
00000120	0D	8B	1E	D3	7D	B8	19	00	29	D8	74	0F	EB	3B	C7	06	‹ Ó}¸)Øt ë;Ç
00000130	D5	7D	01	00	C6	06	D2	7D	01	EB	70	C7	06	D3	7D	17	Õ} Æ Ò} ëpÇ Ó}
00000140	00	C6	06	D2	7D	03	EB	63	31	C0	A1	D3	7D	BB	50	00	Æ Ò} ëc1À¡Ó}»P
00000150	F7	E3	03	06	D5	7D	BB	02	00	F7	E3	89	C5	B4	4F	A0	÷ã Õ}» ÷ã‰Å´O
00000160	D7	7D	65	89	46	00	E9	AB	FE	31	C0	A1	D3	7D	BB	50	×}e‰F é«þ1À¡Ó}»P
00000170	00	F7	E3	03	06	D5	7D	BB	02	00	F7	E3	89	C5	B4	2F	÷ã Õ}» ÷ã‰Å´/
00000180	A0	D7	7D	65	89	46	00	E9	8A	FE	31	C0	A1	D3	7D	BB	×}e‰F éŠþ1À¡Ó}»
00000190	50	00	F7	E3	03	06	D5	7D	BB	02	00	F7	E3	89	C5	B4	P ÷ã Õ}» ÷ã‰Å´
000001A0	6F	A0	D7	7D	65	89	46	00	E9	69	FE	31	C0	A1	D3	7D	o ×}e‰F éiþ1À¡Ó}
000001B0	BB	50	00	F7	E3	03	06	D5	7D	BB	02	00	F7	E3	89	C5	»P ÷ã Õ}» ÷ã‰Å
000001C0	B4	1F	A0	D7	7D	65	89	46	00	E9	48	FE	EB	FE	88	13	´ ×}e‰F éHþëþˆ
000001D0	44	02	01	07	00	00	00	2A									D *

由于实验一只需要一个扇区，可以直接利用 nasm 生成映像文件，更加方便快捷：

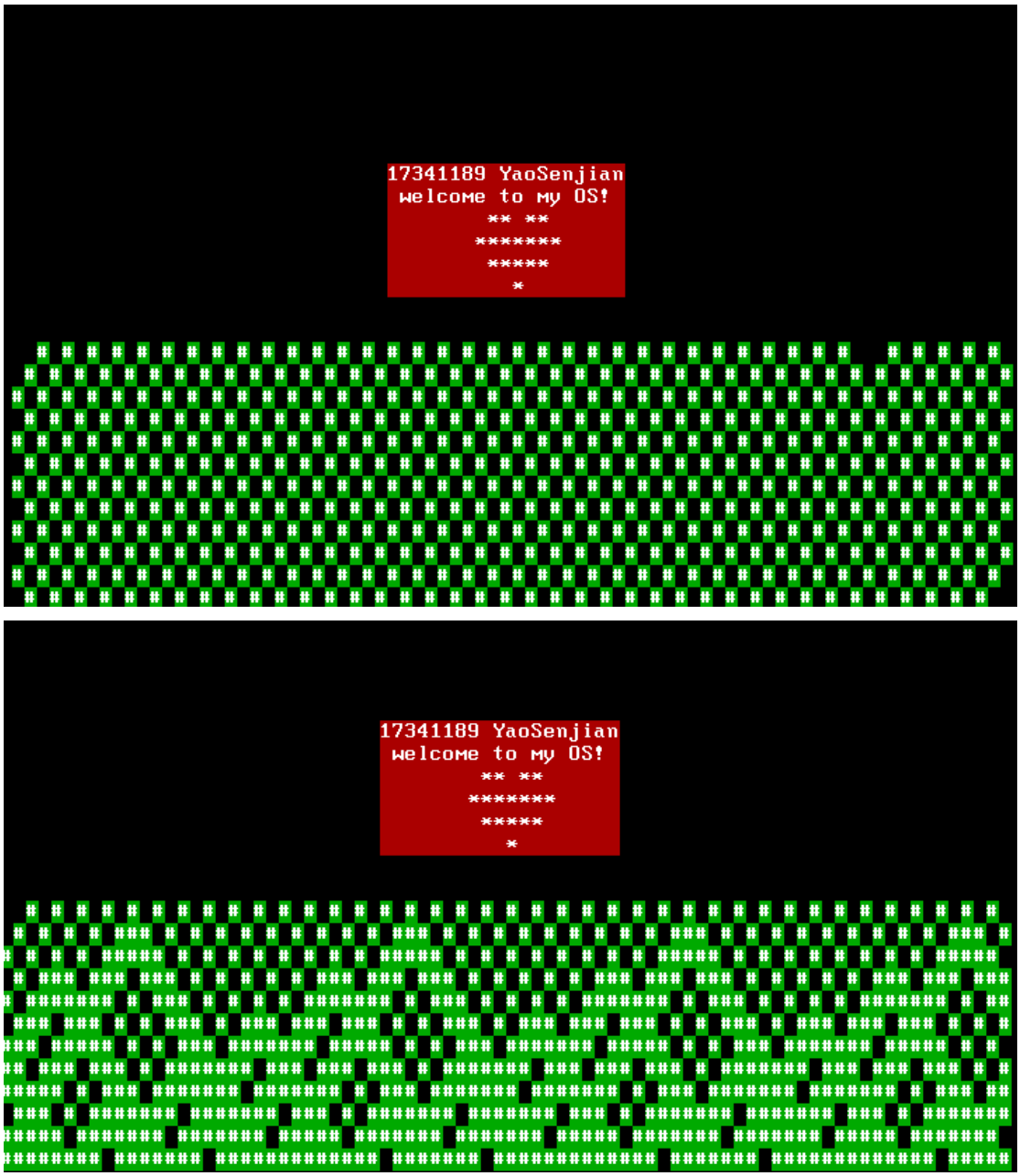
nasm

```
Microsoft Windows [版本 10.0.17134.648]
(c) 2018 Microsoft Corporation。保留所有权利。

C:\Users\user\AppData\Local\bin\NASM>nasm show1.asm

C:\Users\user\AppData\Local\bin\NASM>nasm show1.asm -o show1.img
```

