Project 6 (Honors / Optional)
Due date: May 9, 11:55PM EST (**No freebies for this project!**).

This project is mandatory for students in the honors section (section 1 of this course).

This project is optional for students in section 2 of this course. If you chose to submit it, the lowest grade among your project grades will be dropped.

You may discuss any of the assignments with your classmates and tutors (or anyone else) but all work for all assignments must be entirely your own. Any sharing or copying of assignments will be considered cheating. If you get significant help from anyone, you should acknowledge it in your submission.

In this project you will be reading and understanding real code, rather than writing it yourself. Your answers should demonstrate your understanding of the code that you read and the concepts that we covered in class.

The solutions to all problems should be submitted in PDF format to NYU Classes.

Problem 1 (35 points)

Study `Arrays.java` source code file provided with Java API. Concentrate on the methods related to the sorting. What is the algorithm used when `sort()` method is called? Provide step by step description (pseudocode) for this algorithm. Discuss its advantages and disadvantages as compared to the sort algorithms that we discussed in class. Your answer should provide enough details for someone to be able to "learn" and implement this algorithm.

Note: you may need to use other source code files in order to answer the question.

Problem 2 (35 points)

Study `PriorityQueue.java` source code file that provides the implementation of a priority queue that is provided in Java API. Study both the source code and the documentation and answer the following questions:

A. How many constructors are in the class?

B. The data is stored in an array. What is the name of the array data field? What is the initial capacity of the array? What are the rules for growing the array?

C. Explain the meaning of the `transient` modifier that is used in the declaration of the array used for storing the data.

D. What methods are involved in adding an element to the queue (list all the methods that are called)?

E. What methods are involved in removing the item with the highest preference from the queue (list all the methods that are called)? NOTE: the top priority item is removed using the `poll()` method.

F. List two things in the source code that you are not familiar with (specify the line number). Use Java documentation and Internet search to figure out what these two things are and explain their meaning. Your answer should provide enough details for someone to be able to "learn" these new concepts. Feel free to include simple examples. Note: you are not allowed to use `transient` to answer this question, since you are providing a similar description for question 3 above.

Problem 3 (30 points)

Draw a binary tree for which the inorder and preorder traversals are as follows:

inorder: F E D B A C

preorder: B E F D C A

Can you always determine the shape of the binary tree based on its inorder and preorder traversals?

More generally, can you determine the shape of a binary tree based on a single traversal? How about any two traversals?

Would your answers be different for a binary search tree? If so, what would they be?