Christian Everett

[ceverett@cs.uml.edu](mailto:ceverett@cs.uml.edu)

Write-Up

In assignment one my degree of success was 100%. This assignment was compiled on cs.uml.edu. My approach to the problem began with downloading and analyzing the code for the Donuts.h header file and utilities.c file. I was able to take blocks from the sample code and start writing the code for the producer process in proddonuts.c file as well as the consumer code file consdonuts.c. The producer c file was responsible for setting up a shared memory segment that multiple consumer processes would link to. This shared memory segment was to hold four ring buffers and for out pointers. The programs were to use semaphores to block access to the ring buffers when one of the processes was modifying an element of the buffers. The first part of the assignment required that the program be executed with one producer and five consumers, each consumer would consume 10 dozen donuts and print out each dozen in a separate table. There are five output files with each of the consumers 10 tables (the files are c1, c2, c3, c4, c5). The second task was to find the average deadlock percentage for different sizes of the four ring buffers (see table 1). This graph was liner; as queue size decreased the probability of deadlock increased (see graph 1).

Table

|  |  |
| --- | --- |
| Queue Size | Deadlock % |
| 10 | 100% |
| 20 | 100% |
| 30 | 90% |
| 40 | 70% |
| 50 | 55% |
| 60 | 50% |
| 70 | 30% |
| 80 | 23% |
| 90 | 15% |
| 100 | 10% |

Graph

Lastly with the 50% deadlock queue size I was required to create a second graph (graph 2) and table (table 2) with the number of consumers as the variable data. I ran tests with 1-10 consumers and 1 producer.

Table

|  |  |
| --- | --- |
| Number Consumers | % Deadlocked |
| 1 | 4% |
| 2 | 10% |
| 3 | 16% |
| 4 | 26% |
| 5 | 50% |
| 6 | 70% |
| 7 | 76% |
| 8 | 80% |
| 9 | 86% |
| 10 | 95% |

Graph