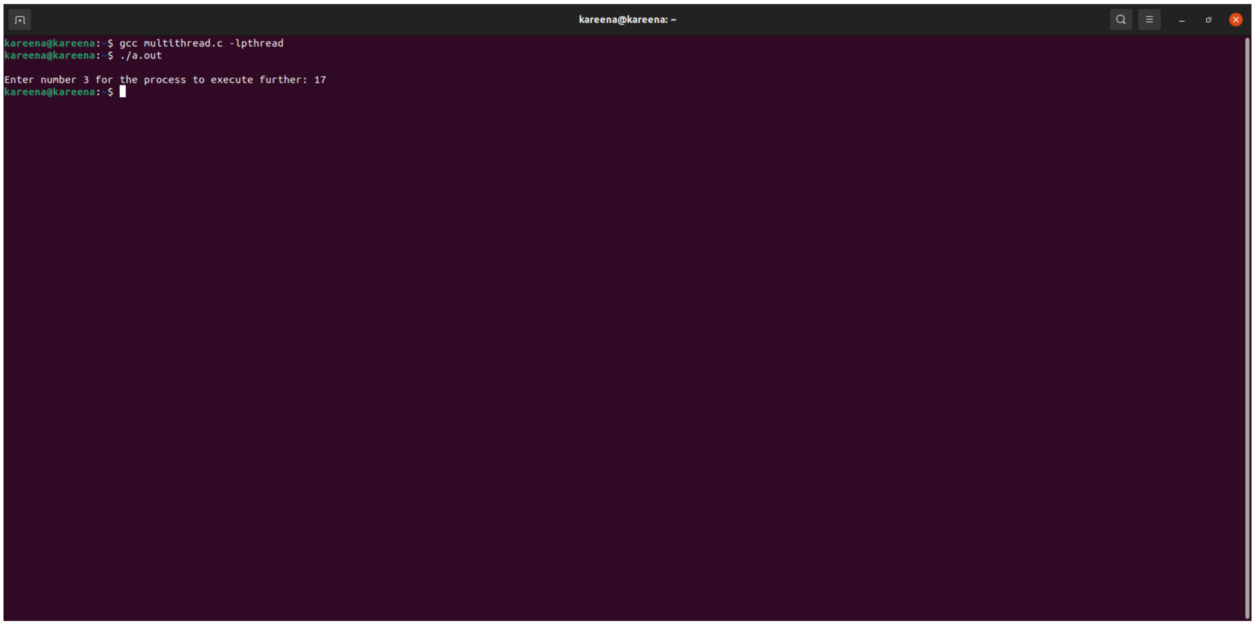
AU1940314
Kareena Matwani

END-SEMESTER PRACTICAL EXAM DESCRIPTION

Question1

Description : Using thread synchronization method the code is implemented in c , here the as we press 3 , 3 threads are created whereas accordingly the numbers get printed with the respect to the thread number entered.The different variables have their different variables and functions with their specific meanings
The turn variable tells you how the threads will enter the critical section for the execution, aslo count variables plays the same role in it , now the for loops are implemented for the printing of the respective number with the respective threads implementation.

Screenshots

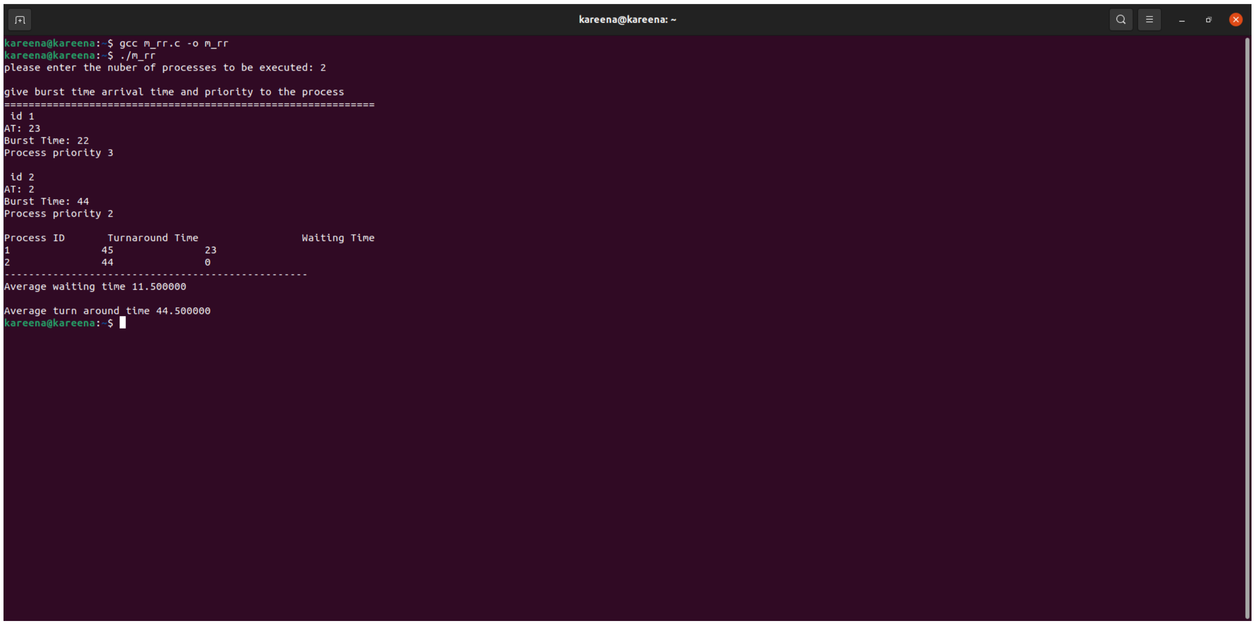


```
kareena@kareena:~$ gcc multithread.c -lpthread
kareena@kareena:~$ ./a.out

Enter number 3 for the process to execute further: 17
kareena@kareena:~$ 
```

