

Fernando Juarez & Michael Trani
Assignment 4 - Producer Consumer
CS 433 - Operating Systems
November 22, 2019

Program 4 Report

Program description: This program is an implementation of the Producer Consumer problem and the use of mutex locks and semaphores that can be used to solve the critical-section problem. We attempted to fully implement monitors using semaphores but due to errors and time we were unable to complete our goal as we need extra credit so we left it as is. So main is the driver and implementation of the Producer Consumer Problem. Producer function produces items into buffer while Consumer function does the opposite and simply deletes the items from the buffer with the use of deleteItem as it simulates that it consumes what was produced first as it uses FIFO implementation. And after each transaction or action the buffer calls showBuffer to print the current result of what is happening at each step.

Only issue we had regarding this assignment was getting the implementation to work as we had some minor issues with our code to implement this procedure of producer and consumer problem but despite this, the assignment runs and performs as expected. First of all, rand() was giving us random values but when dealing with several compilations we noticed that similar values were given so not really random. Then we started to notice that similar values were also being outputted meaning that for example item #16 would be produced then deleted and then produced once again which made us research the implementation of srand and realized that we had to move the implementation before sleep in order to stop any values from repeating. Until we used srand(time(0)) which then gave us true random values every time we compiled as shown in the example output at bottom of page. As reference we used the following site: <https://www.geeksforgeeks.org/rand-and-srand-in-cpp/>.

Overall program was not hard to get working but rather the libraries and specific details needed before hand was the difficulty of this assignment. The website <http://www.cplusplus.com/reference/> came in handy as well as the textbook in order to get our program up and running. The use of g++ -pthread main.cpp was simply weird at first until reading why it wouldn't compile the command of g++ main.cpp as is. When adding items to the buffer we initially had buffer[top]=item; and top = (top +1) % buffSize but were getting a bunch of errors and a segmentation fault so i had to change both the implementation of addItem and deleteItem. By simply swapping the top with end eliminated all errors and segmentation fault was also gone. Another issue was using atoi and usleep vs sleep as these were crucial to the implementation of our program. Since usleep stops the process in microseconds and sleep in seconds we had some difficulties choosing the

correct or best way for this precise assignment. But despite the research and further knowledge with c++ libraries, we were able to get a running program.

Files Included: main.cpp, Makefile, and output

main.cpp - Contains the implementation of producer consumer and a small driver for the assignment.

Makefile - compiles the source files

output - program executable

To run: use the Makefile which will compile the source files and create an executable called output.

```
[juare066@empress A4]$ emacs main.cpp
[juare066@empress A4]$ g++ -pthread main.cpp
[juare066@empress A4]$ ./a.out 4 4 3

=====
CS 433 Programming assignment 4
Author: Fernando Juarez and Michael Trani
Date: 11/22/2019
Course: CS433 - Prof.Zhang: Operating Systems
Description : Program to simulate Producer Consumer
=====

Item 44 inserted by a producer.
The current content of the Buffer is[ 44 ]

Item 56 inserted by a producer.
The current content of the Buffer is[ 44 56 ]

Item 36 inserted by a producer.
The current content of the Buffer is[ 44 56 36 ]

Item 44 removed by a consumer.
The current content of the Buffer is[ 56 36 ]

Item 56 removed by a consumer.
The current content of the Buffer is[ 36 ]

Item 73 inserted by a producer.
The current content of the Buffer is[ 36 73 ]

Item 36 removed by a consumer.
The current content of the Buffer is[ 73 ]

[juare066@empress A4]$ ./a.out 4 4 3

=====
CS 433 Programming assignment 4
Author: Fernando Juarez and Michael Trani
Date: 11/22/2019
Course: CS433 - Prof.Zhang: Operating Systems
Description : Program to simulate Producer Consumer
=====

Item 84 inserted by a producer.
The current content of the Buffer is[ 84 ]

Item 16 inserted by a producer.
The current content of the Buffer is[ 84 16 ]

Item 84 removed by a consumer.
The current content of the Buffer is[ 16 ]

Item 34 inserted by a producer.
The current content of the Buffer is[ 16 34 ]

Item 16 removed by a consumer.
The current content of the Buffer is[ 34 ]

Item 45 inserted by a producer.
The current content of the Buffer is[ 34 45 ]

Item 34 removed by a consumer.
The current content of the Buffer is[ 45 ]

[juare066@empress A4]$
```