

CS380 Algorithm Design & Analysis

Assignment 1: Insertion and Merge Sort

Date Assigned: Friday, February 4, 2011

Date Due: Friday, February 11, 2011

Total Points: 25pts

For this assignment you are to implement both insertion sort and merge sort in C++ using the Visual Studio project (SortProject) that I have placed in the CS380 public folder on Turing.

The SortProject contains four files:

- **main.cpp:** uses the BubbleSortArray to create an array, insert elements in the array, and sort the elements in the array.
- **BubbleSortArray.h:** Defines the header file for the class BubbleSortArray, which inherits from the class IArray.
- **BubbleSortArray.cpp:** Contains the implementation of the constructor and sort function in the BubbleSortArray class.
- **IArray.h:** Defines an abstract class for a dynamically sized array. I have hidden the implementation and just provided you with the interface containing the function headers.

You are to add two classes to this project:

- InsertionSortArray: implements the insertion sort algorithm
- MergeSortArray: implements the merge sort algorithm

You can add as many functions as you like to the classes, in fact the more functions the better. Also remember to follow the coding standards that are linked from the website.

Your program should output the following:

- 30 random numbers before insertion sort, then again after insertion sort.
- 30 random numbers before merge sort, then again after merge sort.

What to Submit

- Submit an electronic copy of your project by 11:45am on the day that it is due. Name your project "01PUNET-SortProjec", replacing PUNET with your PU Net ID (i.e. khoj0332), and place it in the CS380-01 Drop folder on Turing.
- Submit a text file ("01PUNET-Answers.txt") to CS380-01 Drop on Turing answering the following questions:
 - How many hours did you work on your project?
 - In your own words, describe how bubble sort works. A paragraph in English will suffice.
 - Give an example showing how bubble sort works on a set of numbers < 10. Show all the steps.
 - What is the running time of bubble sort? Give both the best case and the worst case.