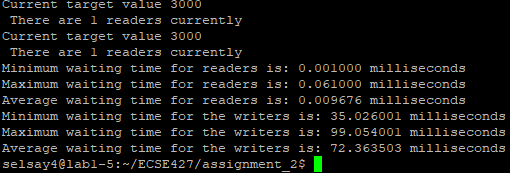
1. Write the program implementing the above pseudo-code for the readers-writers problem. Run the program. Reader repeat count is set at 60 and writer repeat count is 30. Measure the waiting times of the readers and writers. Find the minimum, maximum, and average of the waiting times. Does your implementation have a starvation problem? Briefly explain your answer. That is, explain why you concluded that there is starvation or no starvation.

**Answer:** Yes, my implementation does have a starvation problem where the writers are starving because:

* When I run the code, the output is as follows:



* As you can see the writers’ average waiting time is a lot larger than the readers’ average waiting time which proves that the writers are starving.

2-If there is no starvation, rerun the tests with different parameters so that starvation shows up. If it is not possible to run into starvation, explain why it is not possible to run into starvation.

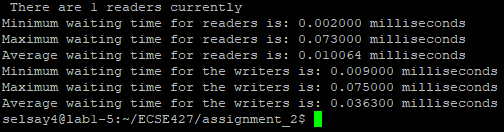
**Answer:** My preliminary implementation already presented a writers’ starvation problem. It is shown in the c file a2\_part1.c that I submitted.

3- Modify the program so that starvation problem is solved. Remember in this version of the readers-writers problem we can have writers starving. So, you create modification to the pseudo-code provided here to solve the starvation problem.

**Answer:** It is shown in the c file a2\_part2.c that I submitted.

4. Repeat the above tests and show that the starvation problem is solved.

**Answer:** After I modified the program the output is as follows:



* As shown above, the waiting average times of both readers and writers are now relatively close to each other. Which proves that the starvation problem of the writers is now solved.