Government Engineering College, Thrissur

CS334 – Network Programming Lab

Documentation -

Exp 8 – First Reader Writer Problem

Date of Submission

06 June 2021

Submitted By

Kowsik Nandagopan D

Roll No 31

TCR18CS031

GECT CSE S6

Experiment 8

Implement the First Readers-Writers Problem (Using Threads and Shared Memory)

Executing program

- Code is provided in the **first.c** (Tested and verified on Ubuntu 20.04)

gcc first.c -lpthread -lrt && ./a.out

- **NOTE: This program is designed to accept only positive number. If the number is negative or zero, then corresponding error message will be displayed**

  ALGORITHM

- Get number of readers and writers required from the console

- Initialize threads reader and writer threads

- Run threads,

- In reader thread – Reader should not wait for writer. When ever reader is invoked then, reader must be executed and **no writer is allowed to execute until all the readers have been executed**. *Disadvantage of writer starvation*. **Count of readers can be maintained by a shared variable and it must be mutex controlled**. Anyway algorithm allows parallel execution of readers.

- In writer thread – **Writer should wait until all readers in queue have been completed execution**. Semaphores must be used to guarantee that only one writer is executing at a time.

- Wait until all threads complete execution

- Destroy semaphores and mutex

P. T, O

Output / Screenshots

Error Messages

Error messages produced for non positive number of readers and writers

```
hp@hp ~/Documents/S6/Network Lab/Exp8 ‹master*›
$ ./a.out
Enter number of readers: -1
Enter number of writers: 2
Error: Number of readers cannot be negative.
Enter number of readers: 2
Enter number of writers: -1
Error: Number of readers cannot be negative.
Enter number of readers: 0
Enter number of writers: 2
No reader thread created, only writer thread created
Enter number of readers: 2
Enter number of writers: 0
No writer thread created, only reader thread created
Enter number of readers: 0
Enter number of writers: 0
No writer thread created and no reader thread created
Enter number of readers:
```

Thread Execution

Consider number of readers and writers as 20

```
Enter number of readers: 20
Enter number of writers: 20
Reader thread and writer thread are created
Reader 1: 99
Reader 2: 98
Reader 4: 97
Reader 5: 95
Reader 6: 94
Reader 7: 93
Reader 3: 96
Reader 9: 92
Reader 8: 91
Reader 10: 90
Reader 12: 89
Reader 11: 87
Reader 15: 86
Reader 14: 85
Reader 13: 84
Reader 16: 88
Writer 4: 85
Writer 5: 86
Writer 6: 87
Writer 16: 88
Writer 7: 89
Writer 17: 90
```

3

Observation and Explanation

- Readers start execution and writers are kept waiting. We cannot see writer in between readers. This is because readers are given higher priority than writes

- We can see readers after writers threads have begun, during this period no writer is allowed to access shared memory space.