

CUAUV Software Addendum

If you are considering applying to the Software Team, please attempt one or more of the following problems and submit your solutions before **6 PM on September 7th**. You may use any programming language that you are familiar with, but if you do not use Python, C, C++, Java, or Matlab, please submit compilation and/or execution instructions. Use your best judgment about how to accept user input if it is needed, but make sure to document it in the code.

Please submit all solutions to Benjamin at bes36@cornell.edu. Feel free to email him with any questions you may have. Good luck and have fun!

Coding Questions (Your code should be readable, properly indented and appropriately commented):

1. “Heading Averager”: In many cases, it is useful to average a few readings from the AUV's compass before proceeding. All headings are integers in the range 0 to 360. Write a heading “averager” that accepts user input and returns an average heading when requested by the user. (Note: this is not as trivial as it looks)
2. Mission Planning: Implement a data structure that you believe is well-suited for planning an autonomous mission and explain why you chose this data structure.
3. Implement a basic 4 function calculator with a command line interface using prefix or postfix notation. Please output information about the usage of your calculator before prompting the user for input (i.e. any escape characters, whether it is prefix or postfix, etc.)
4. Reverse the words in a sentence (e.g. "Hello, my name is Chris." becomes "Chris is name my, Hello."). Make sure to keep the punctuation in logical places (there isn't necessarily a correct place, just keep it consistent). Optimize for speed or space, and if you are really feeling ambitious, you can even submit two programs. Please note which program you are submitting (speed or space optimized).
5. Implement a function to check if a string entered by the user is a palindrome (e.g. “A man, a plan, a canal – Panama!”). Remember to deal with spaces, punctuation, and capitalization, since none of these affects whether or not a string is a palindrome.
6. Write a finite impulse response (FIR) filter implementation. Using the internet to find out what a FIR filter does is fine, but do NOT copy code.

Algorithm Questions. Please answer in clear and concise English. Feel free to provide a semi-formal proof for extra credit.

1. Suppose you have an array of 1001 integers. The integers are in random order, but you know each of the integers is between 1 and 1000 (inclusive). In addition, each number appears exactly once in the array, except for one number, which occurs twice. Assume that you can access each element of the array only once. Describe an algorithm to find the repeated number. If you used auxiliary storage in your algorithm, can you design an algorithm that does not require it?
2. Given that you are receiving samples from an instrument/sensor at a constant rate, and you have constant storage space, how would you design a storage algorithm that would allow a user to get a representative readout of data, no matter when he or she looked at it? In other words, representative of the behavior of the system to date.