

First Readers-Writers Problem

Santhisenan A

April 10, 2018

1 Readers-Writers Problem

In the readers-writers problem there are some processes (termed readers) who only read the shared data, and never change it, and there are other processes (termed writers) who may change the data in addition to or instead of reading it. There is no limit to how many readers can access the data simultaneously, but when a writer accesses the data, it needs exclusive access.

The first readers-writers problem gives priority to readers. In this problem, if a reader wants access to the data, and there is not already a writer accessing it, then access is granted to the reader. A solution to this problem can lead to starvation of the writers, as there could always be more readers coming along to access the data. (A steady stream of readers will jump ahead of waiting writers as long as there is currently already another reader accessing the data, because the writer is forced to wait until the data is idle, which may never happen if there are enough readers.)

1.1 Code

```
import sys

lock = 0
no_of_readers = 0

def reader():
    global no_of_readers
    print("Input: Reader")
    if lock:
        print("A wirter has acquired the lock, please wait till it releases the")
        return
```

```

else:
    no_of_readers += 1
    print("The reader can read.")

def release_reader():
    global no_of_readers
    if(no_of_readers == 0):
        print("No readers")
    else:
        no_of_readers -= 1

def writer():
    global lock
    if lock:
        print("Another writer is reading.")
    elif no_of_readers:
        print("Another reader(s) is(are) reading")
    else:
        lock = 1

def release_writer():
    global lock
    if lock == 0:
        print("No writer is writing")
    else:
        lock = 0

while True:
    read = input()
    if read == 'read':
        reader()
    elif read == 'write':
        writer()
    elif read == 'relread': # release reader
        release_reader()
    elif read == 'relwrite': # release writer
        release_writer()
    else:
        sys.exit()

```

```

first_readers_writers.py - first exp 4
2053 [09:09 PM] ~/.../networking-lab/exp4 master python3 first_readers_writers.py
read
Input: Reader
The reader can read.
read
Input: Reader
The reader can read.
write
Another reader(s) is(are) reading
relread
relread
write
write
Another writer is reading.
read
Input: Reader
A wirtter has acquired the lock, please wait till it releases the lock
relwrite
read
Input: Reader
The reader can read.
[]

```

Figure 1: first-readers-writers

2 output