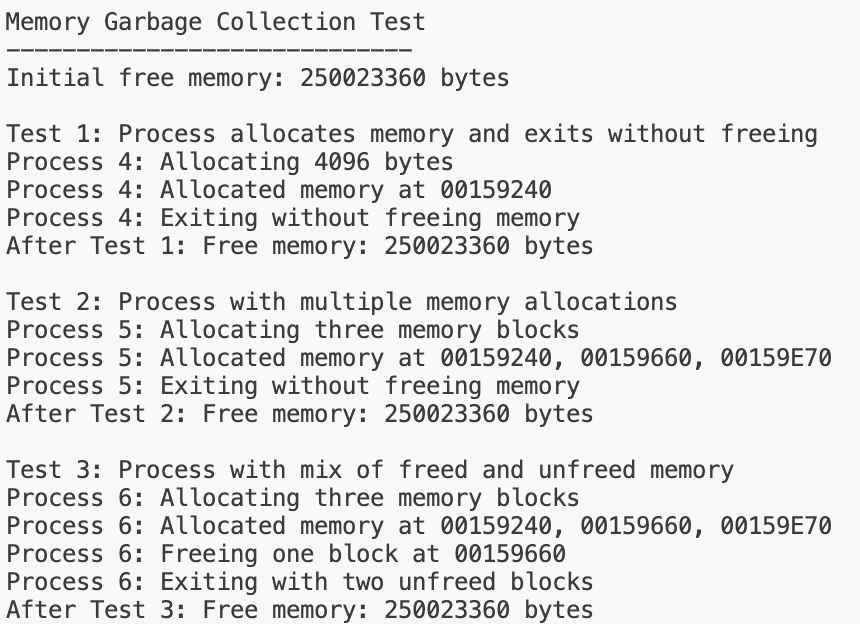
3. We tested multiple cases, including a single memory allocation without freeing, multiple memory allocations without freeing, and multiple memory allocations with one of them freed. Unfreed memory was successfully reclaimed when processes terminated in all 3 cases. See the test results in Figure 1.

Figure 1



4. We first test the basic alarm functionality and verified that it runs correctly. See the test result in Figure 2.

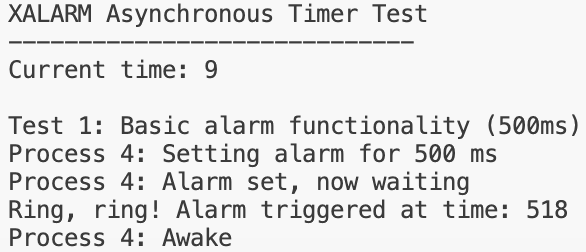


Figure 2

Then, we set alarms with invalid time. Our implementation returned STSERR as expected, see the test results in Figure 3.

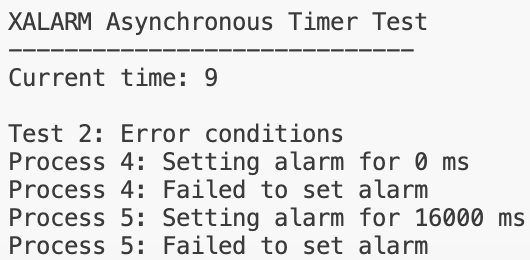


Figure 3

Finally, we set more than one alarms in one process. Since it is stated that “we will disallow xinualarm() being called twice”, our implementation should also return SYSERR in this case. See the test result in Figure 4.

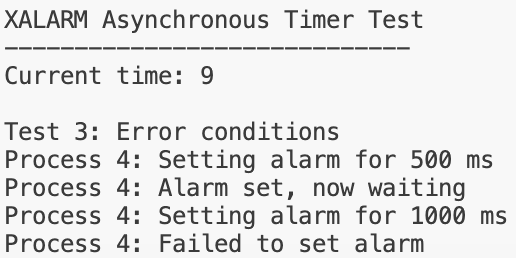


Figure 4

Bonus (b). We implemented xalrmreg() and xalrmset() to allow xalrmset() to be called multiple times where the most recent call overwrites pralrmcounter set by previous calls. We run the test and the alarm was updated successfully. See the test results in Figure 5.

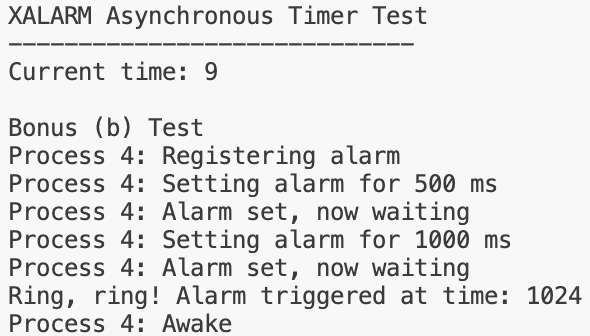


Figure 5