Instructions for programming assignment

1. All students must submit their assignment by the due date through Blackboard. There would be no late submission.

2. For each assignment, there are two parts. First, you have to create multiple Journal entries in Blackboard describing your thought process and the details SDLC process for the given assignment. The Journal in Blackboard automatically keeps the date of entries, so make sure you started thinking and putting them down on the Journal early before. Some single or sparse entries before the submission deadline are not expected. These Journal entries carry 30% of the assignment grades. So please have the entries chronologically from the very beginning.

   For the source code, you should submit it in one single file named “PAx.s” for each assignment, i.e. PA1.s for the first assignment and PA2.s for the second assignment. This single file performs all the steps that are required. DO NOT submit any extra directory or files along with the source code, especially journals, documents or reports. Any extra files will be ignored.

   The basic requirement of the source code is compile-able and runnable. And all of code will be test run on the Omega platform. Any compilation failure or runtime failure will cause deduction of up to all the credits. So it would be wise if you can test run your code on Omega before submitting.

3. A typical series of Journal entries for a lab assignment might look like

   Day 1. Write down how you are thinking of the problem. Break it into major parts. Think about each of the part critically.

   Day 2. Start solving out different parts. Perhaps you may need multiple modules/functions. Start writing down the pseudo code for those parts. Try to verify with logic that the parts will work as intended.

   Day 3. Start writing code for the small parts. Make sure they are working and document them. Make sure each part is working for a sample input. Say if you are required to write a add function for decimal numbers, make sure it working for some random inputs.
Day 4. Start integrating different parts to have the complete program for the given problem. In each part verify it with some sample inputs.

Day 5. Make sure the whole program is working and write down the pseudo code for the whole program. Include some test scenario you tried to verify your program is working.

Finally. Add some overall comment on the assignment, what you have learned, what difficulties you have faced etc.

4. Your Journal is expected to contain entries according to the above guideline (at least). But you are encouraged to add more relevant entries if you can come up with.

5. In addition to this, you have to upload your source code through the Assignment feature in Blackboard. This contains 50% grade for the assignment.

   a) It must have ample comments in most of the lines of code. Tracing assembly code is very difficult if you don't have very good documentation. Even you would not understand on the following day what you have written without proper documentation. Note, without good documentation, your grade would automatically reduce by a significant portion.

   b) On the top of the program, you must write down instruction how to run your program. If you need to tweak any parameter, write it down clearly.

6. So basically, Your grading breakdown for a working assignment would be as follows:

   Journal entries (with all the SDLC steps) - 30%

   Source code 70%

   The source code mark distribution is as follows:

   Documentation - 40%

   Correctness - 60%

   Note, if you just write a few lines of code with documentation in each line, you would not get the 40% credit for documentation. This 40% is intended for a complete working program. So if your program is not working or it doesn't contain significant evidence of hard work, you may only be given partial credit for the documentation.