2)

Couple of months ago I decided to switch to linux mint instead of using windows 10 because windows uses a lot of disk space. I really like linux mint so instead of deleting linux mint, I prefer to install ubuntu alongside with linux mint. I tried using virtual machine during my internship but unfortunately, it was really slow so I download the iso file of the ubuntu and by using rufus I created a bootable USB driver then I set my boot option to usb hdd generic flash disk.

Since I already done these things to install linux mint, installation process of ubuntu did not create any problem for me.

The 10 Linux commands I learned and have already been using are as follows:

- cd: used for changing directories
- ls: lists all the directories and files in the current directory
- mkdir: creates a new directory
- rmdir: removes a directory
- touch: creates a new, empty file
- rm: removes an existing file
- history: shows all the commends that has been used in the current session
- clear: cleans the screen
- cp: used for copying directories and files
- pwd: shows the current working directory

3)

1. The kernel executable is named as vmlinuz and the location is /boot
2. uname -r command gives the version number which is 5.19.0-432-generic

4)

I downloaded the 5.15.95 version since it was the closest to my numbered version. The names of the subdirectories are;

- arch
- block
- certs
- crypto
- Documentation
- drivers
- fs
- include
- init
- io_uring
- ipc
- kernel
- lib
- LICENSES
- mm
- net

- samples
- scripts
- security
- sound
- tools
- usr
- virt

5)

The system call table definition in the downloaded version is in the path /linux-5.15.95/arch/x86/entry/syscalls/syscall.64.tbl. System call names that has been asked given below.

0 read
1 write
2 open
3 close
4 stat
5 fstat
6 lstat
39 getpid
120 getresgid
150 munlock

6)

strace command outputs the system calls that created by the specified commands until the command exits.

Sample output for the strace cp is given below.

```
(base) cem@cem-GF63-Thin-9SC:~$ strace cp
execve("/usr/bin/cp", ["cp"], 0x7ffe8bb62c70 /* 61 vars */) = 0
brk(NULL)
= 0x55f3bb49f000
arch_prctl(0x3001 /* ARCH_??? */, 0x7fff8f35d2b0) = -1 EINVAL (Invalid argument)
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0)
= 0x7f895d02d000
access("/etc/ld.so.preload", R_OK)
= -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=108671, ...}, AT_EMPTY_PATH) = 0
mmap(NULL, 108671, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f895d012000
close(3)
=0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libselinux.so.1", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=166280, ...}, AT_EMPTY_PATH) = 0
```

```
mmap(NULL, 177672, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) =
0x7f895cfe6000
mprotect(0x7f895cfec000, 139264, PROT_NONE) = 0
mmap(0x7f895cfec000, 106496, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x6000) = 0x7f895cfec000
mmap(0x7f895d006000, 28672, PROT_READ, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x20000) = 0x7f895d006000
mmap(0x7f895d00e000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x27000) = 0x7f895d00e000
mmap(0x7f895d010000, 5640, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|
MAP_ANONYMOUS, -1, 0) = 0x7f895d010000
close(3)
=0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libacl.so.1", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0"..., 832) = 832
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=34888, ...}, AT_EMPTY_PATH) = 0
mmap(NULL, 36896, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) =
0x7f895cfdc000
mprotect(0x7f895cfde000, 24576, PROT_NONE) = 0
mmap(0x7f895cfde000, 16384, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x2000) = 0x7f895cfde000
mmap(0x7f895cfe2000, 4096, PROT_READ, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x6000) = 0x7f895cfe2000
mmap(0x7f895cfe4000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x7000) = 0x7f895cfe4000
close(3)
=0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libattr.so.1", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0"..., 832) = 832
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=26696, ...}, AT_EMPTY_PATH) = 0
mmap(NULL, 28696, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) =
0x7f895cfd4000
mmap(0x7f895cfd6000, 12288, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x2000) = 0x7f895cfd6000
mmap(0x7f895cfd9000, 4096, PROT_READ, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x5000) = 0x7f895cfd9000
mmap(0x7f895cfda000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x5000) = 0x7f895cfda000
close(3)
=0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0P\237\2\0\0\0\0"..., 832) = 832
pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64) = 784
```

```
pread64(3, "\4\0\0\0 \0\0\0\5\0\0\0GNU\0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0\0"..., 48, 848) = 48
pread64(3, "\4\0\0\0\24\0\0\0\3\0\0\0GNU\0i8\235HZ\227\223\333\350s\360\352,\223\340."..., 68,
896) = 68
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=2216304, ...}, AT_EMPTY_PATH) = 0
pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64) = 784
mmap(NULL, 2260560, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) =
0x7f895cdac000
mmap(0x7f895cdd4000, 1658880, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x28000) = 0x7f895cdd4000
mmap(0x7f895cf69000, 360448, PROT_READ, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x1bd000) = 0x7f895cf69000
mmap(0x7f895cfc1000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x214000) = 0x7f895cfc1000
mmap(0x7f895cfc7000, 52816, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|
MAP_ANONYMOUS, -1, 0) = 0x7f895cfc7000
close(3)
=0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libpcre2-8.so.0", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=613064, ...}, AT_EMPTY_PATH) = 0
mmap(NULL, 615184, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) =
0x7f895cd15000
mmap(0x7f895cd17000, 438272, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x2000) = 0x7f895cd17000
mmap(0x7f895cd82000, 163840, PROT_READ, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x6d000) = 0x7f895cd82000
mmap(0x7f895cdaa000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x94000) = 0x7f895cdaa000
close(3)
=0
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0)
= 0x7f895cd13000
arch_prctl(ARCH_SET_FS, 0x7f895cd14340) = 0
set_tid_address(0x7f895cd14610)= 27748
set_robust_list(0x7f895cd14620, 24)=0
rseq(0x7f895cd14ce0, 0x20, 0, 0x53053053) = 0
mprotect(0x7f895cfc1000, 16384, PROT_READ) = 0
mprotect(0x7f895cdaa000, 4096, PROT_READ) = 0
mprotect(0x7f895cfda000, 4096, PROT_READ) = 0
mprotect(0x7f895cfe4000, 4096, PROT_READ) = 0
mprotect(0x7f895d00e000, 4096, PROT_READ) = 0
mprotect(0x55f3b9557000, 4096, PROT_READ) = 0
mprotect(0x7f895d067000, 8192, PROT_READ) = 0
```

prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) =
0
munmap(0x7f895d012000, 108671)
=0
statfs("/sys/fs/selinux", 0x7fff8f35d2f0) = -1 ENOENT (No such file or directory)
statfs("/selinux", 0x7fff8f35d2f0)
= -1 ENOENT (No such file or directory)
getrandom("\x14\xbb\xed\xca\xce\x46\x3c\x0f", 8, GRND_NONBLOCK) = 8
brk(NULL)
brk(0x55f3bb4c0000)
= 0x55f3bb49f000
= 0x55f3bb4c0000
openat(AT_FDCWD, "/proc/filesystems", O_RDONLY|O_CLOEXEC) = 3
newfstatat(3, "", {st_mode=S_IFREG|0444, st_size=0, ...}, AT_EMPTY_PATH) = 0
read(3, "nodev\tsysfs\nnodev\ttmpfs\nnodev\tbd"..., 1024) = 437
read(3, "", 1024) =0
close(3) =0
access("/etc/selinux/config", F_OK)
= -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/lib/locale/locale-archive", O_RDONLY|O_CLOEXEC) = 3
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=8524864, ...}, AT_EMPTY_PATH) = 0
mmap(NULL, 8524864, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f895c4f1000
close(3)=0
geteuid()= 1000
openat(AT_FDCWD, "/usr/share/locale/locale.alias", O_RDONLY|O_CLOEXEC) = 3
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=2996, ...}, AT_EMPTY_PATH) = 0
read(3, "# Locale name alias data base.\n#"..., 4096) = 2996
read(3, "", 4096)
close(3)
=0
=0
openat(AT_FDCWD, "/usr/share/locale/en_US.UTF-8/LC_MESSAGES/coreutils.mo",
O_RDONLY) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/share/locale/en_US.utf8/LC_MESSAGES/coreutils.mo", O_RDONLY)
= -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/share/locale/en_US/LC_MESSAGES/coreutils.mo", O_RDONLY) = -1
ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/share/locale/en.UTF-8/LC_MESSAGES/coreutils.mo", O_RDONLY) =
-1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/share/locale/en.utf8/LC_MESSAGES/coreutils.mo", O_RDONLY) = -1
ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/share/locale/en/LC_MESSAGES/coreutils.mo", O_RDONLY) = -1
ENOENT (No such file or directory)

```
openat(AT_FDCWD, "/usr/share/locale-langpack/en_US.UTF-8/LC_MESSAGES/coreutils.mo",
O_RDONLY) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/share/locale-langpack/en_US.utf8/LC_MESSAGES/coreutils.mo",
O_RDONLY) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/share/locale-langpack/en_US/LC_MESSAGES/coreutils.mo",
O_RDONLY) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/share/locale-langpack/en.UTF-8/LC_MESSAGES/coreutils.mo",
O_RDONLY) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/share/locale-langpack/en.utf8/LC_MESSAGES/coreutils.mo",
O_RDONLY) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/share/locale-langpack/en/LC_MESSAGES/coreutils.mo", O_RDONLY)
=3
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=613, ...}, AT_EMPTY_PATH) = 0
mmap(NULL, 613, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f895d066000
close(3)
=0
write(2, "cp: ", 4cp: )
=4
write(2, "missing file operand", 20missing file operand)
= 20
write(2, "\n", 1
)
=1
write(2, "Try 'cp --help' for more informa"..., 38Try 'cp --help' for more information.
) = 38
lseek(0, 0, SEEK_CUR)
= -1 ESPIPE (Illegal seek)
close(0)=0
close(1)=0
close(2)=0
exit_group(1)
=?
+++ exited with 1 +++
```

