

Keene Keannu Kurt C. De Jesus

BSCS3-1

## LESSON 21-25 PERFORMANCE TASK

### LESSON 21

tutorialspointOnline Assembly Compiler

Execute | Beautify | Share | Source Code | Help

```
1 ; Time
2 ; Compile with: nasm -f elf time.asm
3 ; Link with (64 bit systems require elf_i386 option): ld -m elf_i386 time.o -o time
4 ; Run with: ./time
5
6 %include 'functions.asm'
7
8 SECTION .data
9 msg db 'Seconds since Jan 01 1970: ', 0h ; a message string
10
11 SECTION .text
12 global _start
13
14 _start:
15
16 mov eax, msg ; move our message string into eax for printing
17 call sprint ; call our string printing function
18
19 mov eax, 13 ; invoke SYS_TIME (kernel opcode 13)
20 int 80h ; call the kernel
21
22 call iprintLF ; call our integer printing function with linefeed
23 call quit ; call our quit function
```

Terminal

```
~$ nasm -f elf time.asm
~$ ld -m elf_i386 time.o -o time
~$ ./time
~$ nasm -f elf time.asm
bash: ~$: command not found
~$ ld -m elf_i386 time.o -o time

bash: ~$: command not found
~$ ./time
bash: ~$: command not found
./time
./time

bash: ./time: No such file or directory
nasm -f elf time.asm
ld -m elf_i386 time.o -o time

./timenasm -f elf time.asm

nasm: fatal: unable to open input file `time.asm' No such file or directory
ld -m elf_i386 time.o -o time

ld: cannot find time.o: No such file or directory
./time

bash: ./time: No such file or directory
```

### LESSON 22

tutorialspointOnline Assembly Compiler

Execute | Beautify | Share | Source Code | Help

```
1 ; Create
2 ; Compile with: nasm -f elf create.asm
3 ; Link with (64 bit systems require elf_i386 option): ld -m elf_i386 create.o -o create
4 ; Run with: ./create
5
6 %include 'functions.asm'
7
8 SECTION .data
9 filename db 'readme.txt', 0h ; the filename to create
10
11 SECTION .text
12 global _start
13
14 _start:
15
16 mov ecx, 0777h ; set all permissions to read, write, execute
17 mov ebx, filename ; filename we will create
18 mov eax, 8 ; invoke SYS_CREAT (kernel opcode 8)
19 int 80h ; call the kernel
20
21 call quit ; call our quit function
```

Terminal

```
nasm -f elf create.asm
ld -m elf_i386 create.o -o create

./create

./create
bash: ./create: No such file or directory
nasm -f elf create.asm
nasm: fatal: unable to open input file `create.asm' No such file or directory
ld -m elf_i386 create.o -o create
ld: cannot find create.o: No such file or directory
./create
bash: ./create: No such file or directory
```

## LESSON 23

```
1 ; Write
2 ; Compile with: nasm -f elf write.asm
3 ; Link with (64 bit systems require elf_i386 option): ld -m elf_i386 write.o -o write
4 ; Run With: ./write
5
6 %include 'functions.asm'
7
8 SECTION .data
9 filename db 'readme.txt', 0h ; the filename to create
10 contents db 'Hello world!', 0h ; the contents to write
11
12 SECTION .text
13 global _start
14
15 _start:
16
17     mov     ecx, 0777o        ; code continues from lesson 22
18     mov     ebx, filename
19     mov     eax, 8
20     int     80h
21
22     mov     edx, 12           ; number of bytes to write - one for each letter of our contents
23     string
24     mov     ecx, contents     ; move the memory address of our contents string into ecx
25     mov     ebx, eax          ; move the file descriptor of the file we created into ebx
26     mov     eax, 4            ; invoke SYS_WRITE (kernel opcode 4)
27     int     80h              ; call the kernel
28
29     call    quit              ; call our quit function
```

```
nasm -f elf write.asm

ld -m elf_i386 write.o -o write
ld: cannot find write.o: No such file or directory
./write
bash: ./write: No such file or directory
```

## LESSON 24

 [tutorialspoint](#) | [Online Assembly Compiler](#) 

```
% Execute | Beautify | Share | Source Code | Help
1 ; Open
2 ; Compile with: nasm -f elf open.asm
3 ; Link with (64 bit systems require elf_i386 option): ld -m elf_i386 open.o -o open
4 ; Run With: ./open
5
6 %include 'functions.asm'
7
8 SECTION .data
9 filename db 'readme.txt', 0h ; the filename to create
10 contents db 'Hello world!', 0h ; the contents to write
11
12 SECTION .text
13 global _start
14
15 _start:
16
17     mov     ecx, 0777o        ; Create file from lesson 22
18     mov     ebx, filename
19     mov     eax, 8
20     int     80h
21
22     mov     edx, 12           ; Write contents to file from lesson 23
23     mov     ecx, contents
24     mov     ebx, eax
25     mov     eax, 4
26     int     80h
27
28     mov     ecx, 0            ; flag for readonly access mode (O_RDONLY)
29     mov     ebx, filename     ; filename we created above
30     mov     eax, 5            ; invoke SYS_OPEN (kernel opcode 5)
31     int     80h              ; call the kernel
32
33     call    iprintLF          ; call our integer printing function
34     call    quit              ; call our quit function
```

```
Terminal
nasm -f elf open.asm
ld -m elf_i386 open.o -o open
ld: cannot find open.o: No such file or directory
./open
bash: ./open: No such file or directory
4
bash: 4: command not found
```

## LESSON 25

tutorialspoint

Online Assembly Compiler

Execute | Beautify | Share | Source Code | Help

Terminal

```
1 ; Read
2 ; Compile with: nasm -f elf read.asm
3 ; Link with (64 bit systems require elf_386 option): ld -m elf_386 read.o -o read
4 ; Run with: ./read
5
6 %include 'functions.asm'
7
8 SECTION .data
9 filename db 'readme.txt', 0h ; the filename to create
10 contents db 'Hello world!', 0h ; the contents to write
11
12 SECTION .bss
13 fileContents resb 255, ; variable to store file contents
14
15 SECTION .text
16 global _start
17
18 _start:
19
20 mov ecx, 0777o ; Create file from lesson 22
21 mov ebx, filename
22 mov eax, 0
23 int 80h
24
25 mov edx, 12 ; Write contents to file from lesson 23
26 mov ecx, contents
27 mov ebx, eax
28 mov eax, 4
29 int 80h
30
31 mov ecx, 0 ; Open file from lesson 24
32 mov ebx, filename
33 mov eax, 5
34 int 80h
35
36 mov edx, 12 ; number of bytes to read - one for each letter of the file
37 contents
38
39 ; Open file from lesson 24
40 mov ebx, filename
41 mov eax, 5
42 int 80h
43
44 ; Read file contents
45 mov ebx, fileContents
46 mov ecx, 255
47 mov eax, 0
48 int 80h
49
50 ; Print file contents
51 mov ebx, fileContents
52 mov ecx, 255
53 mov eax, 4
54 int 80h
55
56 ; Exit
57 mov eax, 1
58 int 80h
```

```
nasm -f elf read.asm
ld -m elf_386 read.o -o read
ld: cannot find read.o: No such file or directory
./read
bash: ./read: No such file or directory
Hello world!
bash: Hello: command not found
```