

CS69003 ; Computing Systems Lab Assignment 2

Due: August 10

1. [External Sorting] : Sometimes we need to sort huge amounts of data kept in a file. The number of data items may be so large that all of them cannot fit in memory at the same time, so conventional sorting techniques like quicksort, mergesort etc. cannot be applied. External sorting techniques deals with the problem of sorting in such scenarios.

In this assignment, you will first read up external sorting (Fundamentals of Algorithms Vol. 3 : Sorting and Searching by Donald Knuth, or simply from the net). You will then write a C program that will take two filenames *inpdata* and *outdata*, as command line arguments. The file *inpdata* will contain an arbitrary number of integers separated by one or more spaces and/or newline ('\n'). The program will then sort the integers and write them to the file specified in *outfile*. However, you cannot use any array of size more than 1000 integers in your program (you can use other non-array variables as you wish). Also, you can use any number of intermediate files you want, but you must delete them before the program terminates (look up the reference for the standard C library function for a C function that can delete files). Hence, you must use some external sorting technique.

Name your C file *ext_sort.c*.

2. Consider a file containing student records. The record for each student is kept in one line in the file and contains the following, seperated by one or more spaces: First_name (ascii string, one word with no spaces in between), Last_Name (ascii string, one word with no spaces in between), Roll_No (integer), CGPA (float). Roll no. is unique for a student, but others may be the same for two students. You have to write C program that will take the filename as a command line argument. It will then read the contents of the file in memory in appropriate data structures. The program should allow one to search for the following:
 - a. If the first name and the last name are entered, prints out the roll no. and CGPA of all students with those first and last name.
 - b. If a roll no. is entered, prints out the first name, last name and CGPA of the student with that roll no.
 - c. If a CGPA is entered, prints out the names, roll no.s and CGPA of all students with CGPA greater than or equal to the entered CGPA.

Thus, the program should give a menu with 4 options: options {a}, (b), and {c} above, and an option to exit. The program should loop in the menu until the exit option is chosen, at which point the program exits.

Think what data structures you will use to store the records in memory so that searching can be done fast. You can assume that there is a maximum of 1000 student records in the file, and no addition/deletions are done.

Name your C file *search.c*.

Create a single tar file names *<Your roll no>_Assgn2.tar* (for ex. *07CS1002_Assgn2.tar*) containing both the .c files, and submit it to the email address on the course website.