

# Operating Systems - Lab Assignment 4

**Deadline: Dec. 9, 2023**

## Question 1

Write a program that implements the following disk-scheduling algorithms: FCFS, SSTF, SCAN, C-SCAN, Look and C-Look. Your program will service a disk with 5000 cylinders numbered 0 to 4999. The program will generate a random series of 1000 cylinder requests and service them according to each of the algorithms listed above. The program will be passed the initial position of the disk head (as a parameter on the command line) and report the total amount of head movement required by each algorithm.

## Question 2

Create `file1.txt` and put some random text in it. Next, obtain the inode number of this file with the command:

```
ls -li file1.txt
```

This will produce output similar to the following:

```
16980 -rw-r--r-- 2 os os 22 Sep 14 16:13 file1.txt
```

The `ln` command creates a link between a source and target file. This command works as follows:

```
ln [-s] <source file> <target file>
```

UNIX provides two types of links: (i) *hard links* and (ii) *soft links*. A hard link creates a separate target file that has the same inode as the source file. Enter the following command to create a hard link between `file1.txt` and `file2.txt`:

```
ln file1.txt file2.txt
```

What are the inode values of `file1.txt` and `file2.txt`? Are they the same or different? Do the two files have the same - or different - contents?

Next, edit `file2.txt` and change its contents. After you have done so, examine the contents of `file1.txt`. Are the contents of `file1.txt` and `file2.txt` the same or different?

Next, enter the following command which removes `file1.txt`:

```
rm file1.txt
```

Does `file2.txt` still exist as well? Afterwards, remove `file2.txt` by entering the command:

```
strace rm file2.txt
```

The `strace` command traces the execution of system calls as the command `rm file2.txt` is run. What system call is used for removing `file2.txt`?

A soft link (or symbolic link) creates a new file that points to the name of the file it is linking to. In the source code available with this text, create a soft link to `file3.txt` by entering the following command:

```
ln -s file3.txt file4.txt
```

After you have done so, obtain the inode numbers of `file3.txt` and `file4.txt` using the command

```
ls -li file*.txt
```

Are the inodes the same, or is each unique? Next, edit the contents of `file4.txt`. Have the contents of `file3.txt` been altered as well? Last, delete `file3.txt`. After you have done so, explain what happens when you attempt to edit `file4.txt`.