**CSCE 611 600: OPERATING SYSTEMS Project #1**

**Name:** Rohan Chaudhury

**Email Address:** [rohan.chaudhury@tamu.edu](mailto:rohan.chaudhury@tamu.edu)

**UIN:** 432001358

**List of people I have worked with:** Abhishek Sinha, Rohit Sah, Shubham Gupta, Sherine Davis Kozhikadan

**Details of the Project:**

The following document gives a detailed description of the code that I have implemented as a “correct solution” for the multiple dimensioned, multiple producer/consumer problem. My submission chaudhuryr\_proj1.tar.gz consists of the following files:

1. The code chaudhuryr\_proj1.c
2. The makefile
3. The project documentation chaudhuryr\_proj1.docx
4. The executable file chaudhuryr\_proj1.exe

In order to compile the code we need change our present working directory to the directory containing our code and enter the following command:

**gcc project1.c -o chaudhuryr\_proj1 -lpthread -lrt**

It will generate an executable with the name chaudhuryr\_proj1.exe

Then to run the executable we need to run the following command:

**./chaudhuryr\_proj1.exe**

Alternatively I have included the makefile for the project and we need to enter the following commands to compile our code.

**make clean**

**make**

The **make clean** command will delete any pre-existing binary with the name chaudhuryr\_proj1.exe and the **make** command will compile the binary. Then we can run the created executable with the command:

**./chaudhuryr\_proj1.exe**

**Code Explanation and Output:**

The code has the following macros defined which can be changed as per the user demands, their descriptions are provided as comments below:

//Defining macros

#define TRUE 1

#define FALSE 0

//A message can have the following 3 status: PRODUCE, ACKNOWLEDGE, STOPPING\_CRITERIA\_STATUS

//"PRODUCE" message status indicates the signal for the producer to start producing based on the request from a consumer

#define PRODUCE 0

//"ACKNOWLEDGE" message indicates that the producer has produced data for a certain consumer

#define ACKNOWLEDGE 1

//"STOPPING\_CRITERIA\_STATUS" message status indicates that all the consumers have exited and so the producers should start exiting

#define STOPPING\_CRITERIA\_STATUS 2

//"SIZE" indicates the size of the fifo buffer/queue used for message passing

#define SIZE 2

//The FIFO QUEUE/BUFFER

#define FIFO\_QUEUE "./fifo\_queue"

#define QUEUE\_FULL "/Queue\_full"

#define QUEUE\_EMPTY "/Queue\_empty"

//THE Binary semaphore lock

#define BINARY\_SEMAPHORE\_LOCK "/binary\_semaphore\_lock"

//Semaphore to keep count of the number of consumer processes which have exited

#define NUM\_EXITED\_CONSUMERS "/num\_exited\_consumers"

//Number of Producers

#define PRODUCERS\_COUNT 6

//Number of Consumers

#define CONSUMERS\_COUNT 8

//"STOPPING\_CRITERIA" denotes the number of times a consumer will request and consume data

#define STOPPING\_CRITERIA 5

Also we are defining the following semaphores for our solution:

//Semphores for managing the FIFO Queue

//Queue\_full semaphore gives the number of full slots in the Queue/Buffer

sem\_t \*Queue\_full;

//Queue\_empty semaphore gives the number of empty slots in the Queue/Buffer

sem\_t \*Queue\_empty;

//THE Binary semaphore lock for mutual exclusion

sem\_t \*binary\_semaphore\_lock;

//Semaphore to keep count of the number of consumer processes which have exited

sem\_t \*num\_exited\_consumers;

I have used message passing, semaphores and a buffer for the solution. The message structure looks like the following with details about the variables:

struct message

{

    //the data produced by producer

    int data;

    //status of the message

    int status;

    //timestamp of the message in microseconds calculated from the beginning of execution of the program

    long time\_stamp;

    //consumer\_id denotes the consumer process number which originated the message request

    int consumer\_id;

    //producer\_id denotes the producer process number which produced the data

    int producer\_id;

    // int stop\_c;

};

**MAIN FUNCTION**

My main function in the program will first

1. Clear up any previously stored data

int main()

