**Operating System Lab Assignment-3**

Name: Aastha Jain

Batch: 2

Class: CS-B

PRN- 12211635

**Producer Consumer**

**Code:**

#!/bin/bash

BUFFER\_SIZE=10

NUM\_ITEMS=20

BUFFER=()

INDEX=0

produce\_item() {

local item=$RANDOM

BUFFER[$INDEX]=$item

INDEX=$(( (INDEX + 1) % BUFFER\_SIZE ))

echo "Produced item: $item"

}

consume\_item() {

local item=${BUFFER[INDEX]}

BUFFER[$INDEX]=""

INDEX=$(( (INDEX + 1) % BUFFER\_SIZE ))

echo "Consumed item: $item"

}

producer() {

for ((i = 0; i < NUM\_ITEMS; i++)); do

produce\_item

done

}

consumer() {

for ((i = 0; i < NUM\_ITEMS; i++)); do

consume\_item

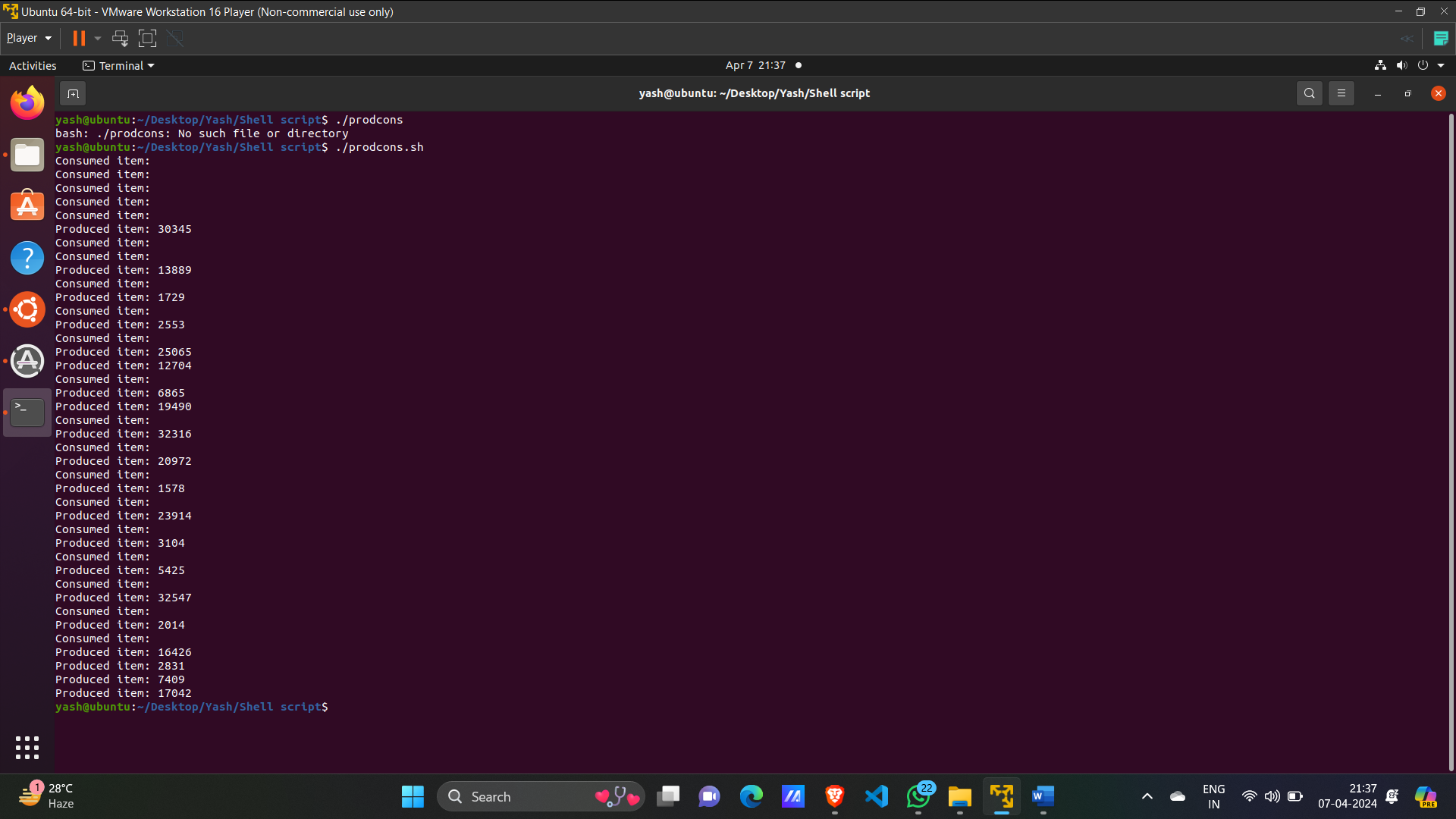
done

}

producer & consumer

wait

Output



#include <stdio.h>

#include <stdlib.h>

int mutex = 1;

int full = 0;

int empty = 10, x = 0;

void producer()

{

    --mutex;

    ++full;

    --empty;

    x++;

    printf("\nProducer produces"

           "item %d",

           x);

    ++mutex;

}

void consumer()

{

    --mutex;

    --full;

    ++empty;

    printf("\nConsumer consumes "

           "item %d",

           x);

    x--;

    ++mutex;

}

int main()

{

    int n, i;

    printf("\n1. Press 1 for Producer"

           "\n2. Press 2 for Consumer"

           "\n3. Press 3 for Exit");

#pragma omp critical

    for (i = 1; i > 0; i++) {

        printf("\nEnter your choice:");

        scanf("%d", &n);

        switch (n) {

        case 1:

                && (empty != 0)) {

                producer();

            }

            else {

                printf("Buffer is full!");

            }

            break;

        case 2:

            if ((mutex == 1)

                && (full != 0)) {

                consumer();

            }

            else {

                printf("Buffer is empty!");

            }

            break;

        case 3:

            exit(0);

            break;

        }

    }

}