# Lab H

**Goals-**

Compare the runtime performance of iterative and recursive versions of an algorithm

1. Create data files.

You will need a large text file filled with random integer values. You can write a small program that prompts for a file name, opens that file and fills it with the specified number of digits. You should prompt the user as you may find you need to create a file with a different number of digits in it.

2. Research.

You will need to find algorithms or code for each of the following:

Recursive implementation of mergesort
Iterative implementation of mergesort
Simple code to time the execution of a program

You can use existing code, but make certain you cite the source! For the timing code you need something simple, a few lines of code. You will find technical analysis of different implementations trying to avoid this or that problem. You don't need that for this lab. You need simple code that gets the start and stop times and calculates the difference. Nothing more.

3. Collect the data.

Mergesort is an algorithm. It shouldn't matter if it's implemented with a loop or a recursive call. Or does it matter? What do you think? Run your program sing both implementations. Collect the measured run time. It's best to do it many times for each size of input.

4. Analysis.

Write up your analysis in a PDF file. What did you observe? Was it what you expected? Why or why not? Where the 2 versions different? Why or why not?

## Grading

Programming style- 1 point

Create the necessary input file(s)- 1 points

Implement and time the iterative algorithm- 3 points

Implement and time the recursive algorithm - 3 points

Analysis- 2 points