Akash Dansinghani
Homework 3
Operating Systems
Professor Yan
03/03/2021

Code:

```c

#include "buffer.h"
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
#include <semaphore.h>
#include <stdbool.h>
#include <unistd.h>
#include <string.h>

buffer_item buffer[BUFFER_SIZE];

//functions to initalize the pthreads data types and semaphores.
pthread_mutex_t mutex;

//semaphore to indicate that the entry from the buffer is null or empty
sem_t emptySem;
//semaphore to indicate that the entry from the buffer is filled
sem_t fullSem;
int left = -1;
int right = 0;
int one = 1;
void *producer(void *param);
void *consumer(void *param);

const char* buffer_to_string() {

    char output[BUFFER_SIZE * 3];

    for (int a = 0; a < BUFFER_SIZE; a++) {

        if (buffer[a] != 0) {
            output[a * 3 + one] = buffer[a] + '0';
        } else {
            output[a * 3 + one] = '_';
        }
```

```
            output[a * 3 + (one * 2)] = ',';
            output[a * 3 + (one * 3)] = ' ';
        }

    }

    output[0] = '[';
    output[BUFFER_SIZE * 3 - one] = ']';

    char *resPtr = output;
    return resPtr;


}

/* Insert item into buffer */
int insert(buffer_item item) {
/* Acquire mutex lock to protect buffer */
    sem_wait(&emptySem);
    pthread_mutex_lock(&mutex);

    int index = -1;

    if (left + 1 == right) {

        index = 0;
        buffer[0] = item;
        left = -1;
        right = 1;

    } else if (left >= BUFFER_SIZE - right && left > -1 || left > 0) {

        index = left;
        buffer[left] = item;
        left--;

    } else if (right < BUFFER_SIZE) {
```

```
        index = right;
        buffer[right] = item;
        right++;

    } else {
        return -1;
    }

    printf("Producer %u produced %d at position %d\t", (unsigned int) pthread_self(), item, index);
    printf("%s", buffer_to_string());
    printf("\n");
    /* Release mutex lock and full semaphore */
