Programming-Assignment-4 Report

Gustavo Hammerschmidt*
CS433 - Assignment 4

March 27, 2020

1 Name and Description of the submitted files:

1.1 ITEM.cpp

This class is used to represent an item, that can be added to the FIFO buffer. This class has all the basic methods to operate on its attributes. Any detail that is not explicitly defined here is well-documented on the class files, please check them for more information.

1.2 FIFO.cpp

This class represents a list of items. It has all the basic methods to operate on its attributes. Any detail that is not explicitly defined here is well-documented on the class files, please check them for more information.

1.3 Main.cpp

This class has the main function, it creates threads to its procedures(either a consumer or a producer procedure) and executes them in a way to make operations on a fifo buffer. Any detail that is not explicitly defined here is well-documented on the class files, please check them for more information.

1.4 Makefile

The Makefile used is the same provided in the class server with some modifications. Any detail that is not explicitly defined here is well-documented on the class files, please check them for more information.

*hamme032@cougars.csusm.edu

1.5 Assign4

This file is the c++ project executable file.

2 Compilation of the program

To compile the program, use the Makefile to create the object classes of each .cpp class. I used the makefile file available on the example code on the server, making small changes to it in order to compile the program and to be able to execute it. It was compiled and executed on linux. I solely recommend executing it in Linux.

3 Results of the program

I believe that I've made all the requirements that should be done. The program was first written on visual studio and then transposed to the unix virtual machine environment, with some adjustments made and some errors fixed. I'd say that my executable file performs well and it implements the producer-consumer problem properly. I've used the Pthreads API to make the code threaded and used the mutex thread to evict collisions or misuse of the data by the threads on the critical sections of the procedures. My program allows many producer and/or consumer threads to be created and I also added some flags. So, for gaining extra credits, I've implemented a fifo queue to make the buffer with some methods to make insertion and remotion of items more efficient; my item class was designed to make the buffer more flexible, so any items can be inserted into the buffer as long as the ITEM class is modified properly and still has an identifier(but I have not focused on this aspect in this program, I have only made it possible for another implementations of items); and I have defined flags that can modify the size of the buffer to whatever the user defines on the kernel, that can modify the delay time for either producer and consumer after they finish their procedures and that can modify the minimum delay time waited(a percentage of the delay time defined) by the producer or consumer procedure(because the time defined is only a range of possible waiting times, once the waiting time is defined randomly to make the program more interesting; more information on these is found on the source code; the last two modifications are also user definable). These flags are optional so they won't interfere within the program's execution if not defined, these flags alter the default values of predefined variables. It may look a little complex, but I have defined, in the program's execution, what the flags are and how to use them, also any detail that is not explicitly defined here is well-documented on the class files, please check them for more information. Last of all, I have colored the output to make it more easily understood, so finding the usage of the flags is easily identifiable. I affirm that I have not made use of any means of code copy or even used other's code. I have not taken any reference on any procedure involved on my code, except that I took the example given in the Operating Systems ed. 10 book as a reference, which guided the development of my program with much emphasis. I've executed the code both on a linux and a windows environment and the code performed well on both of them.