Question2

Description : CPU scheduling plays an important roles while scheduling processes , so it is important to manage the TAT and Waiting time so there is a need to implement modified round robin scheduling because it leads to less context switching while also reducing the TAT and wait time of the processes executing, it also has higher efficiency than traditional method. Probably the time tum can fairly be reduced leading to the have better execution of multiple processes.

Screenshots



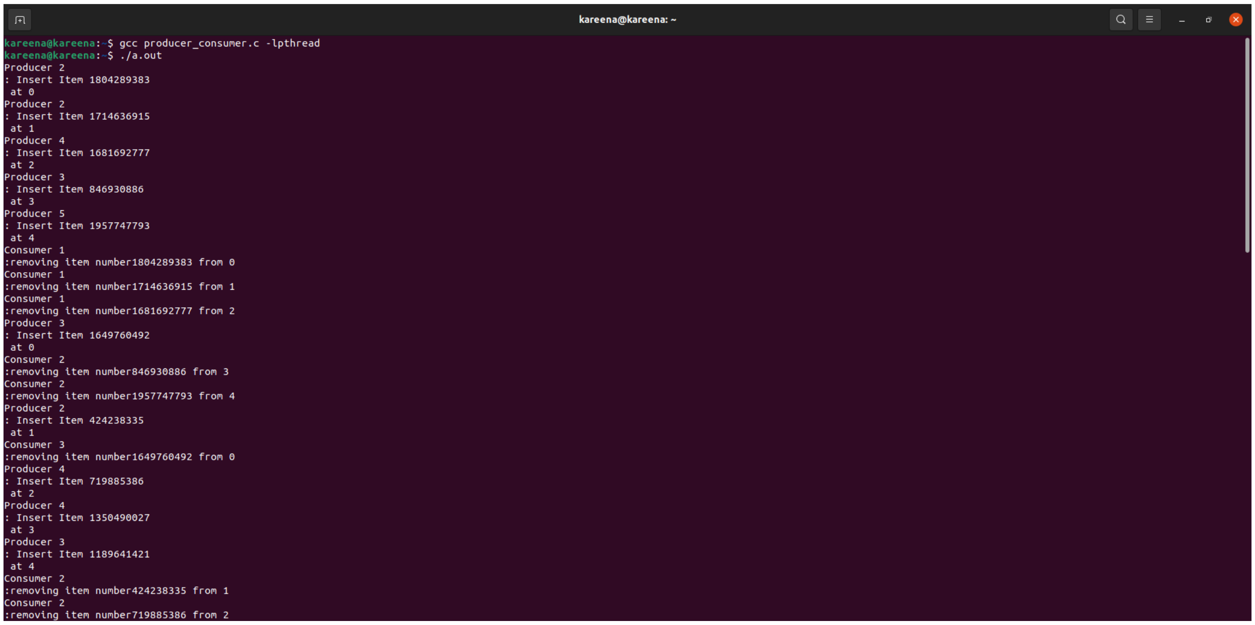
```
kareena@kareena:~$ gcc m_rr.c -o m_rr
kareena@kareena:~$ ./m_rr
please enter the nuber of processes to be executed: 2

give burst time arrival time and priority to the process
=============================================================
 id 1
AT: 23
Burst Time: 22
Process priority 3

 id 2
AT: 2
Burst Time: 44
Process priority 2

Process ID       Turnaround Time                 Waiting Time
1               45               23
2               44               0
--------------------------------------------------
Average waiting time 11.500000

Average turn around time 44.500000
kareena@kareena:~$ 
```

