-
-
- BIOS

- COM
- 1/4 'A' 45 1/4
- 512
-

Mac OS 10.12

Vim 8.0.1400 VS Code 1.21.0

Nasm 2.13.02

Bochs 2.6.9

Git 2.15.1

GNU Make 3.8.1

1

0 0x55 0xAA BIOS 0x7C00 BIOS 0x10 BIOS
0x16 0 AH=0

2

(1) 20h 21h PSP

jmp

call call CS IP ret CS IP

DOS 0xA100 256 0xA000 PSP 0x00 CDh
20h int 20h 0x0A IP CS

DOS DOS ret PSP int 20h 21h 4ch DOS

```
interrupt_20h: ; 20h     21h
    mov ah, 4ch
interrupt_21h:
    cmp ah, 4ch ;     4ch
    jnz panic_21h_func_not_impl ;    kernel panic
    jmp dword[0xA00A] ;   jmp psp
    iret
panic_21h_func_not_impl: ;    4ch
    print_string panic_21h_msg, paini_21h_len, 0, 0
    jmp $
```

DOS

(2)

0 1k 256 N IP CS 0000:[4N], 0000:[4N + 2]

(2)

BIOS 13H 2 dl, ch, dh, cl ah bx

0 N 2N + 1 al 2 bx = 0xA100

pusha jmp

3 1/4

(1)

BIOS 0x16 1 ZF 0 1 0x16 0 ASCII

(2)

NASM macro 4 macro macro 4 x y 4 asm 4

(3)

ESC 27 DOS ret PSP int 20h int 21h 4ch

1

Vn Pn 1

$$P_{n+1} = P_n + V_n (1)$$

80

nasm %rep movsb

2 " " * wasd " " o

BIOS 0x1A 0x11ee div N ah 0 N-

1

2 Makefile

4 Makefile user asm

```makefile
include gmsl                                                    # plus
disk_index = 1                                                  #
SHELL=/bin/bash
AS = nasm
ASFLAG = -f bin
user_src = $(sort $(wildcard user*.asm))          #
user_bin = $(user_src:.asm=.o)                                 #
user%.o : user%.asm common.asm                                 #
        $(AS) $(ASFLAG) $< -o $@
kernal_src = myos1.asm
kernal_bin = $(kernal_src:.asm=.o)
kernal : $(kernal_src)
        $(AS) $(ASFLAG) $(kernal_src) -o $(kernal_bin)
floppyfile = disk.img
clean_disk:                                                     #
        dd if=/dev/zero of=$(floppyfile) bs=512 count=2880
write_kernal: clean_disk kernal                                #
        dd if=$(kernal_bin) of=$(floppyfile) conv=notrunc
define DO_write                                                #
```
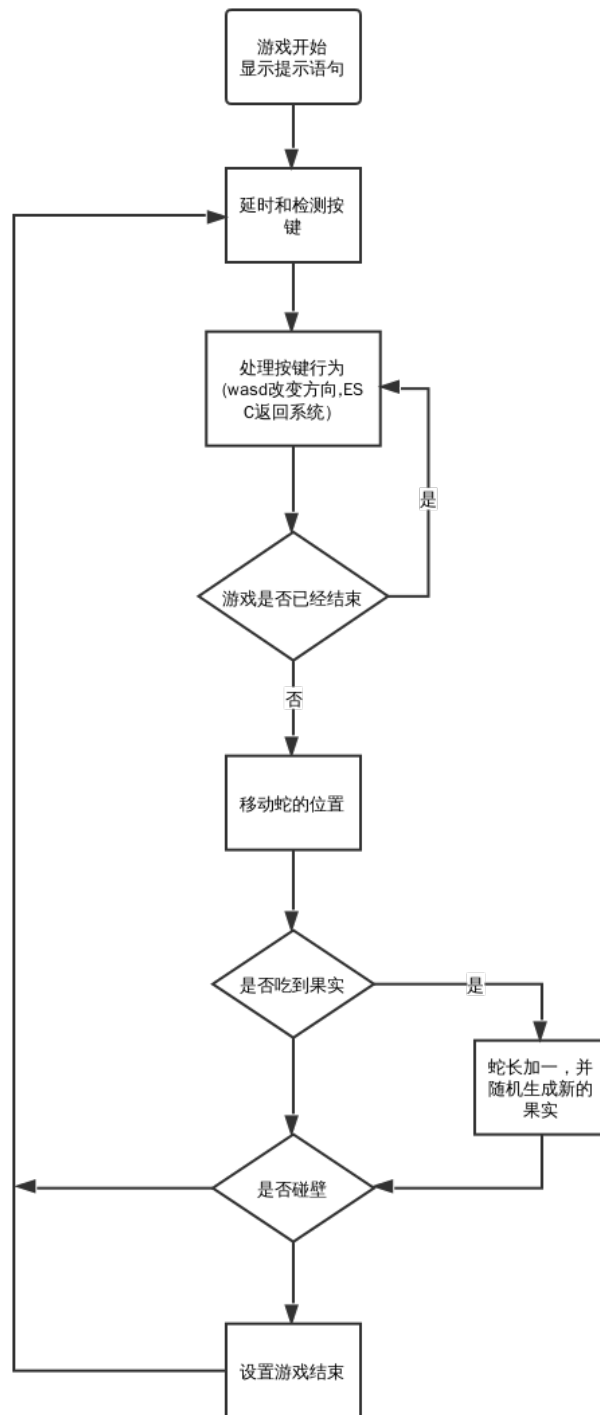
3

