**Objective 1: Implementing System call using write () and read() method.**

1. **Using write () system call:**

**A screenshot of a computer

Description automatically generated**

**Output:**

**A screenshot of a computer

Description automatically generated**

1. **Using read() system call:**

**A screenshot of a computer

Description automatically generated with medium confidence**

**Output:**

**A screenshot of a computer

Description automatically generated with medium confidence**

**Objective 2: Process management in Linux.**

1. **Using top command for managing Linux processes:**

**A screenshot of a computer

Description automatically generated**

**Terminologies used:**

**PID**: Unique Process ID given to each process.

**User**: Username of the process owner.

**PR**: Priority given to a process while scheduling.

**NI**: ‘nice’ value of a process.

**VIRT**: Amount of virtual memory used by a process.

**RES**: Amount of physical memory used by a process.

**SHR**: Amount of memory shared with other processes.

**S**: state of the process

‘**D’** = uninterruptible sleep

‘**R’** = running

‘**S’** = sleeping

‘**T’** = traced or stopped

‘**Z’** = zombie

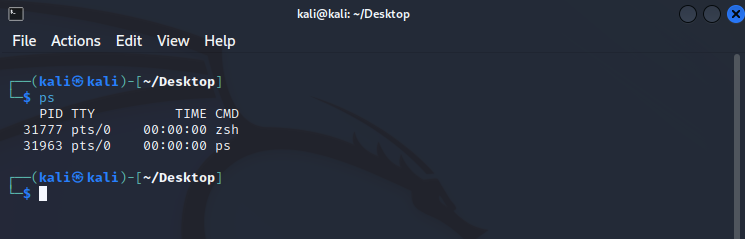
**%CPU:** Percentage of CPU used by the process.

**%MEM**; Percentage of RAM used by the process.

**TIME+:** Total CPU time consumed by the process.

**Command**: Command used to activate the process.

1. **Using the ‘ps’ command:**

****

To get more info we use ‘ps -u’:

**A screenshot of a computer

Description automatically generated**

1. **Stop a process using Kill command:**

**A screenshot of a computer

Description automatically generated**

The syntax for killing a process is:

**$ kill [pid]**

Copy

Alternatively you can also use :

**$ kill -9 [pid]**

Copy

1. **Changing the priority of a process:**

In Linux, you can prioritize between processes. The priority value for a process is called the ‘Niceness’ value.

Niceness value can range from -20 to 19. 0 is the default value.

The fourth column in the output of top command is the column for niceness value.

**A screenshot of a computer program

Description automatically generated with medium confidence**

To start a process and give it a nice value other than the default one, use:

**$ nice -n [value] [process name]**

Copy

To change nice value of a process that is already running use:

**renice [value] -p 'PID'**

**Objective 3: Create a process using fork() method:**

**Program:**

**A screenshot of a computer

Description automatically generated**

**Output:**

**A screenshot of a computer

Description automatically generated**

Fork command works on basis of 2n. Since there are 2 fork() in program, the output is printed 4 times.

**Objective 4: Thread simulation using inbuild function:**

**Program 1: C program to demonstrate use of pthread basic functions**

**A screenshot of a computer screen

Description automatically generated with medium confidence**

**Output:**

**A screenshot of a computer

Description automatically generated**

**Program 2: C program to show multiple threads with global and static variables.**

A screenshot of a computer screen

Description automatically generated with medium confidence

**Output:**

A screenshot of a computer

Description automatically generated

**Conclusion:**

Hence, using kali Linux, the objectives of the lab were successfully completed.