{

    //clearing up any data which might be stored previously

    unlink(FIFO\_QUEUE);

    sem\_unlink(QUEUE\_EMPTY);

    sem\_unlink(QUEUE\_FULL);

    sem\_unlink(BINARY\_SEMAPHORE\_LOCK);

    sem\_unlink(NUM\_EXITED\_CONSUMERS);

    sem\_destroy(Queue\_empty);

    sem\_destroy(Queue\_full);

    sem\_destroy(binary\_semaphore\_lock);

    sem\_destroy(num\_exited\_consumers);

1. Set up the semaphores and the FIFO Queue

    //setting up the semaphores

    Queue\_empty = sem\_open(QUEUE\_EMPTY, O\_CREAT, 0777, SIZE);

    Queue\_full = sem\_open(QUEUE\_FULL, O\_CREAT, 0777, 0);

    binary\_semaphore\_lock = sem\_open(BINARY\_SEMAPHORE\_LOCK, O\_CREAT, 0777, 1);

    num\_exited\_consumers=sem\_open(NUM\_EXITED\_CONSUMERS, O\_CREAT, 0777, 0);

    int a,b,c,d;

    sem\_getvalue(binary\_semaphore\_lock, &a);

    sem\_getvalue(Queue\_empty, &b);

    sem\_getvalue(Queue\_full, &c);

    sem\_getvalue(num\_exited\_consumers, &d);

    //Printing the semaphore values

    printf("\nValues of binary\_semaphore\_lock = %d, Queue\_empty = %d, Queue\_full = %d\n", a,b,c );

    //declaring the FIFO Queue

    const char \*Queue = FIFO\_QUEUE;

    //creating the Fifo queue

    mkfifo(Queue, 0666);

1. Then create the Producer child processes and Consumer child processes

    //Spawning Child Processes for Producers

    printf("\nSpawning child Processes for producers\n");

    for (i = 0; i < PRODUCERS\_COUNT; i++)

    {

        producer\_id[i] = fork(); //creating a child process

        if (producer\_id[i] == 0)

        {

            //if inside the child process then start the producer function and then break

            producer(i + 1);

            break;

        }

    }

    //Spawning Child Processes for Consumers

    printf("\nSpawning child Processes for consumers\n");

     for (j = 0; j < CONSUMERS\_COUNT; j++)

    {

        consumer\_id[j] = fork(); //creating a child process

        if (consumer\_id[j] == 0)

        {

            //if inside the child process then start the consumer function and then break

            consumer(j + 1);

            break;

        }

    }

1. Then wait for the child processes to finish executing

    // This for loop with the wait(NULL) statement will move on only when a process ends,

    // so it waits until PRODUCERS\_COUNT + CONSUMERS\_COUNT number of processes ends.

    for (k = 0; k < PRODUCERS\_COUNT + CONSUMERS\_COUNT; k++)

        wait(NULL);

    //The main function enters here when all the child processes have terminated

    printf("\nAll producer and consumer processes terminated\n");

    printf("\nExiting the program\n");

1. Then at the end it will clear all the stored data

    //clearing up all the stored data

    unlink(FIFO\_QUEUE);

    sem\_unlink(QUEUE\_EMPTY);

    sem\_unlink(QUEUE\_FULL);

    sem\_unlink(BINARY\_SEMAPHORE\_LOCK);

    sem\_unlink(NUM\_EXITED\_CONSUMERS);

    sem\_destroy(Queue\_empty);

    sem\_destroy(Queue\_full);

    sem\_destroy(binary\_semaphore\_lock);

    sem\_destroy(num\_exited\_consumers);

    return 0;

}

The producer child processes will be created first followed by the consumer child processes

I have defined the producer and consumer functions which will be executed by the producer and consumer child processes respectively.