游戏开始
显示提示语句

延时和检测按键

处理按键行为
(wasd改变方向,ESC返回系统)

游戏是否已经结束

是

否

移动蛇的位置

是否吃到果实

是

蛇长加一，并随机生成新的果实

是否碰壁

设置游戏结束

Diagram.bb

Figure 1:

```makefile
dd if=$(strip $(1)) of=$(floppyfile) bs=1024 seek=$(strip $(disk_index)) conv=notrunc
$(eval disk_index = $(call plus,$(disk_index),1))

endef
write_all_progs: write_kernal $(user_bin)          # foreach
        $(foreach user_prog, $(user_bin), $(call DO_write, $(user_prog)))
```

VS Code   myos1.asm user1.asm user5.asm   Terminal   make
bochs   bochs

Figure 2:

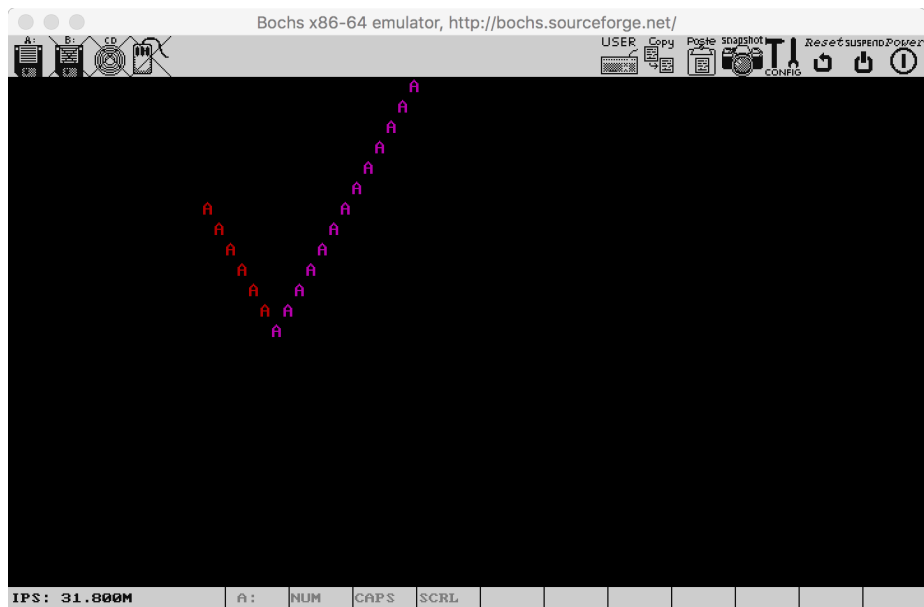VS Code Terminal   c

1 5   ESC
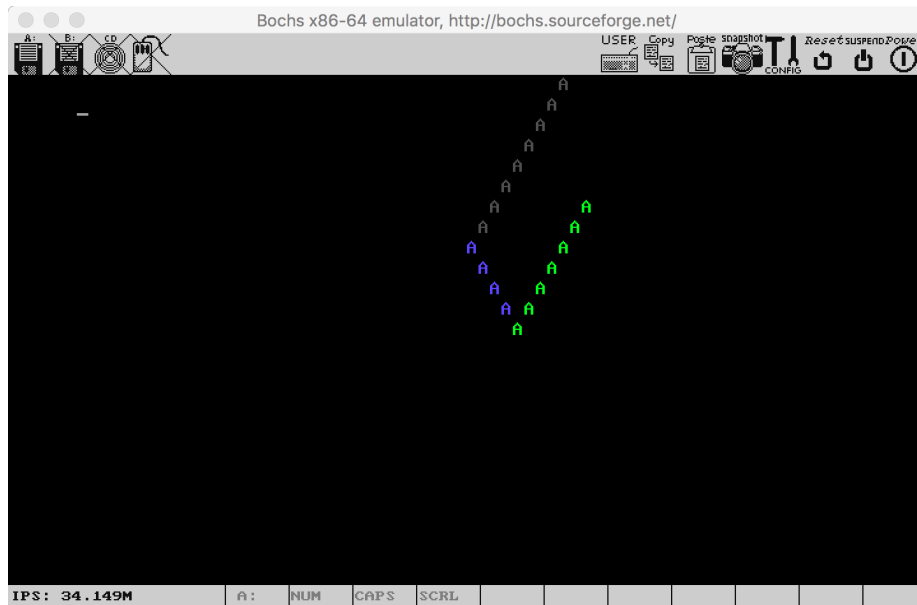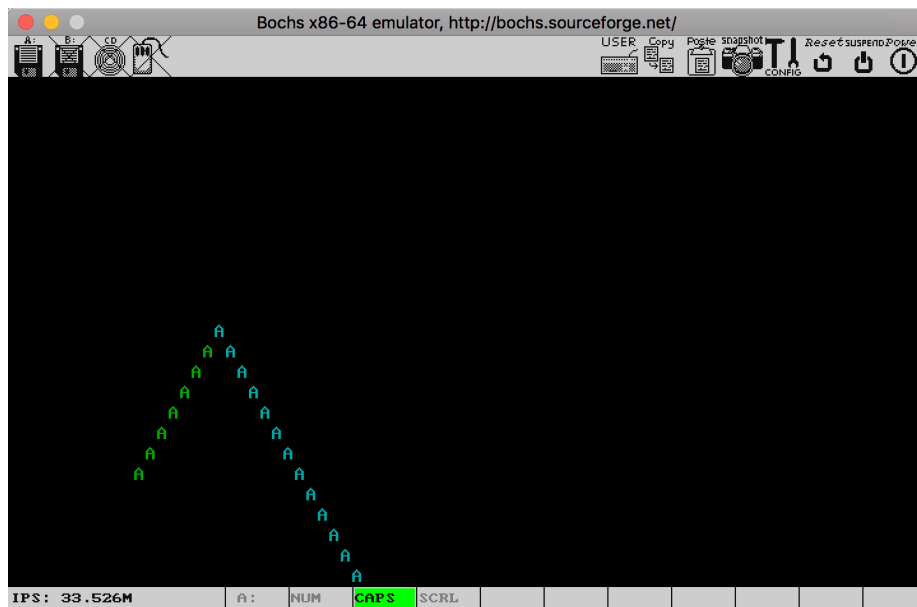
1 4   4   4
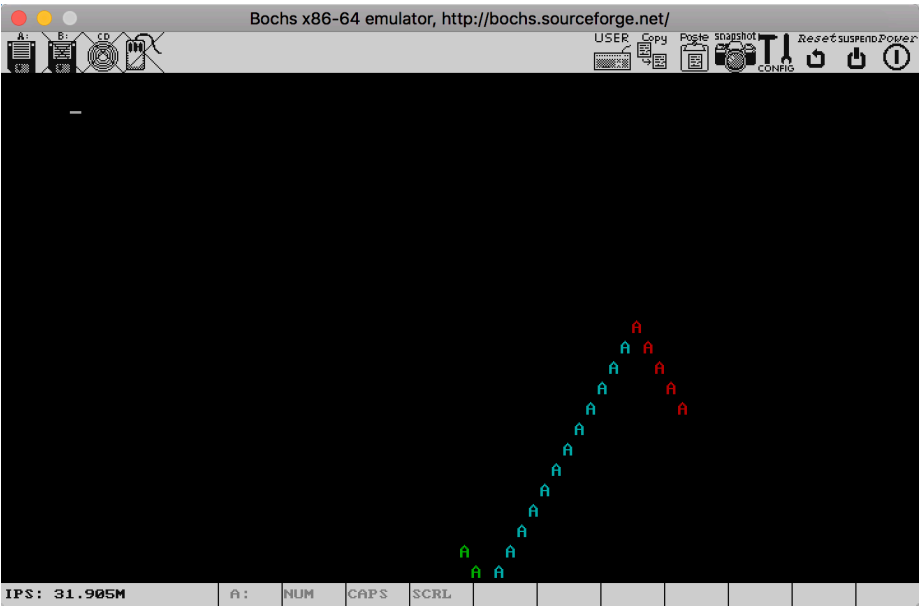
Figure 3:

