

# Forum:[Hack Forums](#)

Thread: [Simple OS coded in NASM \(16-bit, both for QEMU or VBox\)](#)

Infamia - December 21, 2016, 1:46 pm

OS ([Click to View](#))

Code BITS 16

start:

```
mov ax, 07C0h ; SET UP STACK
mov ss, ax
mov sp, 4096
sub ax, 512
mov ds, ax
```

main:

```
xor ah, ah ; MAKE REGISTER AH 0
int 16h ; GET KEYBOARD INPUT (one key)
cmp ah, 50h ; BIOS SCANCODE FOR DOWN ARROW
je down ; GO TO LABEL DOWN
cmp ah, 48h ; COMPARE BIOS SCANCODE OF KEY TO THAT OF UP ARROW
je up ; JUMP TO UP LABEL LATER IN CODE
cmp ah, 4Bh ; AND SO ON
je left
cmp ah, 4Dh
je right
mov ah, 0Eh ; AFTER USERINPUT IS CALLED, ASCII CODE IS PLACED IN AL
int 10h ; CALL INTERRUPT AND PRINT ASCII CODE FROM KEYBOARD
cmp al, 10 ; www .asciitable. com
je newline
cmp al, 13
je newline
cmp al, 8
je backspace
cmp al, 27
je shutdown
cmp al, 96
je sshutdown
jmp main
```

newline:

```
mov al, 13 ; MAKE A CR/LF
int 10h
mov al, 10
int 10h
jmp main
```

down:

```
mov ah, 0Fh ; MOVE POINTER DOWN
int 10h
mov ah, 03h
int 10h
inc dh
```

```

    mov ah, 2
    int 10h
    jmp main
up:
    mov ah, 0Fh ; MOVE POINTER UP
    int 10h
    mov ah, 03h
    int 10h
    dec dh
    mov ah, 2
    int 10h
    jmp main
left:
    mov ah, 0Fh ; MOVE LEFT
    int 10h
    mov ah, 03h
    int 10h
    dec dl ; I COULD'VE JUST DONE A BACKSPACE BUT IT WOULD HAVE DELETED USER
TEXT
    mov ah, 2
    int 10h
    jmp main
right:
    mov ah, 0Fh ; MOVE POINTER RIGHT
    int 10h
    mov ah, 03h
    int 10h
    inc dl
    mov ah, 2
    int 10h
    jmp main
backspace:
    mov al, 0 ; MOVE A NULL WHICH DELETES CURRENT CHARACTER
    int 10h
    mov al, 8 ; THEN MAKE A SPACE SINCE NOW THE CHARACTER BEFORE IT IS SELECTED
    int 10h
    jmp main ; GO BACK TO MAIN
shutdown:
    int 20h ; full shutdown
sshutdown:
    int 19h ; restart

```

```

times 510-($-$$) db 0
dw 0AA55h ; BOOTLOADER SIGNATURE AT END OF SECTOR
times 1474560-($-$$) db 0 ; FILL FILE WITH 1440KB OF DATA SO IT CAN BECOME A VALID
FLOPPY

```

Build File (PYTHON) ([Click to View](#))

```

Code import os, sys
cmd = "nasm -o " + sys.argv[1] + " -f bin " + sys.argv[1]
os.system("cls")
os.system("color 1f")

```

```
while True:
    os.system(cmd)
    print("complete...")
    opt = raw_input("Compile again? ").lower()
    sys.exit() if opt == "no" else True
```

BAT Exec ([Click to View](#))

Code @echo off

nasm -o \_os.flp -f bin \_os.asm

It isn't much, but it could be used for learning- it's written in pure NASM with-out any macros or C functions, so this can be just run under DOSBox with certain commands to build it to be so. But this does require prerequisites for it to work, one- being Python- is optional, and the other- NASM- isn't so optional unless you've found a way of building it manually.

Save the Assembly file as something like \_os.asm (so it is at the top of the directory listing- usually) and the Python file under \_build.py or something of the likes. If you cannot afford to download Python, then simply save the .bat as something like \_build.bat and run it. (I don't know the equivalent for Linux, but it should be the same as calling NASM and just selecting the file)

This is a floppy file, so if you run it in Virtual Box or so; if any additions of code plus over 1440KB, you will encounter an error; thus you'll need to run it in QEMU as as far as I know, Virtual Box doesn't support plain binary formats. So just simply add an instruction in the Python file which calls qemu-system-i386 -fda \_os.bin/flp (thus you'll need to have qemu installed, which isn't that big of a file.

