

National Institute of Technology, Calicut
Department of Computer Science and Engineering

CS2094 - Data Structures Lab

Assignment #2 (Advanced)

Date of Submission: On or before 24-02-2015(Tuesday) 5:00pm

Naming Conventions for submission

Submit a single ZIP (.zip) file (do not submit in any other archived formats like .rar or .tar.gz). The name of this file must be ASSG2A<Number>_<ROLLNO>_<FIRSTNAME>.zip (For example: ASSG2A_BxyyyyyCS_LAXMAN.zip). DO NOT add any other files (like temporary files, input files, etc.) except your source code, into the zip archive.

The source codes must be named as ASSG2A<Number>_<ROLLNO>_<FIRSTNAME>_<PROGRAM-NUMBER>.<extension> (For example: ASSG2A_BxyyyyyCS_LAXMAN_1.c).

Questions

1. Consider how you would sort a file that was too large to fit in your machines RAM. External Sorting refers to the sorting of data which is not in the primary memory of the computer. In this problem, you have to implement a miniature version of external sort. The file sizes have been intentionally kept small because of time constraints. The methodology followed has to be the same as for really big files.
 - i) First generate a 256 MB file Bigfile containing integers. You can make use of a random number generator, with suitable modifications on the generator output, to generate suitable integers.
 - ii) Then sort the above file in the following manner.
 - a) Bring in 1024*1024 integers into memory at a time, and sort into an output file, thus generating 64 sorted files of integers.
 - b) Do an 8 way merge on the contents of these files to generate 8 sorted files.
 - c) Do a 4-way merge on the above files to generate 2 sorted files
 - d) Merge the two files obtained above into Bigfile, to obtain the sorted Bigfile.
2. Implement an algorithm to find the median of n integers in $O(n)$ time in all cases and with $\theta(1)$ space complexity.(Reference: Cormen chapter 9)
Input: n integers
Output: Median of the n integers.

Sample Input

7

712

-45
456765
0
-8907
4566
3456

Output
712

Sample Input
6
23
5
2
4
1
3

Output
3.5