Figure 4:

Figure 5:

Figure 6:

Figure 7:

screen_record

3

7

200 Nasm   Nasm   Makefile   200 GNU
Make   Makefile

bug

rand   C   "   "

U

[1]. Nasm Documentation, http://www.nasm.us/doc

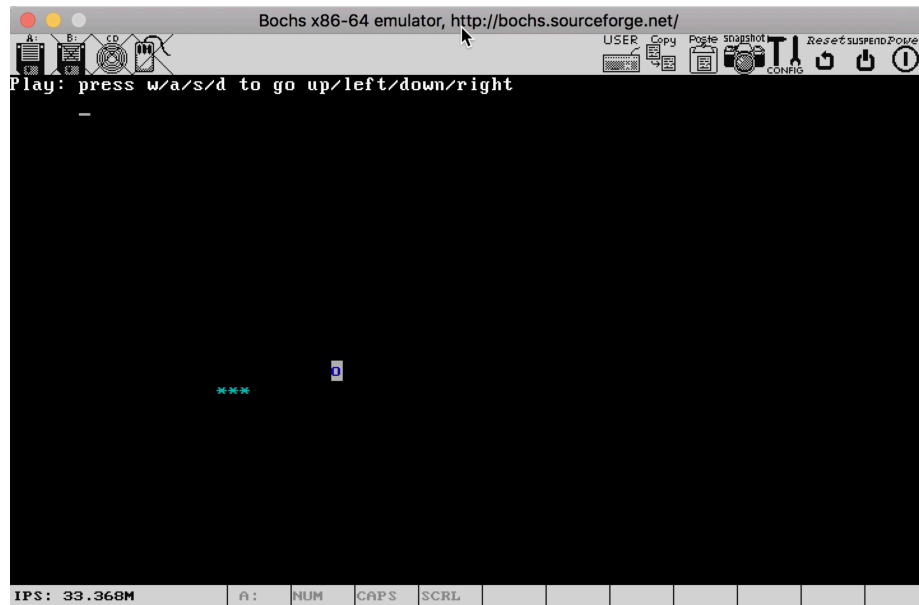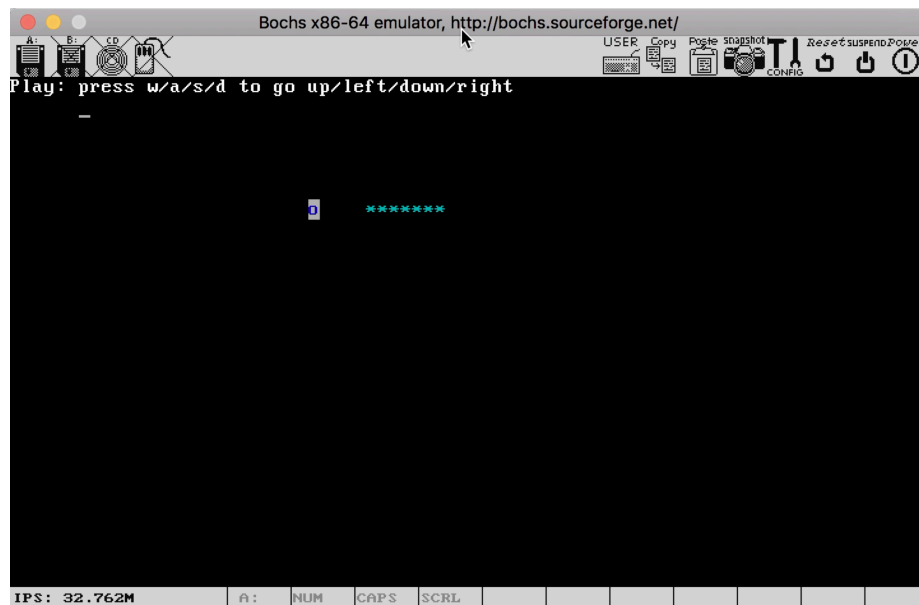[2].   GNU   Make   Documentation,   https://www.gnu.org/software/make/
manual/

Figure 8:

Figure 9:

Figure 10:

[3]. Phoenix BIOS 4.0 User's Manual, http://www.esapcsolutions.com/ecom/drawings/PhoenixBIOS4_rev6UserMan.pdf

[4]. Ascii Table, https://www.asciitable.com/

[5]. PSP - DOS Program Segment Prefix Layout, http://stanislavs.org/helppc/program_segment_prefix.html

[6]. C Calling Convention and the 8086, http://ece425web.groups.et.byu.net/stable/labs/StackFrame.html