My Github is at: <https://github.com/unazed/OS1> if you want to see older versions (or however Github works) or commit code. Thanks for reading.

Here's a picture of it being run in QEMU

[imgur.com/GKdJcoK.png](http://imgur.com/GKdJcoK.png)

**Malicious Intent** - December 21, 2016, 1:50 pm

Well done man. I hope you will develop this project into something bigger over the time. Good luck

**Infamia** - December 21, 2016, 1:54 pm

(Dec 21, 2016 02:50 PM)michypc Wrote: [\[CITE\]](#) Well done man. I hope you will develop this project into something bigger over the time. Good luck

Thanks, I've commented the code since somebody on Discord suggested that people trying to read it might be baffled. I do plan on adding integration with languages like D or C later on, but I think I should make it a shell first. But nonetheless thanks for your post, my friend said people here would embarrass me or something ;)

**Malicious Intent** - December 21, 2016, 2:04 pm

(Dec 21, 2016 02:54 PM)Infamia Wrote: [\[CITE\]](#) Thanks, I've commented the code since somebody on Discord suggested that people trying to read it might be baffled. I do plan on adding integration with languages like D or C later on, but I think I should make it a shell first. But nonetheless thanks for your post, my friend said people here would embarrass me or something ;)

Lol don't listen to your friend haha.. Yeah shell would be nice for sure. Its nice to see projects with big potential just taking of. As i said i hope you will continue to develop this. Bright future ahead buddy.

**Infamia** - December 21, 2016, 3:59 pm

(Dec 21, 2016 03:04 PM)michypc Wrote: [\[CITE\]](#) Lol don't listen to your friend haha.. Yeah shell would be nice for sure. Its nice to see projects with big potential just taking of. As i said i hope you will continue to develop this. Bright future ahead buddy.

Oh God, I've just fallen into the problem of running out of space in the sector; I've written the shell already but I can't fit the most vital 2 bytes into there which signifies that it's actually bootable and an OS, I guess I'll have to write a bootloader first.

**Malicious Intent** - December 21, 2016, 4:10 pm

(Dec 21, 2016 04:59 PM)Infamia Wrote: [\[CITE\]](#) Oh God, I've just fallen into the problem of running out of space in the sector; I've written the shell already but I can't fit the most vital 2 bytes into there which signifies that it's actually bootable and an OS, I guess I'll have to write a bootloader first.

Lol .. there you go... you will probably run into many problems like this but by the time you done with them you will have nice lil functional OS..

**Infamia** - December 21, 2016, 4:38 pm

(Dec 21, 2016 05:10 PM)michypc Wrote: [\[CITE\]](#) Lol .. there you go... you will probably run into many problems like this but by the time you done with them you will have nice lil functional OS..

True man, I hope the rest of Hackforums is mostly like you, thusfar I've posted a few times in Lounge and I think the people are kind of nice but prone to getting mad at something annoying. I will try to keep the OS in on or two files however since I hate switching back and forth files, with the exception of build files of course.

**Malicious Intent** - December 21, 2016, 4:45 pm

Well ill be following this development..

**Venom0** - February 15, 2017, 2:10 pm

Hey, nice work. I also have some experience with this. Have you planned on adding support for protected mode and memory segmentation? It would allow you to put in your shell. I could help you with that.

**Infamia** - February 15, 2017, 11:29 pm

(Feb 15, 2017 03:10 PM)Venom0 Wrote: [\[CITE\]](#) Hey, nice work. I also have some experience with this. Have you planned on adding support for protected mode and memory segmentation? It would allow you to put in your shell. I could help you with that.  
this idea is quite gone, sorry.

i'm learning more c/++ so i can cross-compile it later with the assembly and make an actual OS rather than a text editor with protected mode and all the other neat shit.

**\_L0cky\_** - May 30, 2017, 3:18 pm

(Feb 16, 2017 12:29 AM)Infamia Wrote: [\[CITE\]](#) this idea is quite gone, sorry.

i'm learning more c/++ so i can cross-compile it later with the assembly and make an actual OS rather than a text editor with protected mode and all the other neat shit.

Such a great post, it's a pity you're no longer into this.