Name: ______________________________

This assignment continues the thread of *JJ's Jolly Jumping Journey*, or *J5*, from previous ones. The primary **goal** is to gain experience on mapping concrete problems to abstract ones, finding solutions to the abstract problems (using known solutions from books, papers, and other sources), and implementing the solutions in a concrete context. Secondary goals are gaining more experience with programming, documenting algorithms and programs, studying performance, and conducting and summarizing experiments.

The main problem JJ has reached the signaling chamber and must use a specific arrangement of three jewels to summon help from the *jacketed justice* by shining the *JJ sign* into the sky. The three jewels, (ruby, garnet, and beryl, in that order), need to be mounted on poles located exactly 1, 2, and 3 meters (respectively) from the chamber's center. For the setup to work, the three jewels must lie on a straight line that points toward or above the horizon in the direction from ruby to garnet to beryl. Luckily, the chamber contains a large collection of jewel-mounting poles of different lengths. The *main task* is figuring out which poles to use for the ruby, garnet, and beryl jewels in order to produce the required straight-line arrangement. The input is the collection of poles, provided by specifying the length of each available pole. The desired output is a list of all triples of poles that satisfy the requirements.

Questions

1. (1 pt.) Write your name in the space provided above.

2. (9 pts.) Provide an abstract formulation of the main problem, using familiar mathematical concepts that are independent of JJ's journey or any other specific application. Describe the abstract formulation as precisely and as concisely as possible. Indicate how a solution to the abstract problem may be used to solve JJ's specific problem above.

3. (10 pts.) Describe an efficient algorithm for solving the problem of Question 2. Describe the algorithm in English as precisely as possible. Clearly indicate how the algorithm uses widely known solutions to the problem of Question 2, its subproblems, or related problems. Provide suitable citations for such work.

4. (10 pts.) Explain why the algorithm of Question 3 is correct.

5. (10 pts.) Provide pseudocode, using the textbook's style as a guide, for the algorithm of Question 3. Include explanatory comments and outline a proof of its correctness.

6. (10 pts.) State and justify the running time of the algorithm of Question 5 as a function of the number n of poles.

7. (150 pts.) Implement the algorithm. Test and document your work carefully and submit your packaged source code and supporting documentation.

8. (20 pts.) Conduct a brief experimental study of your implementation, measuring the running time for a suitable collection of inputs. Include your test code in your electronic submission, with suitable documentation.

9. (30 pts.) Summarize your experimental results by making effective use of charts and tables. Comment on how well the experimental results match the predictions based on your answer to Question 6. Highlight any significant differences and explain them the best you can. Include these results, comments, and explanations as a single PDF file in your submission.

[additional space for answering the earlier question]