

## Lab 4: Sorting Vectors With Comparators

**DUE: Monday, 6 Mar, 10:00am**

### Lab Preparation

---

Nothing to turn in for Wednesday. Catch up on the readings in the book in preparation for next week's exam.

### Lab Description

---

Do the laboratory at the end of Chapter 5.

- In step 2, you are to write a sort method. The text says you may use any sort that you like, but you should use either merge sort or quicksort for full credit.
- Be sure to test your vector thoroughly before going on to part 3 of the lab assignment. You can do this by creating a class that constructs a `MyVector` object and initializes it with a few data fields, then sorts them, then prints them out.
- For part 3 of the lab procedure, write two applications. They should each work on a different data file and each should perform two or three “interesting” sort processes. Both will use the same `MyVector` class, but each application will call its own class for encapsulating the data objects you are working with, and a number of Comparators to sort the data in different ways. For each application, you will have the application class with just a main method (which does all the I/O), a data object class (i.e. `Student`) and several comparators. Both applications will use the same `MyVector` class.
- You may choose from the data files I have provided in the “labs/comparators” directory in the `cs136` area on `cortland` or in my shared area on `FreeBSD`. See the `README` file there for more information. You may also use some other data that you find interesting. If you have ideas about this, check with me before or during lab to make sure it's a reasonable thing to do.

### Submitting Your Work

---

When you're finished, create and submit a gzipped tar file “`youruserid.lab4.tar`” that includes the following:

- Your well-documented source code for all Java files used including:
  - `MyVector.java`
  - your classes that hold the data objects you decide to work with
  - your `Comparator` classes
  - your two applications
- Any additional data files you choose to use, other than the ones provided.
- A `README` file that describes what is in each Java file, describes the two applications you have chosen and what you are sorting and how for each example.
- Also include your answer to the first thought question in the text in the `README` file