COP 4600 Operating Systems - Project 3

A Shared Protected Circular Queue and Communication Between Threads

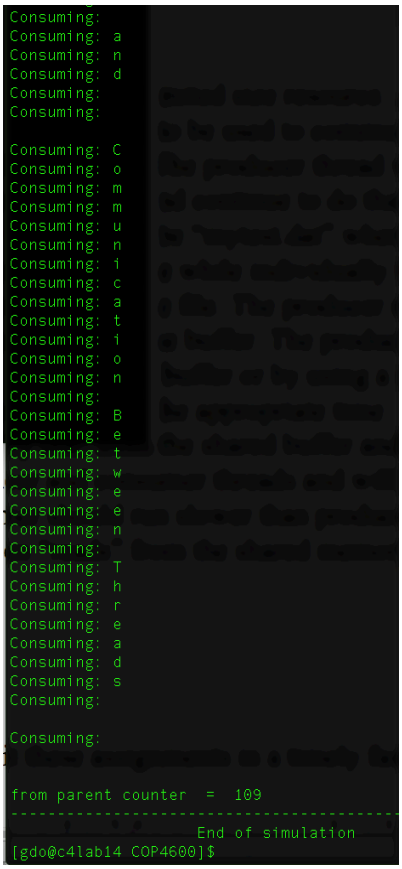
Gregory O'Marah Net ID: gdo U34240613

The purpose of this project was to demonstrate how to use semaphores to protect a limited size resource. A circular buffer was created and placed in shared memory for communication between producer and consumer threads. The producer reads characters from a file one by one and placed them in the buffer queue. The consumer pulled one character from the buffer at a time, and printed it to the screen. When the end of file was reached, the producer signaled by placing the character '*' to let the consumer know that it was finished. A parent process was initiated that created both consumer and producer threads, and which deleted the semaphores after they were both terminated. The program terminated by printing out a successful completion greeting, and releasing the shared memory. As specified in the project assignment, a one second delay was implemented in the consumer function, to ensure the consumer would run slower than the producer.

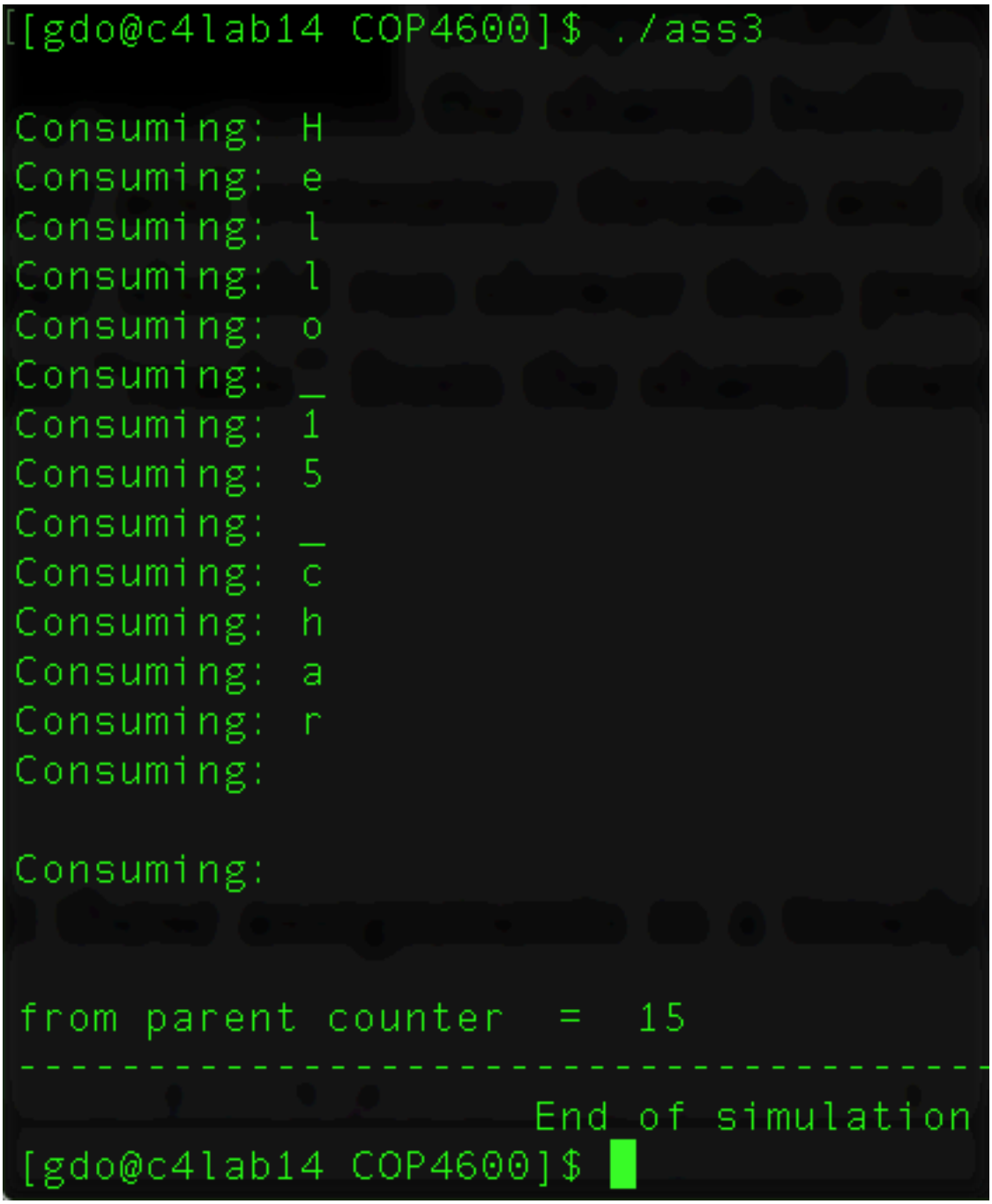
The results were as expected. The mytest.dat file was successfully read and placed on the buffer by the producer. The consumer picked up and placed to the stdout each character one at a time. The semaphored did their job by protecting the shared memory segment so that each thread had mutually exclusive access to the buffer at a time, and they were able to signal each other when they were full and empty as required.