**Consumer Function:**

The consumer child process has 2 states which will be repeated as the pre-defined STOPPING\_CRITERIA number of times:

1. **Requesting state:**

In this state the consumer child process will:

1. Check if STOPPING CRITERIA is met, if it is met then it will exit the process
2. Else, it will check if the Queue has atleast 1 empty slot before writing the request message onto the Queue using the sem\_wait(Queue\_empty) command
3. Then wait to acquire the binary\_semaphore\_lock which I am using as a mutex lock using the sem\_wait(binary\_semaphore\_lock) command
4. Then write the request onto the message queue and change its state to consuming state
5. **Consuming State:**
6. Check if STOPPING CRITERIA is met, if it is met then it will exit the process
7. Else, it will check if the Queue has atleast 1 message to be read using the sem\_wait(Queue\_full) command
8. Then wait to acquire the binary\_semaphore\_lock which I am using as a mutex lock using the sem\_wait(binary\_semaphore\_lock) command
9. Then read the message from the front of the Queue
10. The consumer process will only consume the message which has the same consumer\_id as the consumer process. The consumer\_id was assigned to the message by the consumer process when it had written the request meant for the producers onto the Queue.
11. If the consumer\_id variable of the message matches with the id of the consumer process and the message is acknowledged by the producer then the data is consumed by the consumer process
12. If the consumer\_id variable of the message matches with the id of the consumer process and the message is not acknowledged by the producer then the message is again put back on the Queue to be acknowledged by the Producer
13. If the read message has a different consumer\_id variable that the consumer id of the current process which means that this message is meant for a different consumer process, then the consumer process will write this message back onto Queue without making any changes to it for the appropriate consumer process to receive

When all the consumer processes have exited, then at the end of the last consumer process I am entering a termination message with message status as STOPPING\_CRITERIA\_STATUS which signals the producer processes that all the consumer processes are done executing and have exited so the producer processes should also exit.

**Producer Function:**

The producer function does the following:

1. First checks if there is any data on the Queue using the sem\_wait(Queue\_full) command
2. Acquires the binary\_semaphore\_lock lock using the sem\_wait(binary\_semaphore\_lock) command
3. The reads the message from the front of the Queue.

The producer function has three different modes of functioning based on the received message status.

1. **When message status is STOPPING\_CRITERIA\_STATUS:**
   1. Then the producer process exits after pushing the same message back to the queue so that the other producer processes can exit. This status signals the producer processes that all the consumer processes are done executing and have exited so the producer processes should also exit.
2. **When message status is PRODUCE:**
   1. The message is acknowledged and data is published in the message. The message is then written back onto the Queue
3. **When message status is ACKNOWLEDGE:**
   1. If the read messsage has the status ACKNOWLEDGE then the message is meant for a consumer process to be consumed. Then we simply push back this message onto the queue and release the binary\_semaphore\_lock

This is the entire flow of the code.

**CODE OUTPUT:**

I am using the following pre-defined macro values to show the output:

//"SIZE" indicates the size of the fifo buffer/queue used for message passing

#define SIZE 2

//Number of Producers

#define PRODUCERS\_COUNT 6

//Number of Consumers

#define CONSUMERS\_COUNT 8

//"STOPPING\_CRITERIA" denotes the number of times a consumer will request and consume data

#define STOPPING\_CRITERIA 5

**Explanation of output:**

**Following is the output of the code. In this output we will be able to see that 8 consumer child processes were spawned and 6 producer child processes were spawned. The FIFO Queue size was defined as 2. Each consumer child process requested for data exactly STOPPING\_CRITERIA (=5) number of times and consumed data when its id matched with the consumer\_id variable of the message exactly STOPPING\_CRITERIA (=5) number of times. After that the consumer processes exited. Based on the requests from the consumer child processes the producer child processes were producing the data and sending it back to queue. When all the consumer processes had exited the producer child processes started exiting one by one. When all the producer and consumer child process had exited the main parent process exited and the program terminated.**

**The log statements will not be printed serially as they are printed by different processes.**

**OUTPUT ALONG WITH TIMESTAMPS:**

Values of binary\_semaphore\_lock = 1, Queue\_empty = 2, Queue\_full = 0

Spawning child Processes for producers

Inside Producer Process --> Producer created with id: 1

Inside Producer Process --> Producer created with id: 2

Inside Producer Process --> Producer created with id: 3

Spawning child Processes for consumers

Inside Producer Process --> Producer created with id: 5

Inside Consumer Process --> Consumer created with id: 1

Inside Consumer Process --> Consumer with process number 1 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 1 produced data (Data= 1067398392) for consumer process number (assigned as consumer id) 1 as it received a PRODUCE request message (Message timestamp: [676 microseconds])

