CSD204 - OS - Lab05

Lalit Maurya, 231010164

Question 01

```
~/d/s/CSD204-OS/lab05/src on  main !1 ?2
❯ make run q01
^Cmake: *** [Makefile:11: run] Interrupt

~/d/s/CSD204-OS/lab05/src on  main !1 ?2
❯ make run q01
P2 +83 [CAPACITY 1]
P3 +86 [CAPACITY 2]
P1 +77 [CAPACITY 3]
P3 +86 [CAPACITY 4]
P3 +49 [CAPACITY 5]
^Cmake: *** [Makefile:11: run] Interrupt

~/d/s/CSD204-OS/lab05/src on  main !1 ?2
❯ make run q01
P2 +83 [CAPACITY 1]
P3 +77 [CAPACITY 2]
P1 +86 [CAPACITY 3]
C1 -83 [CAPACITY 2]
C3 -77 [CAPACITY 1]
C2 -86 [CAPACITY 0]
P3 +21 [CAPACITY 1]
C3 -21 [CAPACITY 0]
P2 +90 [CAPACITY 1]
P1 +59 [CAPACITY 2]
C2 -90 [CAPACITY 1]
C1 -59 [CAPACITY 0]
P3 +72 [CAPACITY 1]
C2 -72 [CAPACITY 0]
P3 +68 [CAPACITY 1]
C3 -68 [CAPACITY 0]
^Cmake: *** [Makefile:11: run] Interrupt

~/d/s/CSD204-OS/lab05/src on  main ?2
❯
```

I have illustrated three cases in the attached screenshot.

1. **Case 1: No Producers, 3 Consumers**

In this case it is visible that no activity is observed. This is because consumers are blocked until there are items in the buffer

2. **Case 2: 3 Producers, no Consumers**

Here, we can see that producers produce 5 items and then stop all activity as they are blocked until the buffer has empty space.

3. **Case 3: 3 Producers, 3 Consumers**

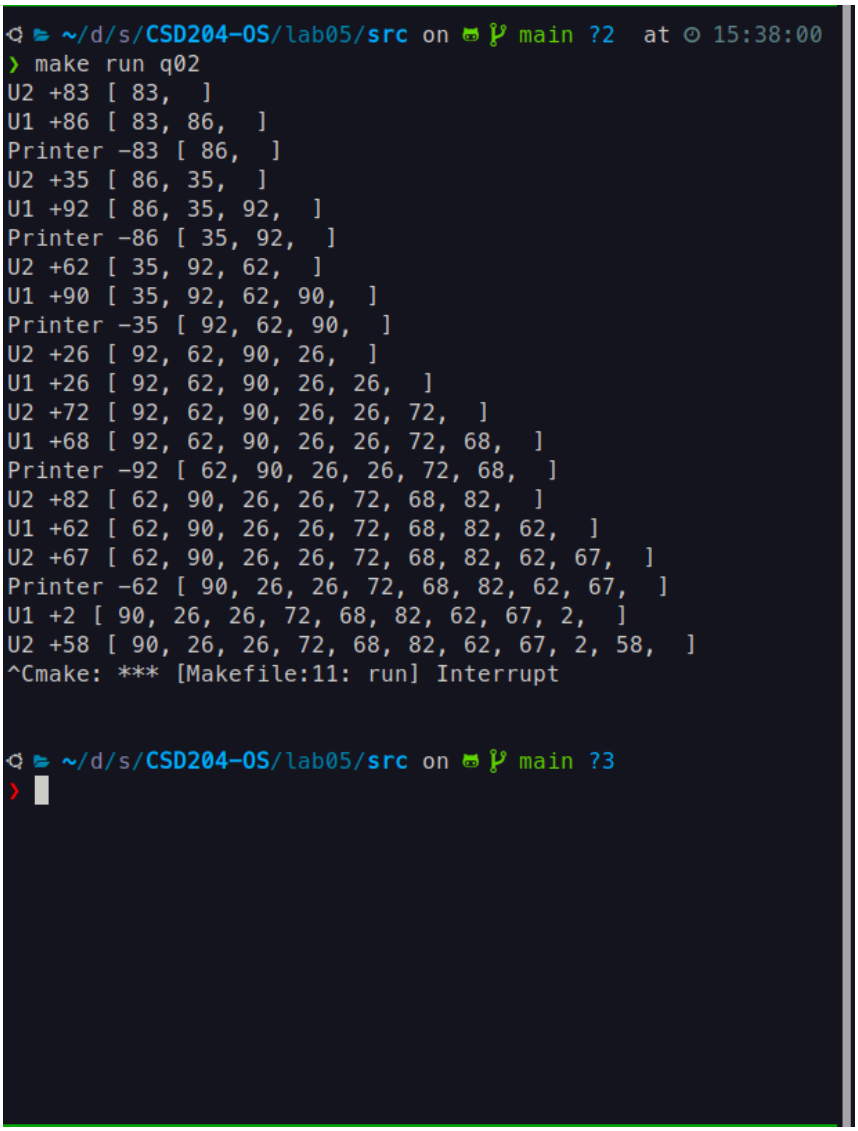
We see continous activity here as the buffer's contents are changing from second to second. Blocking occurs rarely as there is an equal presence of both consumers and producers.

The above is implemented using the following:

- A mutex (binary semaphore) that is used to ensure exclusive access to the buffer
- A counting semaphore that is used to signal production and consumption of items in the buffer

With the above, we can ensure fearless concurency in the producer consumer problem

Question 02



The printer spooler problem is solved via a similar approach to question 1. We Use:

- A mutex to ensure exclusive access to the spooler queue
- A counting semaphore to signal enqueuing and completion of printing jobs

Currently the program runs with only 1 printer thread and n user threads.