

COSC 311 Programming Project #4 Merge Sort**Distributed:** 4/6/2010**Due:** 4/16/2010 (Friday)**Precis:** *Demonstrate $O(n \log n)$ performance for external merge sort.***Background**

A database stores data in tables that are, essentially, 1D arrays of records. Each record contains information about a specific entity; each record will typically have several fields.

Assignment***Demonstrate merge sort algorithm is $O(n \log n)$.***

In order to convince your reader (me), you will need at least 20 data points for different values of n . The execution time for each join should range between 1 second and 600 seconds (approximately).

Generate the data files to conform to the following:

Each record contains two fields. The first field (4 bytes) contains uniformly distributed integers between the values of 0 and 9999. The second field contains anything at all, size 252 bytes (e.g., 126 chars). The total length of each record is 256 bytes.

Sorting the files

The files must be sorted with an external merge sort. The maximum number of records of each file that you can have in main memory at one time is one (i.e., bucket size is 1).

The experiment

Generate 20 different data file with varying sizes of n (number of data elements). Measure the time it takes to sort the data. There has to be at least seven different values for n .

Caveat

While describing the external merge sort is simple, coding it is a bit tricky. Don't delay getting a start on this project.

Turn in

- Hard-copy of code
- Table and graph of timing results.
- One paragraph text describing how your results support (or do not support) the hypothesis that this algorithm is $O(n \log n)$.