Inside Consumer Process --> Consumer created with id: 2

Inside Consumer Process --> Consumer with process number 2 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 2 produced data (Data= 1067398392) for consumer process number (assigned as consumer id) 2 as it received a PRODUCE request message (Message timestamp: [908 microseconds])

Inside Consumer Process --> Consumer created with id: 6

Inside Consumer Process --> Consumer created with id: 5

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 1 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 1, hence consumed data from Queue, data= 1067398392, producer id= 1 (Message timestamp: [676 microseconds])

Inside Consumer Process --> Consumer with process number 6 sent request to producers

Inside Consumer Process --> Consumer created with id: 8

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 2 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 2, hence consumed data from Queue, data= 1067398392, producer id= 2 (Message timestamp: [908 microseconds])

Inside Producer Process --> Producer with process number (assigned as producer id) 1 produced data (Data= 1774275636) for consumer process number (assigned as consumer id) 6 as it received a PRODUCE request message (Message timestamp: [1075 microseconds])

Inside Consumer Process --> Consumer created with id: 7

Inside Consumer Process --> Consumer with process number 7 sent request to producers

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 6 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 6, hence consumed data from Queue, data= 1774275636, producer id= 1 (Message timestamp: [1075 microseconds])

Inside Producer Process --> Producer with process number (assigned as producer id) 1 produced data (Data= 2099175287) for consumer process number (assigned as consumer id) 7 as it received a PRODUCE request message (Message timestamp: [1153 microseconds])

Inside Consumer Process --> Consumer with process number 5 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 5 produced data (Data= 1067398392) for consumer process number (assigned as consumer id) 5 as it received a PRODUCE request message (Message timestamp: [1300 microseconds])

Inside Consumer Process --> Consumer with process number 2 sent request to producers

Inside Consumer Process --> Consumer with process number 8 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 2 produced data (Data= 1774275636) for consumer process number (assigned as consumer id) 8 as it received a PRODUCE request message (Message timestamp: [1517 microseconds])

Inside Consumer Process --> Consumer with process number 1 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 1 produced data (Data= 126592119) for consumer process number (assigned as consumer id) 2 as it received a PRODUCE request message (Message timestamp: [1733 microseconds])

Inside Producer Process --> Producer with process number (assigned as producer id) 2 produced data (Data= 2099175287) for consumer process number (assigned as consumer id) 1 as it received a PRODUCE request message (Message timestamp: [1756 microseconds])

Inside Consumer Process --> Consumer with process number 6 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 3 produced data (Data= 1067398392) for consumer process number (assigned as consumer id) 6 as it received a PRODUCE request message (Message timestamp: [1994 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 8 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 8, hence consumed data from Queue, data= 1774275636, producer id= 2 (Message timestamp: [1517 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 2 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 2, hence consumed data from Queue, data= 126592119, producer id= 1 (Message timestamp: [1733 microseconds])

Inside Consumer Process --> Consumer with process number 2 sent request to producers

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 7 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 7, hence consumed data from Queue, data= 2099175287, producer id= 1 (Message timestamp: [1153 microseconds])

Inside Consumer Process --> Consumer with process number 8 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 5 produced data (Data= 1774275636) for consumer process number (assigned as consumer id) 8 as it received a PRODUCE request message (Message timestamp: [2712 microseconds])

Inside Producer Process --> Producer with process number (assigned as producer id) 2 produced data (Data= 126592119) for consumer process number (assigned as consumer id) 2 as it received a PRODUCE request message (Message timestamp: [2736 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 8 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 8, hence consumed data from Queue, data= 1774275636, producer id= 5 (Message timestamp: [2712 microseconds])

Inside Consumer Process --> Consumer with process number 7 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 2 produced data (Data= 1576740829) for consumer process number (assigned as consumer id) 7 as it received a PRODUCE request message (Message timestamp: [2858 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 1 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 1, hence consumed data from Queue, data= 2099175287, producer id= 2 (Message timestamp: [1756 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 7 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 7, hence consumed data from Queue, data= 1576740829, producer id= 2 (Message timestamp: [2858 microseconds])