    pthread_mutex_unlock(&mutex);
    sem_post(&fullSem);

    return 0;
}

/* remove an object from buffer placing it in item */
int remove_item(buffer_item *item) {
/* Acquire mutex lock to protect buffer */
    sem_wait(&fullSem);
    pthread_mutex_lock(&mutex);

    int index= -1;

    if (left < BUFFER_SIZE - right && left > -2) {

        index = left + 1;
        *item = buffer[left + 1];
        buffer[left + 1] = 0;
        left++;

    } else if (right <= BUFFER_SIZE) {

        index = right - 1;
        *item = buffer[right - 1];
```

```
        buffer[right - 1] = 0;
        right--;

    } else {
        return -1;
    }

    printf("Consumer %u consumed %d at position %d\t", (unsigned int) pthread_self(), *item, index);
    printf("%s", buffer_to_string());
    printf("\n");
    /* Release mutex lock and empty semaphore */
    pthread_mutex_unlock(&mutex);
    sem_post(&emptySem);

    return 0;
}


int main(int argc, char *argv[]) {

    if (argc != 4) {
        fprintf(stderr, "Usage: %s <sleep> <# of producers> <# of consumers>\n", argv[0]);
        return -1;
    }
/* Get command line arguments argv[1],argv[2],argv[3] */
// intialize sleep which will measure the time the function is idle or sleep
    int sleepDuration = atoi(argv[1]);
    int producer_count = atoi(argv[2]);
    int consumer_count = atoi(argv[3]);

    if (producer_count < 2 || producer_count == 0) {
        fprintf(stderr, "Error\n");
        return -1;
    }

    if (consumer_count < 2 || consumer_count == 0) {
        fprintf(stderr, "Error\n");
```

```
        return -1;
    }
    /* Initialize buffer related synchronization tools */
    // set a seed for the starting point for the random number generation
    srand(time(NULL));
    void *consumer(void *param);
    pthread_mutex_init(&mutex, NULL);

    // Create the empty semaphore and intialize it to BUFFER_SIZE
    sem_init(&emptySem, 0, BUFFER_SIZE);