Question3

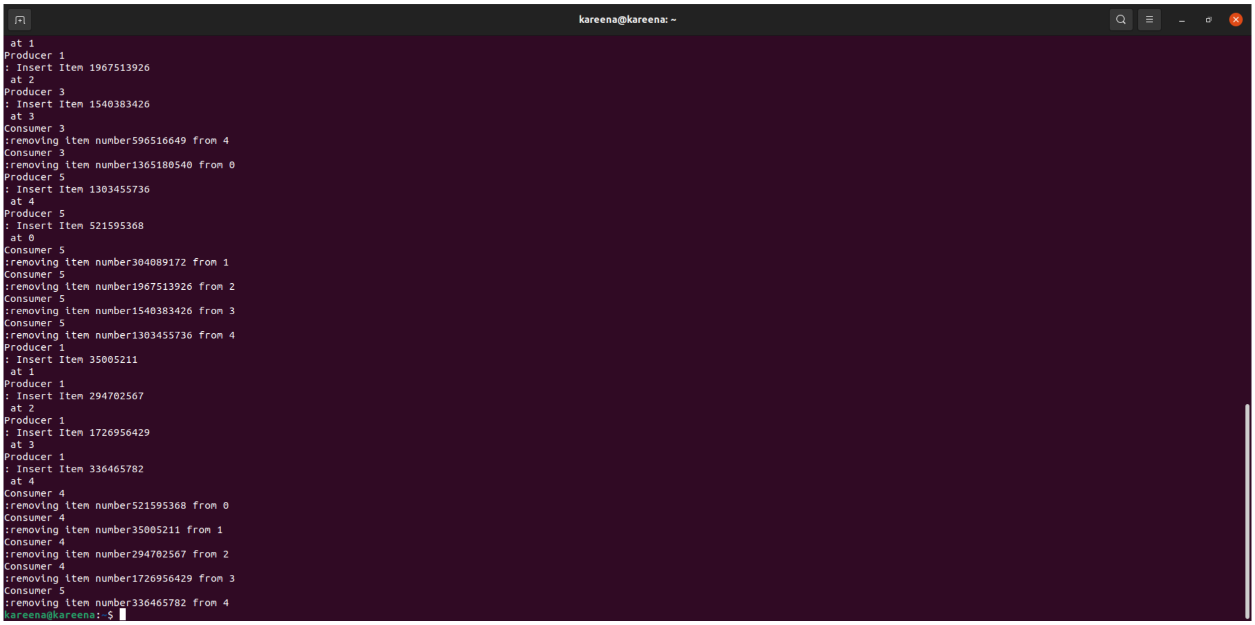
Description: Here in the producer consumer problem the logic goes behind this as follows in the implemented code 5 is the buffer size so as the semaphore is 1 the consumer understands that there is something that has been produced which has to be consumed also goes same with semaphore integer value is 0 it tells the the buffer is empty , there are specific conditions for the buffer as full and empty. Even if there is at least one space left in the buffer , the producer produces different items which have to be consumed.

Also , even if there is 1 item in the buffer it will have to be consumed by the consumer , thus both enter critical sections checking the values of semaphore and thus this happens with the implementation of mutex reaching mutual exclusion availability.

Screenshots



```
kareena@kareena:~$ gcc producer_consumer.c -lpthread
kareena@kareena:~$ ./a.out
Producer 2
: Insert Item 1804289383
 at 0
Producer 2
: Insert Item 1714636915
 at 1
Producer 4
: Insert Item 1681692777
 at 2
Producer 3
: Insert Item 846930886
 at 3
Producer 5
: Insert Item 1957747793
 at 4
Consumer 1
:removing item number1804289383 from 0
Consumer 1
:removing item number1714636915 from 1
Consumer 1
:removing item number1681692777 from 2
Producer 3
: Insert Item 1649760492
 at 0
Consumer 2
:removing item number846930886 from 3
Consumer 2
:removing item number1957747793 from 4
Producer 2
: Insert Item 424238335
 at 1
Consumer 3
:removing item number1649760492 from 0
Producer 4
: Insert Item 719885386
 at 2
Producer 4
: Insert Item 1350490027
 at 3
Producer 3
: Insert Item 1189641421
 at 4
Consumer 2
:removing item number424238335 from 1
Consumer 2
:removing item number719885386 from 2
```



```
 at 1
Producer 1
: Insert Item 1967513926
 at 2
Producer 3
: Insert Item 1540383426
 at 3
Consumer 3
:removing item number596516649 from 4
Consumer 3
:removing item number1365180540 from 0
Producer 5
: Insert Item 1303455736
 at 4
Producer 5
: Insert Item 521595368
 at 0
Consumer 5
:removing item number304089172 from 1
Consumer 5
:removing item number1967513926 from 2
Consumer 5
:removing item number1540383426 from 3
Consumer 5
:removing item number1303455736 from 4
Producer 1
: Insert Item 35005211
 at 1
Producer 1
: Insert Item 294702567
 at 2
Producer 1
: Insert Item 1726956429
 at 3
Producer 1
: Insert Item 336465782
 at 4
Consumer 4
:removing item number521595368 from 0
Consumer 4
:removing item number35005211 from 1
Consumer 4
:removing item number294702567 from 2
Consumer 4
:removing item number1726956429 from 3
Consumer 5
:removing item number336465782 from 4
kareena@kareena:~$ 
```