**Operating Systems**

**File I/O with system calls – part 2**

In this lab, you will experiment with some of the system calls introduced last week, in particular: open, close, read, write and lseek. Each person should submit individually, at the end of the lab, regardless of whether the task is finished or not.

Start with your solution from Friday. Approach the problem by writing the steps of your planned solution in comments before beginning coding them.

The program, call it replace, must take 3 arguments: a file name, a file offset, and a string. It should write the argument string over the chars already in the file argument filename at the argument offset.

replace filename offset string

Your program must contain the following function to perform the replacement, calling it with n being the full length of the string parameter:

int replace\_at( int fd, off\_t offset, char \*s, int n );

This function will replace n chars from the buffer s into the file fd at the given offset, overwriting what is currently there. It should return the new offset (just past the replaced chars), or -1 if there is an error.

However, you should not call this function unless there are enough bytes at the offset in the file to be replaced – that is, do not write before the beginning or past the original end of the file.

Do not use stdio functions, except you may use printf for printing messages. Do not make any assumptions about the contents of the file.