Name – Harshit

MIS – 112316018

OS lab

Assignment 6

To write a C-program to implement the producer – consumer problem using semaphores.

Code:  
#include <stdio.h>

#include <stdlib.h>

#include <pthread.h>

#include <semaphore.h>

#include <unistd.h>

#define BUFFER\_SIZE 5

int buffer[BUFFER\_SIZE];

int in = 0, out = 0;

sem\_t empty;

sem\_t full;

pthread\_mutex\_t mutex;

void \*producer(void \*arg) {

    int item;

    for (int i = 0; i < 10; i++) {

        item = rand() % 100;

        sem\_wait(&empty);

        pthread\_mutex\_lock(&mutex);

        buffer[in] = item;

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

        in = (in + 1) % BUFFER\_SIZE;

        pthread\_mutex\_unlock(&mutex);

        sem\_post(&full);

        sleep(1);

    }

    return NULL;

}

void \*consumer(void \*arg) {

    for (int i = 0; i < 10; i++) {

        sem\_wait(&full);

        pthread\_mutex\_lock(&mutex);

        int item = buffer[out];

        printf("Consumer consumed: %d\n", item);

        out = (out + 1) % BUFFER\_SIZE;

        pthread\_mutex\_unlock(&mutex);

        sem\_post(&empty);

        sleep(2);

    }

    return NULL;

}

int main() {

    pthread\_t prod, cons;

    sem\_init(&empty, 0, BUFFER\_SIZE);

    sem\_init(&full, 0, 0);

    pthread\_mutex\_init(&mutex, NULL);

    pthread\_create(&prod, NULL, producer, NULL);

    pthread\_create(&cons, NULL, consumer, NULL);

    pthread\_join(prod, NULL);

    pthread\_join(cons, NULL);

    sem\_destroy(&empty);

    sem\_destroy(&full);

    pthread\_mutex\_destroy(&mutex);

    return 0;

}

Output:

 