**CSC-3044 Operating Systems & System Programming**

**Laboratory 10 Report**

**NAME:** Joseph Leskey

1. **Document the result of this experiment by copying the exact input that was entered in each terminal window to your lab report document.**  
     
   **[Terminal Window A]**  
   ubuntu@csc3044:~/lab10$ cat - >> file.txt  
   A 1  
   A 2  
   A 3  
   A 4  
   A 5  
   A 6  
   A 7  
     
   **[Terminal Window B]**  
   ubuntu@csc3044:~/lab10$ cat - >> file.txt  
   B 1  
   B 2  
   B 3  
   B 4  
   B 5  
   B 6  
   B 7  
     
   **[File Contents]**  
   ubuntu@csc3044:~/lab10$ cat file.txt  
   A 1  
   B 1  
   A 2  
   B 2  
   A 3  
   B 3  
   A 4  
   B 4  
   A 5  
   B 5  
   A 6

B 6

A 7

B 7

1. **Copy the output that is returned after performing each of the 4 operations shown in Figure 55-4.**  
     
   PID=5739> w 0 10 10

Doing fcntl with fd 3, cmd 7

[PID=5739] got lock

PID=5724> w 0 50 10

Doing fcntl with fd 3, cmd 7

[PID=5724] got lock  
  
PID=5739> w 0 50 10

Doing fcntl with fd 3, cmd 7  
  
PID=5724> w 0 10 10

Doing fcntl with fd 3, cmd 7

[PID=5724] failed (deadlock)

1. **Copy this output to your report document.**  
     
   ubuntu@csc3044:~$ cat /proc/locks

1: FLOCK ADVISORY WRITE 682 00:1b:1116 0 EOF

2: POSIX ADVISORY WRITE 350 00:1b:515 0 EOF

3: POSIX ADVISORY WRITE 5724 08:01:306082 50 59

3: -> POSIX ADVISORY WRITE 5739 08:01:306082 50 59

4: FLOCK ADVISORY WRITE 716 08:01:87428 0 EOF

5: POSIX ADVISORY WRITE 5739 08:01:306082 10 19

1. **Copy the output from each of the 3 processes and explain the results.**  
     
   PID=5781> o 0 tests/file1

PID=5781> w 0 0 10

Doing fcntl with fd 3, cmd 7

[PID=5781] got lock  
  
PID=5775> o 0 tests/file1

PID=5775> w 0 10 10

Doing fcntl with fd 3, cmd 7

[PID=5775] got lock

PID=5775> o 1 tests/file2

PID=5775> w 1 0 10

Doing fcntl with fd 4, cmd 7

[PID=5775] got lock  
  
PID=5776> o 0 tests/file1

PID=5776> w 0 20 10

Doing fcntl with fd 3, cmd 7

[PID=5776] got lock

PID=5776> o 1 tests/file3

PID=5776> w 1 0 10

Doing fcntl with fd 4, cmd 7

[PID=5776] got lock  
  
Locks on different portions of the same file and definitely locks on different files shouldn’t interfere with each other, hence the above output. All operations succeeded without incident.

1. **Do the results match the expected linear behavior?**  
     
   They absolutely did. Each increase of the index by 10,000 resulted in approximately 200 additional milliseconds.
2. **Copy these results with your explanation into your report document.**  
     
   joseph@LJ-L23 lab10 % ./out/program1 tests/lock\_target.txt

Set 40001 locks successfully. Time to sleep.  
  
  
joseph@LJ-L23 lab10 % time ./out/program2 0 tests/lock\_target.txt

Attempted 10000 locks. 0 were successful.

./out/program2 0 tests/lock\_target.txt 0.01s user 0.01s system 83% cpu 0.022 total

joseph@LJ-L23 lab10 % time ./out/program2 10000 tests/lock\_target.txt

Attempted 10000 locks. 0 were successful.

./out/program2 10000 tests/lock\_target.txt 0.00s user 0.22s system 98% cpu 0.228 total

joseph@LJ-L23 lab10 % time ./out/program2 20000 tests/lock\_target.txt

Attempted 10000 locks. 0 were successful.

./out/program2 20000 tests/lock\_target.txt 0.00s user 0.42s system 99% cpu 0.422 total

joseph@LJ-L23 lab10 % time ./out/program2 30000 tests/lock\_target.txt

Attempted 10000 locks. 0 were successful.

./out/program2 30000 tests/lock\_target.txt 0.00s user 0.61s system 99% cpu 0.616 total

joseph@LJ-L23 lab10 % time ./out/program2 40000 tests/lock\_target.txt

Attempted 10000 locks. 0 were successful.

./out/program2 40000 tests/lock\_target.txt 0.00s user 0.80s system 99% cpu 0.806 total  
  
  
As seen here, the execution time of the program grows linearly with the index. This implies increasing convolution in index lookup as the index grows and indicates that lock position is a significant factor in lookup time.