

SRN: PES1UG19EC326

NAME: Tanya Chanchalani

WEEK: 7

SUBJECT: OS

**//SHOWING RACE CONDITION**

```
#include <stdio.h>

#include <stdlib.h>

#include <stdbool.h>

#include <unistd.h>

#include <pthread.h>

#define BUFFER_SIZE 50

int start = 0, end = 0;

int *buffer;

int item = 0;

void *producer()

{

    while (true)

    {

        item += 1;

        printf("Producer: Produced Item: %d\n", item);

        sleep(1);

        while (((start + 1) % BUFFER_SIZE) == end)

            ;

        buffer[start] = item;

        start = (start + 1) % BUFFER_SIZE;

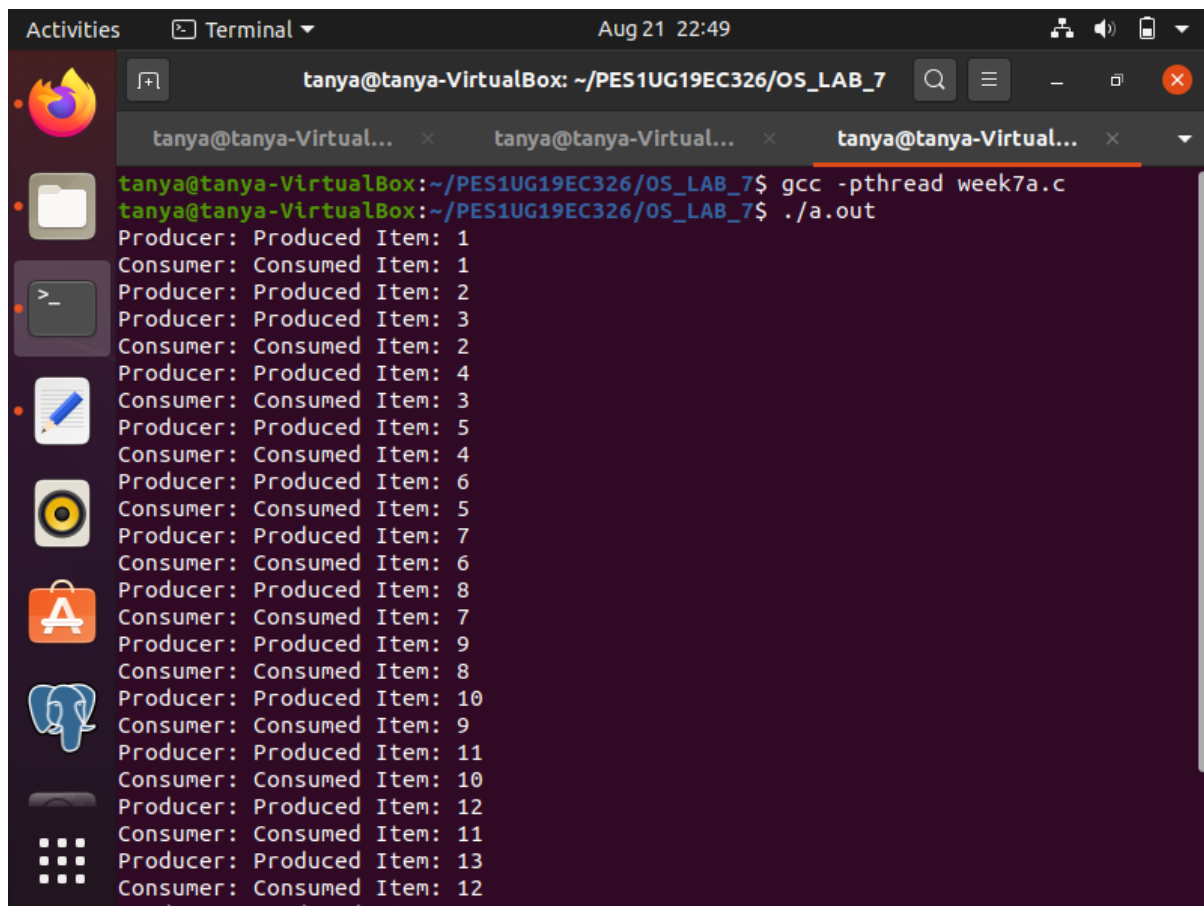
    }

}
```

```
void *consumer()
{
    while (true)
    {
        while (start == end)
            ;

        int consumed = buffer[end];
        printf("Consumer: Consumed Item: %d\n", consumed);
        sleep(1);
        end = (end + 1) % BUFFER_SIZE;
    }
}

int main()
{
    buffer = (int *)malloc(sizeof(int) * BUFFER_SIZE);
    pthread_t producer_thread, consumer_thread;
    pthread_create(&producer_thread, NULL, producer, NULL);
    pthread_create(&consumer_thread, NULL, consumer, NULL);
    pthread_join(producer_thread, NULL);
    pthread_join(consumer_thread, NULL);
    free(buffer);
    return 0;
}
```



```
tanya@tanya-VirtualBox: ~/PES1UG19EC326/OS_LAB_7
tanya@tanya-VirtualBox:~/PES1UG19EC326/OS_LAB_7$ gcc -pthread week7a.c
tanya@tanya-VirtualBox:~/PES1UG19EC326/OS_LAB_7$ ./a.out
Producer: Produced Item: 1
Consumer: Consumed Item: 1
Producer: Produced Item: 2
Producer: Produced Item: 3
Consumer: Consumed Item: 2
Producer: Produced Item: 4
Consumer: Consumed Item: 3
Producer: Produced Item: 5
Consumer: Consumed Item: 4
Producer: Produced Item: 6
Consumer: Consumed Item: 5
Producer: Produced Item: 7
Consumer: Consumed Item: 6
Producer: Produced Item: 8
Consumer: Consumed Item: 7
Producer: Produced Item: 9
Consumer: Consumed Item: 8
Producer: Produced Item: 10
Consumer: Consumed Item: 9
Producer: Produced Item: 11
Consumer: Consumed Item: 10
Producer: Produced Item: 12
Consumer: Consumed Item: 11
Producer: Produced Item: 13
Consumer: Consumed Item: 12
```

