

Assignment overview

In this lab you will be implementing THREE decrease-and-conquer and divide-and-conquer algorithms:

1. Find the position of the max element of an array
2. MergeSort
3. Determine if an array has an element where $A[i]=i$

This lab requires Java programming. You may (and should!) discuss the lab and coding techniques with your classmates, but all of the code you submit to Learning Hub must be your own.

Submission information

Due date: As shown on Learning Hub. Late assignments will not be graded.

What to submit:

- Java source file(s)
- Screen prints of your output for all three problems

Note: You may choose to complete all of this exercise in a single Java file, or you may use separate Java files for each of the three problems. Whichever way you do it, make sure you satisfy the output requirements for all three problems.

Please do not zip or compress your submissions. Learning Hub allows you to upload multiple files.

Grading

This lab is graded on a 10-point scale. Point breakdown:

- Divide-and-conquer “position of max” [3 points]
- MergeSort [3 points]
- Decrease-and-conquer “ $A[i]=i$ ” [4 points]

Assignment details

Input

There are no required data files for these problems.

Each of these problems requires your program to perform calculations on an array. Please declare and initialize suitable arrays of integers directly within your program code.

Test your algorithms with a variety of input arrays, including:

- Arrays of both odd and even length
- Arrays whose length is 0, 1, and 2
- (Problem 1) Arrays with the largest number at the start/end and in between; arrays with more than one largest number
- (Problem 2) Arrays that are already sorted

- (Problem 3) Arrays that do/don't have an answer to be found

Problem 1

Design and implement a **divide-and-conquer** algorithm for finding the *position* of the largest number in an array of n numbers. For example, if the input array is $A[0..8] = [2, 5, 8, 3, 6, 9, 1, 6, 5]$, then your algorithm should return 5.

Your program should output:

1. The initial contents of the array
2. The position of the maximum element found

Problem 2

Consider the MergeSort algorithm, which is a **divide-and-conquer** sorting algorithm. The pseudocode for MergeSort is given in the textbook and the lecture notes. Your task is to implement this algorithm.

Your program should output:

1. The initial contents of the array
2. The sorted array after MergeSort

Problem 3

Given a sorted array of *distinct* integers $A[0..n-1]$, design and implement a **decrease-and-conquer** algorithm that will determine whether there exists an index i for which $A[i]=i$.

Your program should output:

This is defined as 'Fixed point'.

1. The initial contents of the array
2. The results of the algorithm that looks for such an index

Your code may assume that the array has distinct integers. (Since you are creating your own input arrays, you are responsible to yourself to make sure your input is correct in this way!)