**Bharatiya Vidya Bhavan's**
# Sardar Patel Institute of Technology
Bhavan's Campus, Munshi Nagar, Andheri (West), Mumbai-400058-India
(Autonomous College Affiliated to University of Mumbai)

## Information Technology Department

**Academic Year:** 2021-2022

**Class:** S.Y.B.Tech **Sem.:** IV **Course:** IT206 Operating Systems

**Name:** Amal Thundiyil

**UID:** 2020400066

**Class:** IT (Batch-D)

**Experiment No.:** 9

**Title:** File Handling

**Aim:** Write a program to prevent destructive update of files by locking as follows: Suppose the inode contains a new permission setting such that it allows only one process at a time to open the file for writing, but many processes can open the file for reading

**Code:**

reader.txt

```
#include <fcntl.h>
#include <pthread.h>
#include <stdio.h>
#include <sys/file.h>
#include <unistd.h>

int main()
   FILE *fp;
   fp = fopen("msg.txt", "r");
   flock(fileno(fp), LOCK_SH);
   do {
      char c = fgetc(fp);
      if (feof(fp)) {
         break;
```

```
        }
        sleep(1);
        printf("%c", c);
        fflush(stdout);
    } while (1);
    flock(fileno(fp), LOCK_UN);
    printf("\n");
    return 0;
}
```

writer.txt

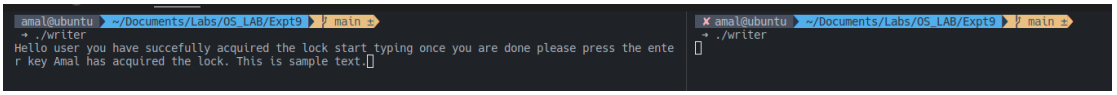
```
#include <fcntl.h>
#include <pthread.h>
#include <stdio.h>
#include <sys/file.h>
#include <unistd.h>

int main() {
    FILE *fp;
    fp = fopen("msg.txt", "a+");
    flock(fileno(fp), LOCK_EX);
    char buff[1000];
    printf("Hello user you have succefully acquired the lock start typing once you are done please press the enter key");
    scanf("%[^\n]s", buff);
    printf("%s", buff);
    fprintf(fp, "%s\n", buff);
    fclose(fp);
```

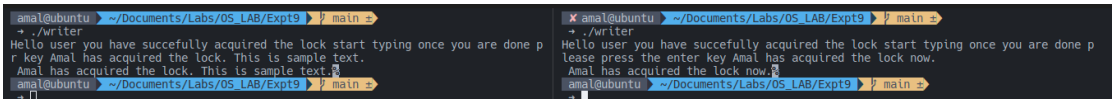
```
    return 0;
}
```

**Output:**

Writer locked case (Exclusive lock)




Writer lock released case (Exclusive lock)




Reader case (Shared lock)