    // Used to create the full semaphore and initialize it to 0
    sem_init(&fullSem, 0, 0);

    pthread_attr_t attr;
    pthread_attr_init(&attr);
    /* Create producer threads based on the command line input */
    pthread_t producers[producer_count];
    /* Create consumer threads based on the command line input */
    pthread_t consumers[consumer_count];
    /* Sleep for user specified time based on the command line input */
    for (int a = 0; a < producer_count; a++) {
        pthread_create(&producers[a], &attr, producer, NULL);
    }

    for (int c = 0; c < consumer_count; c++) {
        pthread_create(&consumers[c], &attr, consumer, NULL);
    }

    sleep(sleepDuration);

    return 0;
}

void *producer(void *param) {
    /* producer thread that calls insert_item() */
    buffer_item item;
    int ten = 10;
```

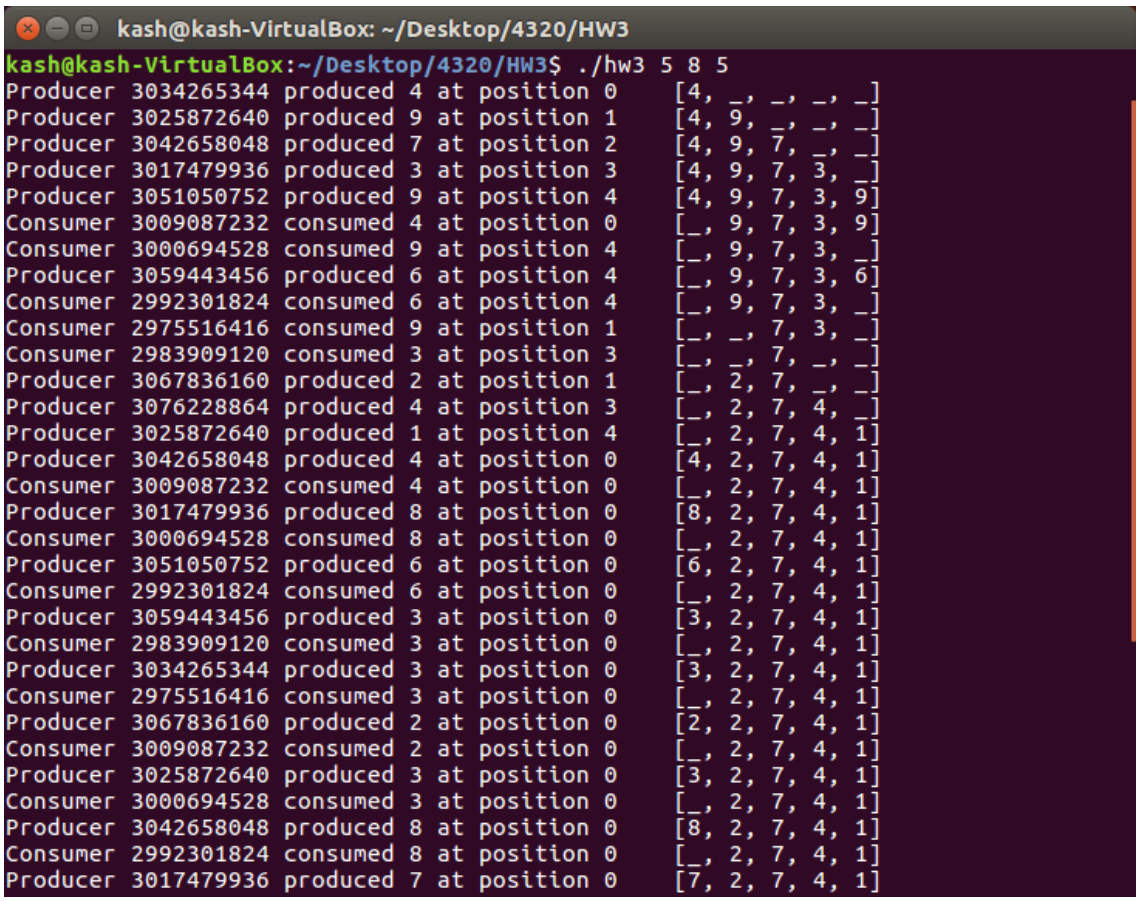
```
    int ten = 10;

    while (true) {

        sleep(1);

        item = rand() % ten; // the item produced would be the first ouput

        if (insert(item)) {
            fprintf(stderr, "Error\n");
        }

    }

}

void *consumer(void *param) {
/* consumer thread that calls remove_item() */
    buffer_item item;

    while (true) {

        sleep(1);

        if (remove_item(&item)) {
            fprintf(stderr, "Error\n");
        }


    }

}
//end of main function
```

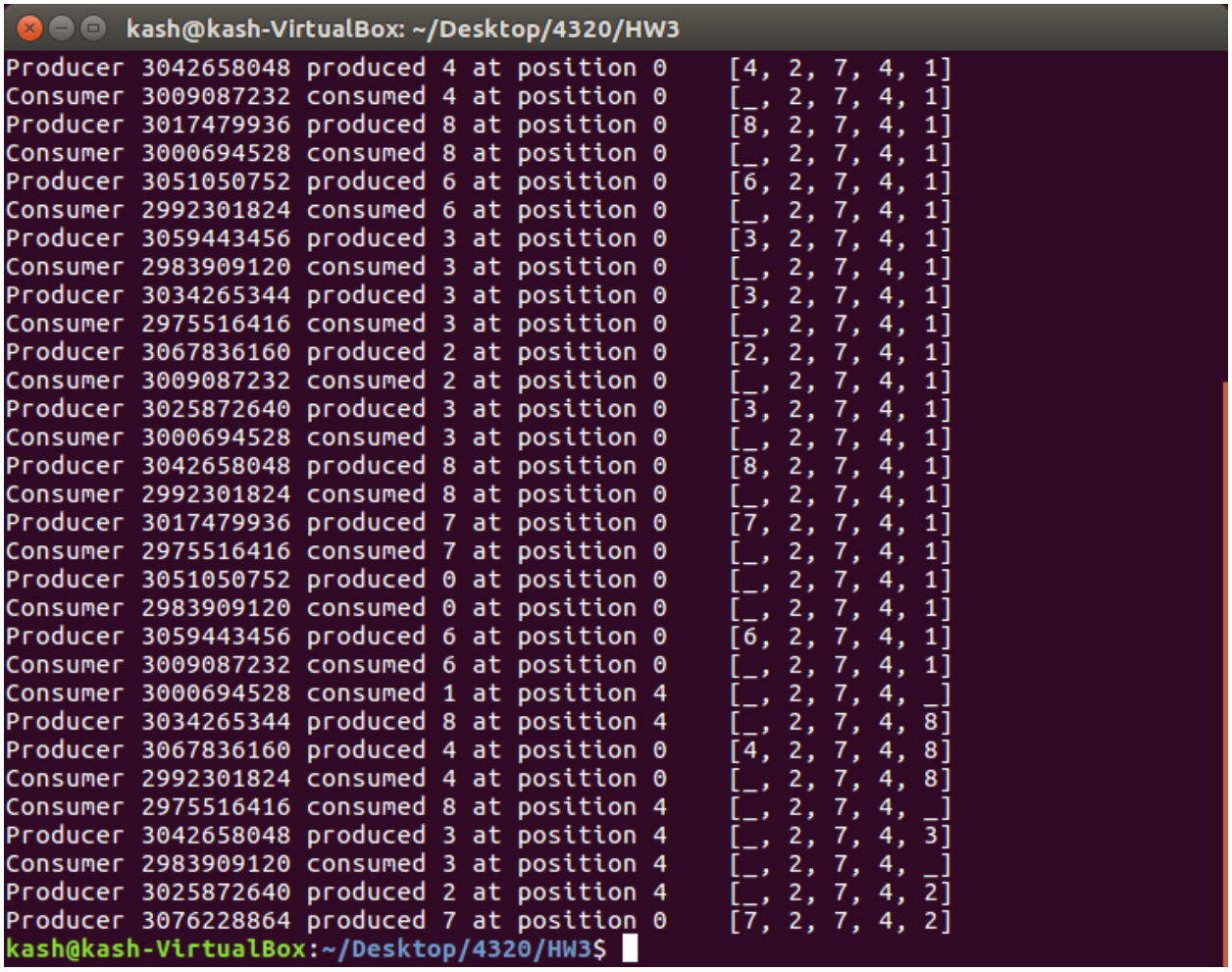
Output first time:

```
kash@kash-VirtualBox:~$ cd Desktop/
kash@kash-VirtualBox:~/Desktop$ cd 4320/
kash@kash-VirtualBox:~/Desktop/4320$ cd HW3/
kash@kash-VirtualBox:~/Desktop/4320/HW3$ gcc -pthread -o hw3 hw3.c
kash@kash-VirtualBox:~/Desktop/4320/HW3$ ls
buffer.h  hw3  hw3.c
kash@kash-VirtualBox:~/Desktop/4320/HW3$ ./hw3 5 8 5
Producer 3543025408 produced 9 at position 0    [9, _, _, _, _]
Producer 3551418112 produced 6 at position 1    [9, 6, _, _, _]
Producer 3534632704 produced 7 at position 2    [9, 6, 7, _, _]