Below are some screen shots to show the result from several test runs of the program, with various sized data files. Each one completed without errors and within the expected time frame, given the 1 second delay that was implemented on the consumer side function. There was no loss of data and all characters were passed as expected in the order they were found in the test data files.

```
[gdo@c4lab14 COP4600]$ ./ass3

Consuming: C
Consuming: O
Consuming: P
Consuming:
Consuming: 4
Consuming: 6
Consuming: 0
Consuming: 0
Consuming:
Consuming: O
Consuming: p
Consuming: e
Consuming: r
Consuming: a
Consuming: t
Consuming: i
Consuming: n
Consuming: g
Consuming:
Consuming: S
Consuming: y
Consuming: s
Consuming: t
Consuming: e
Consuming: m
Consuming: s
Consuming:
Consuming: -
Consuming:
Consuming: P
Consuming: r
Consuming: o
Consuming: j
Consuming: e
Consuming: c
Consuming: t
Consuming:
Consuming: 3
Consuming:

Consuming: A
Consuming:
Consuming: S
Consuming: h
Consuming: a
Consuming: r
Consuming: e
Consuming: d
Consuming:
Consuming: P
Consuming: r
Consuming: o
Consuming: t
Consuming: e
Consuming: c
Consuming: t
Consuming: e
Consuming: d
Consuming:
Consuming: C
Consuming: i
Consuming: r
Consuming: c
Consuming: u
Consuming: l
Consuming: a
Consuming: r
Consuming:
Consuming: Q
Consuming: u
Consuming: e
Consuming: u
Consuming: e
```


Consuming:
Consuming: a
Consuming: n
Consuming: d
Consuming:
Consuming:
Consuming: C
Consuming: o
Consuming: m
Consuming: m
Consuming: u
Consuming: n
Consuming: i
Consuming: c
Consuming: a
Consuming: t
Consuming: i
Consuming: o
Consuming: n
Consuming:
Consuming: B
Consuming: e
Consuming: t
Consuming: w
Consuming: e
Consuming: e
Consuming: n
Consuming:
Consuming: T
Consuming: h
Consuming: r
Consuming: e
Consuming: a
Consuming: d
Consuming: s
Consuming:
Consuming:
from parent counter = 109
End of simulation
[gdo@c4lab14 COP4600]$

```
[[gdo@c4lab14 COP4600]$ ./ass3

Consuming:  g
Consuming:  r
Consuming:  e
Consuming:  g
Consuming:  _
Consuming:  o
Consuming:  m
Consuming:  a
Consuming:  r
Consuming:  a
Consuming:  h
Consuming:  _
Consuming:  c
Consuming:  o
Consuming:  p
Consuming:  _
Consuming:  4
Consuming:  6
Consuming:  0
Consuming:  0
Consuming:  _
Consuming:  p
Consuming:  r
Consuming:  o
Consuming:  j
Consuming:  e
Consuming:  c
Consuming:  t
Consuming:  _
Consuming:  3
Consuming:


from parent counter  =  31
---------------------------------------
                    End of simulation
[gdo@c4lab14 COP4600]$
```



Attached are the completed program code, ass3.c, the data files used in testing, and a readme.txt file to explain the procedure required to compile and run the program.