Experiment 09

Aim: To study and implement the read(), write(), and fork() system calls in Unix/Linux operating systems.

System Calls in Unix/Linux

A system call is a direct interface between a user program and the operating system kernel. It allows programs to request services such as file I/O, process control, and inter-process communication.

In this experiment, we focus on the following three fundamental system calls:

1. **write() – For low-level output operations.**
2. **read() – For low-level input operations.**
3. **fork() – For process creation.**

write() System Call

The write() system call is used to **output data** to a file descriptor, such as the standard output (screen).

```
localhost:~# vi writesc.c
#include <stdio.h>
#include <unistd.h>

int main() {
    int count;
    count = write(1, "hello\n", 6);
    printf("Total bytes written: %d\n", count);
    return 0;
}

localhost:~# gcc writesc.c -o writesc
localhost:~# ./writesc
hello
Total bytes written: 6
```

read() System Call

The read() system call is used to **read data** from an input file descriptor, such as the keyboard (standard input).

```
localhost:~# vi readsc.c
#include <stdio.h>
#include <unistd.h>

int main() {
    int nread;
    char buff[20];

    // Read 10 bytes from standard input
    nread = read(0, buff, 10);

    // Write the read bytes to standard output
    write(1, buff, 10);

    return 0;
}
```

6. WAP to display system information

```
localhost:~/tejaswini# vi system.sh
#!/bin/bash
echo "Date"
date
echo "Uptime"
uptime
echo "Memory usage"
free -m
echo "Network usage"
ip a
~
localhost:~/tejaswini# chmod 711 system.sh
localhost:~/tejaswini# ./system.sh
Date
Mon Mar 10 18:00:56 UTC 2025
Uptime
 18:00:57 up 5 min,  load average: 0.00, 0.00, 0.00
Memory usage
               total        used        free      shared  buff/cache   available
Mem:             119           4         113           0           1         112
Swap:              0           0           0
Network usage
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN qlen 1000
    link/ether 02:b4:01:49:4e:45 brd ff:ff:ff:ff:ff:ff
```

7. WAP to find and replace text in a string

```
localhost:~/tejaswini# vi findreplace.sh
#!/bin/bash
first="I drive BMW and Volvo"
second="Audi"
echo "${first/BMW/$second}"
~
localhost:~/tejaswini# chmod u+rwx findreplace.sh
localhost:~/tejaswini# ./findreplace.sh
I drive Audi and Volvo
```

Sreelekha 24/3/25