classmate
Date
Page

LAB-09

Arpana M Ramaswamy
1BM18CS164
17/12/20

Q2) Perform reverse reading of file using File lseek API.

```
#include<stdlib.h>
#include<sys/stat.h>
#include<unistd.h>
#include<fcntl.h>

int main(int argc, char *argv[])
{
  char buf;
  int size, fd;

  fd = open(argv[1], O_RDONLY);
  size = lseek(fd, -1, SEEK_END);

  while(size-- >= 0)
  {
    read(fd, &buf, 1);
    write(STDOUT_FILENO, &buf, 1);
    lseek(fd, -2, SEEK_CUR);
  }
  return 0; }
```

classmate

Date/
Page

Ariana MR
Ariana

Q2) Write a C program to create new process using Fork operator

```
#includ<stdio.h>
# include< sys/types.h>
# include< stdlib.h>
#include< unistd.h>

int main()
{
  int pid;
  printf("Before Fork\n");
  pid=fork();
  if(pid>0){
      sleep(3);
      printf("Parent--PID %d\t PPID %d\t
           CHILD PID:%d\n", getpid, getppid(),
            pid);
  }
  else if(pid==0)
    print("CHILD--PID:%d\t PPID:%d\n",
     getpid(), getppid());
  else{
      printf("Error\n");
      exit(1);
  }
  return 0; }
```

classmate
Date
Page

Arpana MR
Arpana

Q7 Write a C program to emulate the ls command using stat and lstat API functions

```
int main (int argc, char *argv[])
{
  struct stat statbuf;

  if (lstat (argv[1], &statbuf) == -1) {
      printf ("couldn't state file");
      exit(0);
  }
  printf ("File is : %s \n", argv[1]);
  printf ("Inode Number: %d \n", statbuf.st_ino);
  printf ("UID = %d", statbuf.st_uid);
  printf ("GID = %d", statbuf.st_gid);
  printf ("Type and Permission : %o \n", statbuf.st_mode);
  printf ("No. of links : %d \n", statbuf.st_nlink);
  printf ("Size in Bytes : %d \n", statbuf.st_size);
  printf ("Blocks allocated : %d \n", statbuf.st_blocks);
  return 0;
}
```