

Program #3 Grading Criteria - 100 pts total

Code Points (25 pts)

Has coordinate struct - 2 pts

Has a compareTo function that returns an integer - 2 pts

compareTo attempts to do what the specification asks - 3 pts

Your location (x, y) are the only two global variables - 2 pts (give pts if there are no globals - it turns out that making these globals wasn't necessary at all)

There is a binary search function appropriately designed - 3 pts

There is a wrapper sort function with three parameters (array, length, threshold) - 2 pts

There is a recursive sort function with 4 parameters - 2 pts

The recursive function uses threshold to determine base case vs recursive case - 2 pts

There is an insertion sort function - 2 pts

There is a merge function - 5 pts

Feel free to award partial credit in each category based on “how well” the program satisfies the criteria presented.

Style Points (10 pts)

Header comment w/name, program, date – 2 pts

Appropriate variable names – 1 pts

Appropriate white space and indenting – 1 pt

Appropriate comments about each function (what it does) - 3 pts

Internal comments in code – 3 pts

Process Document (15 pts)

Planning Phase - 5 pts

Assistance - 0 pts

Debugging Phase - 5 pts

Testing Phase - 5 pts

Be fairly lenient here, but make sure this corresponds to their work. But the quality is way worse than what you expect, please take off some points.

Execution Points (50 pts)

10 pts for doing stdin, stdout - I am being really nice here =)

There are 5 test cases, each is worth 8 pts. (One is the sample...)

Alternatively, if you debug and fix it, you may take off points for the actual error depending on how big you think it is.

DO NOT SPEND MORE THAN 5 MINUTES LOOKING FOR AN ERROR. IF YOU CAN'T FIND IT, JUST GRADE BASED ON THE ACTUAL TEST CASES.