Inside Consumer Process --> Consumer with process number 1 sent request to producers

Inside Consumer Process --> Consumer with process number 7 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 5 produced data (Data= 2099175287) for consumer process number (assigned as consumer id) 7 as it received a PRODUCE request message (Message timestamp: [3179 microseconds])

Inside Consumer Process --> Consumer with process number 8 sent request to producers

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 7 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 7, hence consumed data from Queue, data= 2099175287, producer id= 5 (Message timestamp: [3179 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 2 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 2, hence consumed data from Queue, data= 126592119, producer id= 2 (Message timestamp: [2736 microseconds])

Inside Consumer Process --> Consumer with process number 7 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 3 produced data (Data= 1774275636) for consumer process number (assigned as consumer id) 7 as it received a PRODUCE request message (Message timestamp: [3525 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 7 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 7, hence consumed data from Queue, data= 1774275636, producer id= 3 (Message timestamp: [3525 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 6 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 6, hence consumed data from Queue, data= 1067398392, producer id= 3 (Message timestamp: [1994 microseconds])

Inside Consumer Process --> Consumer with process number 6 sent request to producers

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 5 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 5, hence consumed data from Queue, data= 1067398392, producer id= 5 (Message timestamp: [1300 microseconds])

Inside Consumer Process --> Consumer with process number 7 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 2 produced data (Data= 499724341) for consumer process number (assigned as consumer id) 6 as it received a PRODUCE request message (Message timestamp: [3710 microseconds])

Inside Producer Process --> Producer with process number (assigned as producer id) 3 produced data (Data= 2099175287) for consumer process number (assigned as consumer id) 7 as it received a PRODUCE request message (Message timestamp: [3739 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 7 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 7, hence consumed data from Queue, data= 2099175287, producer id= 3 (Message timestamp: [3739 microseconds])

Inside Consumer Process --> Consumer with process number 5 sent request to producers

Inside Consumer Process --> Consumer with process number 7 exiting

Number of consumers exited 1

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 6 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 6, hence consumed data from Queue, data= 499724341, producer id= 2 (Message timestamp: [3710 microseconds])

Inside Producer Process --> Producer with process number (assigned as producer id) 2 produced data (Data= 1812720505) for consumer process number (assigned as consumer id) 5 as it received a PRODUCE request message (Message timestamp: [4044 microseconds])

Inside Consumer Process --> Consumer with process number 2 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 3 produced data (Data= 126592119) for consumer process number (assigned as consumer id) 8 as it received a PRODUCE request message (Message timestamp: [4198 microseconds])

Inside Consumer Process --> Consumer with process number 6 sent request to producers

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 5 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 5, hence consumed data from Queue, data= 1812720505, producer id= 2 (Message timestamp: [4044 microseconds])

Inside Producer Process --> Producer with process number (assigned as producer id) 3 produced data (Data= 1576740829) for consumer process number (assigned as consumer id) 1 as it received a PRODUCE request message (Message timestamp: [4298 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 8 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 8, hence consumed data from Queue, data= 126592119, producer id= 3 (Message timestamp: [4198 microseconds])

Inside Producer Process --> Producer with process number (assigned as producer id) 5 produced data (Data= 126592119) for consumer process number (assigned as consumer id) 2 as it received a PRODUCE request message (Message timestamp: [4771 microseconds])

Inside Consumer Process --> Consumer with process number 8 sent request to producers

Inside Consumer Process --> Consumer with process number 5 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 1 produced data (Data= 1576740829) for consumer process number (assigned as consumer id) 6 as it received a PRODUCE request message (Message timestamp: [4985 microseconds])

Inside Producer Process --> Producer with process number (assigned as producer id) 1 produced data (Data= 499724341) for consumer process number (assigned as consumer id) 5 as it received a PRODUCE request message (Message timestamp: [5009 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 1 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 1, hence consumed data from Queue, data= 1576740829, producer id= 3 (Message timestamp: [4298 microseconds])

Inside Consumer Process --> Consumer with process number 1 sent request to producers