### //WITHOUT RACE CONDITION

```
#include <stdio.h>

#include <stdlib.h>

#include <stdbool.h>

#include <unistd.h>

#include <pthread.h>

#define BUFFER_SIZE 50

void *producer();

void *consumer();

int *buffer;

int start = 0, end = 0;

pthread_mutex_t mutex = PTHREAD_MUTEX_INITIALIZER;
```

```

pthread_mutex_t empty = PTHREAD_MUTEX_INITIALIZER;
pthread_mutex_t full = PTHREAD_MUTEX_INITIALIZER;
void *producer()
{
    int item = 0;
    while (true)
    {
        pthread_mutex_lock(&empty);
        pthread_mutex_lock(&mutex);
        item += 1;
        printf("Producer: Produced Item: %d\n", item);
        buffer[start] = item;
        pthread_mutex_unlock(&mutex);
        pthread_mutex_unlock(&full);
        start = (start + 1) % BUFFER_SIZE;
    }
}
void *consumer()
{
    while (true)
    {
        pthread_mutex_lock(&full);
        pthread_mutex_lock(&mutex);
        int consumed = buffer[end];
        printf("Consumer: Consumed Item: %d\n", consumed);
        sleep(1);
        end = (end + 1) % BUFFER_SIZE;
        pthread_mutex_unlock(&mutex);
        pthread_mutex_unlock(&empty);
    }
}

```

```

int main()
{
    buffer = (int *)malloc(sizeof(int) * BUFFER_SIZE);
    pthread_t producer_thread, consumer_thread;
    pthread_create(&producer_thread, NULL, producer, NULL);
    sleep(1);
    pthread_create(&consumer_thread, NULL, consumer, NULL);
    pthread_join(producer_thread, NULL);
    pthread_join(consumer_thread, NULL);
    free(buffer);
    return 0;
}

```

The screenshot shows a terminal window titled "tanya@tanya-VirtualBox: ~/PES1UG19EC326/OS\_LAB\_7". The terminal displays the following output:

```

tanya@tanya-VirtualBox:~/PES1UG19EC326/OS_LAB_7$ gcc -pthread week7.c
tanya@tanya-VirtualBox:~/PES1UG19EC326/OS_LAB_7$ ./a.out
Producer: Produced Item: 1
Consumer: Consumed Item: 1
Producer: Produced Item: 2
Consumer: Consumed Item: 2
Producer: Produced Item: 3
Consumer: Consumed Item: 3
Producer: Produced Item: 4
Consumer: Consumed Item: 4
Producer: Produced Item: 5
Consumer: Consumed Item: 5
Producer: Produced Item: 6
Consumer: Consumed Item: 6
Producer: Produced Item: 7
Consumer: Consumed Item: 7
Producer: Produced Item: 8
Consumer: Consumed Item: 8
Producer: Produced Item: 9
Consumer: Consumed Item: 9
Producer: Produced Item: 10
Consumer: Consumed Item: 10
Producer: Produced Item: 11
Consumer: Consumed Item: 11
Producer: Produced Item: 12
Consumer: Consumed Item: 12
^C

```