CS39002: Operating Systems Lab
Spring 2017
Assignment 1
Due: January 16, 2017

1. Write a C program **search.c** that simulates linear search on a file using multiple processes as follows. Consider a file *f* storing positive integers. The program first reads a filename *f* (a string with no whitespace characters) from the keyboard and then reads all integers from *f* to an array *A*. It then does the following in a loop:
   - Read an integer *k* from the keyboard.
   - If $k > 0$, search for *k* in *A*, and print an appropriate message (found or not found). Else terminate the program.

   To do the search faster, the program adopts the following scheme. If the number of integers is less than or equal to 10, it just linear searches the array. Else, it breaks the array into two halves, creates two child processes, and gives each child process the task of searching in one half of the array. The child process will return the found or not found status through the exit() call (exit(0) for not found, exit(1) for found). The parent process waits for the results from the two child processes, merges them appropriately, and passes it to his parent through its exit status (if it is not the first process) or prints an appropriate message if it is the first process. This process is done recursively until the array to be searched is small enough (≤ 10).

   Write this program without using any exec* calls. The found or not found message should be printed exactly once. You should ensure that when one iteration of the loop is completed (one search done), all child processes created by your program have terminated. Submit a single file *search.c*.

2. Write a C program that takes a file name as a command line parameter and sorts a set of integers stored in the file (use any sorting method). You can assume that the file will always be there in the current directory and that it will always contain a set of integers (maximum no. of integers is 1000). The sorted output is written to the display and the input file is left unchanged. Compile the C file into an executable named "*sort1*". Name the C file *sort1.c*.

   Now write a C program (*xsort.c*) that implements a command called "xsort" that you will invoke from the shell prompt. The syntax of the command is "xsort <filename>". When you type the command, the command opens a new xterm window, and then sorts the integers stored in the file <filename> using the program *sort1*. Look up the man pages for *xterm*, *fork* and the different variations of *exec** calls (such as *execv*, *execvp* etc.) to do this assignment. Submit the files *sort1.c* and *xsort.c*.