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 6 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 6, hence consumed data from Queue, data= 1576740829, producer id= 1 (Message timestamp: [4985 microseconds])

Inside Consumer Process --> Consumer with process number 6 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 5 produced data (Data= 1576740829) for consumer process number (assigned as consumer id) 6 as it received a PRODUCE request message (Message timestamp: [5487 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 6 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 6, hence consumed data from Queue, data= 1576740829, producer id= 5 (Message timestamp: [5487 microseconds])

Inside Consumer Process --> Consumer with process number 6 exiting

Inside Producer Process --> Producer created with id: 4

Number of consumers exited 2

Inside Producer Process --> Producer with process number (assigned as producer id) 1 produced data (Data= 1812720505) for consumer process number (assigned as consumer id) 8 as it received a PRODUCE request message (Message timestamp: [5602 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 8 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 8, hence consumed data from Queue, data= 1812720505, producer id= 1 (Message timestamp: [5602 microseconds])

Inside Consumer Process --> Consumer with process number 8 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 4 produced data (Data= 1067398392) for consumer process number (assigned as consumer id) 1 as it received a PRODUCE request message (Message timestamp: [162758 microseconds])

Inside Producer Process --> Producer with process number (assigned as producer id) 4 produced data (Data= 1774275636) for consumer process number (assigned as consumer id) 8 as it received a PRODUCE request message (Message timestamp: [162798 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 8 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 8, hence consumed data from Queue, data= 1774275636, producer id= 4 (Message timestamp: [162798 microseconds])

Inside Consumer Process --> Consumer with process number 8 exiting

Number of consumers exited 3

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 1 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 1, hence consumed data from Queue, data= 1067398392, producer id= 4 (Message timestamp: [162758 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 5 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 5, hence consumed data from Queue, data= 499724341, producer id= 1 (Message timestamp: [5009 microseconds])

Inside Consumer Process --> Consumer with process number 1 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 1 produced data (Data= 265683174) for consumer process number (assigned as consumer id) 1 as it received a PRODUCE request message (Message timestamp: [163101 microseconds])

Inside Consumer Process --> Consumer with process number 5 sent request to producers

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 2 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 2, hence consumed data from Queue, data= 126592119, producer id= 5 (Message timestamp: [4771 microseconds])

Inside Consumer Process --> Consumer with process number 2 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 1 produced data (Data= 940148457) for consumer process number (assigned as consumer id) 5 as it received a PRODUCE request message (Message timestamp: [163263 microseconds])

Inside Producer Process --> Producer with process number (assigned as producer id) 5 produced data (Data= 499724341) for consumer process number (assigned as consumer id) 2 as it received a PRODUCE request message (Message timestamp: [163285 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 2 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 2, hence consumed data from Queue, data= 499724341, producer id= 5 (Message timestamp: [163285 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 5 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 5, hence consumed data from Queue, data= 940148457, producer id= 1 (Message timestamp: [163263 microseconds])

Inside Consumer Process --> Consumer with process number 2 exiting

Number of consumers exited 4

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 1 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 1, hence consumed data from Queue, data= 265683174, producer id= 1 (Message timestamp: [163101 microseconds])

Inside Consumer Process --> Consumer with process number 1 exiting

Number of consumers exited 5

Inside Consumer Process --> Consumer with process number 5 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 5 produced data (Data= 1812720505) for consumer process number (assigned as consumer id) 5 as it received a PRODUCE request message (Message timestamp: [163629 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 5 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 5, hence consumed data from Queue, data= 1812720505, producer id= 5 (Message timestamp: [163629 microseconds])

Inside Consumer Process --> Consumer with process number 5 exiting

Number of consumers exited 6

Inside Producer Process --> Producer created with id: 6

Inside Consumer Process --> Consumer created with id: 3

Inside Consumer Process --> Consumer created with id: 4

Inside Consumer Process --> Consumer with process number 3 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 2 produced data (Data= 265683174) for consumer process number (assigned as consumer id) 3 as it received a PRODUCE request message (Message timestamp: [165576 microseconds])

