

Lab 3 – 2nd Feb 2017

Topics –Exploring maximum size of dynamic array, Sorting

1. Measuring the maximum size of dynamically allocated array

In this exercise you need to find the maximum “allocatable” size of a one dimensional array of type integer on your system. You can iteratively allocate (and free) a dynamic array while doubling the size every time until you incur failure.

2. Merge Sort using insert operation

Implement two variations of merge sort algorithm: one with the (classical) merge operation and the second where merge is replaced by a procedure named mergeByInsert. mergeByInsert(L1, L2) inserts each element of one list, say L2, into the other i.e. L1, one at a time in sorted order. Not that this Insert operation should be in place, i.e. it should use no more than one additional memory location.

Measure the time taken and space usage for each variation by the techniques implemented in the labs last week.

Here is your sample test case. The first line of the input file denotes the number of elements in the file. Your program must simply print the sorted array at the end of execution as shown in the sample output.

Test case 1:

<u>Input</u>	<u>Output</u>
20	1406092
15582942	1581446
11295391	2102834
18505892	4548031
15007002	7916084
10612863	9507662
12892263	10364260
10364260	10612863
21087559	10795924
9507662	11295391
2102834	12892263
14081033	12929586
12929586	14081033
14309798	14309798
1406092	14810325
4548031	15007002
10795924	15582942
14810325	17008852
7916084	18505892
1581446	21087559
17008852	

3. Sorting very large files

Given a very large file of few GBs of data (few billions of records stored in it). You need to sort this file. We came to know from the above exercise, the maximum size you can allocate to an array. So obviously you can't make an array of size of a few billions. However, you can merge two sorted files of smaller size and make a larger sorted file using FILE I/O.

Use this technique to design a sorting algorithm to sort very large files: Use merge sort at the top level to create multiple files of smaller sizes. Sort these smaller files using insertion sort and then merge all of them (two at a time) in a hierarchical manner to get the bigger sorted file. You can assume that the number of records in the file is a multiple of 2.

Also, measure the time taken and space used by the techniques referred to in the last lab session.