所以提交的时候直接交的这个.img 映像文件，而不是.flp 软盘文件。

4.运行测试：

将软盘的第一个扇区替换为自己编写的汇编指令的二进制代码后，开启虚拟机以虚拟软盘启动，启动后虚拟机就会将我们的程序加载到虚拟机的内存空间中并运行，如下图：



六. 实验心得和总结

通过这个实验，熟悉了安装、创建虚拟机的过程，对引导程序也有了更深的理解。

通过汇编语言，能够在不到 512B 的空间上实现许多功能，深深体会到了汇编语言的高效。之前计算机组成原理课上，我们也写过一些 x86 的汇编，但是是 Masm 格式，和现在的 Nasm 有些许不同，比如数据段，代码段不需要自己声明，稍微方便一些，但也是对我们汇编能力的考验。

这次实验也遇到了很多问题，克服了很多困难。例如：为什么要在代码前加上 org 7c00h？查阅资料后了解到 bios 会自动将引导程序加载到 0000:7c00h 处执行，所以程序要指定偏移量。

还有一些比较坑的地方，比如 x，y 分别指行号和列号，而不是平常说的坐标；显存显示一个字符需要两个字节，一个字节是装字符串的属性，另一个字节是字符串的 ASCII 码，这就容易出现问题。如果不是利用 bios 的中断而是选择自己去将字符串搬到显存显示的话，每次显示一个字符，字符串这边的偏移量要加 1，而显存那边需要加 2，这常常容易忘记，造成显示的结果错误。另外，这样做时还有一个问题就是，我开始用 si 寄存器来放字符偏移量，然后将字符串起始地址放到了 es 寄存器，然后通过[es:si]来访问字符，结果也是错误的，原因是，[es:si]最后访问到的地址是 es*16+si 而不是我想的 es+si，这个错误有点坑，比较难发现，也可能是因为我之前的汇编学得不够扎实。当然，如果调用中断来显示字符串，就不会遇到这些问题了。

关于将一个虚拟软盘用 DOS 格式化为 DOS 引导盘，我是根据后面的参考资料做的，安装步骤中其实有一些是不太理解的，可能需要循序渐进的学习和理解。

此外，因为引导程序大小不能超过 512B，所以有时候不注意超了，也会出现错误，而且如果不知道这个问题的话，根本就不知道错误出在哪。所以理论知识也很重要。

总的来说，实验一也花了很多时间，从一开始不知从何下手，到现在对引导程序有了一个大概的轮廓，也为后面的实验打下了基础，收获了不少经验。

七. 参考资料

[1] 0x7c00解疑

[2] NASM汇编笔记

[3] MS-DOS 6.22的安装

[4] VMware12 中安装MS-DOS 7.10