Inside Consumer Process --> Consumer with process number 4 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 4 produced data (Data= 2099175287) for consumer process number (assigned as consumer id) 4 as it received a PRODUCE request message (Message timestamp: [165693 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 4 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 4, hence consumed data from Queue, data= 2099175287, producer id= 4 (Message timestamp: [165693 microseconds])

Inside Consumer Process --> Consumer with process number 4 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 3 produced data (Data= 499724341) for consumer process number (assigned as consumer id) 4 as it received a PRODUCE request message (Message timestamp: [165835 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 3 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 3, hence consumed data from Queue, data= 265683174, producer id= 2 (Message timestamp: [165576 microseconds])

Inside Consumer Process --> Consumer with process number 3 sent request to producers

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 4 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 4, hence consumed data from Queue, data= 499724341, producer id= 3 (Message timestamp: [165835 microseconds])

Inside Producer Process --> Producer with process number (assigned as producer id) 6 produced data (Data= 1067398392) for consumer process number (assigned as consumer id) 3 as it received a PRODUCE request message (Message timestamp: [166013 microseconds])

Inside Consumer Process --> Consumer with process number 4 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 6 produced data (Data= 1774275636) for consumer process number (assigned as consumer id) 4 as it received a PRODUCE request message (Message timestamp: [166235 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 3 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 3, hence consumed data from Queue, data= 1067398392, producer id= 6 (Message timestamp: [166013 microseconds])

Inside Consumer Process --> Consumer with process number 3 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 2 produced data (Data= 940148457) for consumer process number (assigned as consumer id) 3 as it received a PRODUCE request message (Message timestamp: [166372 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 4 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 4, hence consumed data from Queue, data= 1774275636, producer id= 6 (Message timestamp: [166235 microseconds])

Inside Consumer Process --> Consumer with process number 4 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 2 produced data (Data= 656698474) for consumer process number (assigned as consumer id) 4 as it received a PRODUCE request message (Message timestamp: [166553 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 3 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 3, hence consumed data from Queue, data= 940148457, producer id= 2 (Message timestamp: [166372 microseconds])

Inside Consumer Process --> Consumer with process number 3 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 6 produced data (Data= 2099175287) for consumer process number (assigned as consumer id) 3 as it received a PRODUCE request message (Message timestamp: [166779 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 4 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 4, hence consumed data from Queue, data= 656698474, producer id= 2 (Message timestamp: [166553 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 3 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 3, hence consumed data from Queue, data= 2099175287, producer id= 6 (Message timestamp: [166779 microseconds])

Inside Consumer Process --> Consumer with process number 4 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 5 produced data (Data= 265683174) for consumer process number (assigned as consumer id) 4 as it received a PRODUCE request message (Message timestamp: [166852 microseconds])

Inside Consumer Process --> Consumer with process number 3 sent request to producers

Inside Producer Process --> Producer with process number (assigned as producer id) 2 produced data (Data= 979597877) for consumer process number (assigned as consumer id) 3 as it received a PRODUCE request message (Message timestamp: [166969 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 3 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 3, hence consumed data from Queue, data= 979597877, producer id= 2 (Message timestamp: [166969 microseconds])

Inside Consumer Process --> Consumer with process number (assigned as consumer id) 4 received an ACKNOWLEDGEMENT MESSAGE with the same consumer id 4, hence consumed data from Queue, data= 265683174, producer id= 5 (Message timestamp: [166852 microseconds])

Inside Consumer Process --> Consumer with process number 3 exiting

Inside Consumer Process --> Consumer with process number 4 exiting

Number of consumers exited 7

Number of consumers exited 8

Inside Producer Process --> Producer with process number 6 exiting as all consumer work is done and all the consumer processes have exited

Inside Producer Process --> Producer with process number 1 exiting as all consumer work is done and all the consumer processes have exited

Inside Producer Process --> Producer with process number 2 exiting as all consumer work is done and all the consumer processes have exited

Inside Producer Process --> Producer with process number 4 exiting as all consumer work is done and all the consumer processes have exited

Inside Producer Process --> Producer with process number 5 exiting as all consumer work is done and all the consumer processes have exited

Inside Producer Process --> Producer with process number 3 exiting as all consumer work is done and all the consumer processes have exited

All producer and consumer processes terminated

Exiting the program