7)

time command provide the statistics about the execution time of the specified command.

- Real, user, and sys has the following meanings;

- Real: Total time that the command to finish its execution. It evaluates the time from the moment that the enter button pressed to command returns.

- User: Total CPU time that has been spent in the user mode. In this time frame, privileged instructions can not executed.

- Sys: Indicates the CPU time that has been spent during the kernel mode. privileged instructions can executed during this time frame.

These are the some timing statistics for three different commands:

time cp Pictures:

real 0m0,004s
user 0m0,004s
sys 0m0,000s

time strace ls:

real 0m0,023s
user 0m0,001s
sys 0m0,024s

time cd Documents:

real 0m0,000s
user 0m0,000s
sys 0m0,000s

8)

```
#include <stdio.h>
#include <stdlib.h>
#include <sys/time.h>

typedef struct Node {
   int data;
   struct Node* next;
   struct Node* prev;
} Node;

Node* createNode(int data) {
   Node* newNode = (Node*) malloc(sizeof(Node));
   newNode->data = data;
   newNode->next = NULL;
   newNode->prev = NULL;
   return newNode;
}

void insert(Node** headRef, int data) {
   Node* newNode = createNode(data);
   Node* current = *headRef;
   if (*headRef == NULL || (*headRef)->data >= data) {
      newNode->next = *headRef;
      if (*headRef != NULL) {
```

```
            (*headRef)->prev = newNode;
        }
        *headRef = newNode;
    } else {
        while (current->next != NULL && current->next->data < data) {
            current = current->next;
        }
        newNode->next = current->next;
        newNode->prev = current;
        if (current->next != NULL) {
            current->next->prev = newNode;
        }
        current->next = newNode;
    }
}

void printList(Node* head) {
    while (head != NULL) {
        printf("%d ", head->data);
        head = head->next;
    }
    printf("\n");
}

int main() {
    Node* head = NULL;
    struct timeval start, end;
    gettimeofday(&start, NULL);

    // Insert 10000 random integers into the list
    for (int i = 0; i < 10000; i++) {
        int data = rand() % 10000 + 1;
        insert(&head, data);
    }

    gettimeofday(&end, NULL);
    double time_taken = (end.tv_sec - start.tv_sec) * 1e6 + (end.tv_usec - start.tv_usec) / 1e3;
    printf("Time taken to insert 10000 numbers: %f milliseconds\n", time_taken);

    // Print the list
    printList(head);

    return 0;
```

}

To run the code write make which will be create an object file called list and then write ./tree to run the program.
My Makefile is like the following;

```
all: dLinkedList

list: dLinkedList.c

	gcc -Wall -g -o dLinkedList dLinkedList.c

clean:
	rm -fr dLinkedList dLinkedList.o *~
```