Producer 3559810816 produced 8 at position 3    [9, 6, 7, 8, _]
Consumer 3526240000 consumed 9 at position 0    [_, 6, 7, 8, _]
Producer 3568203520 produced 7 at position 4    [_, 6, 7, 8, 7]
Consumer 3517847296 consumed 7 at position 4    [_, 6, 7, 8, _]
Consumer 3509454592 consumed 6 at position 1    [_, _, 7, 8, _]
Consumer 3501061888 consumed 8 at position 3    [_, _, 7, _, _]
Producer 3576596224 produced 9 at position 3    [_, _, 7, 9, _]
Consumer 3492669184 consumed 9 at position 3    [_, _, 7, _, _]
Producer 3584988928 produced 9 at position 3    [_, _, 7, 9, _]
Producer 3593381632 produced 4 at position 1    [_, 4, 7, 9, _]
Producer 3551418112 produced 1 at position 4    [_, 4, 7, 9, 1]
Producer 3534632704 produced 8 at position 0    [8, 4, 7, 9, 1]
Consumer 3526240000 consumed 8 at position 0    [_, 4, 7, 9, 1]
Producer 3559810816 produced 6 at position 0    [6, 4, 7, 9, 1]
Consumer 3517847296 consumed 6 at position 0    [_, 4, 7, 9, 1]
Producer 3568203520 produced 2 at position 0    [2, 4, 7, 9, 1]
Consumer 3509454592 consumed 2 at position 0    [_, 4, 7, 9, 1]
Consumer 3501061888 consumed 1 at position 4    [_, 4, 7, 9, _]
Producer 3576596224 produced 2 at position 4    [_, 4, 7, 9, 2]
Consumer 3492669184 consumed 2 at position 4    [_, 4, 7, 9, _]
Producer 3543025408 produced 3 at position 4    [_, 4, 7, 9, 3]
Producer 3584988928 produced 2 at position 0    [2, 4, 7, 9, 3]
Consumer 3526240000 consumed 2 at position 0    [_, 4, 7, 9, 3]
```

```
Producer 3551418112 produced 1 at position 4    [_, 4, 7, 9, 1]
Producer 3534632704 produced 8 at position 0    [8, 4, 7, 9, 1]
Consumer 3526240000 consumed 8 at position 0    [_, 4, 7, 9, 1]
Producer 3559810816 produced 6 at position 0    [6, 4, 7, 9, 1]
Consumer 3517847296 consumed 6 at position 0    [_, 4, 7, 9, 1]
Producer 3568203520 produced 2 at position 0    [2, 4, 7, 9, 1]
Consumer 3509454592 consumed 2 at position 0    [_, 4, 7, 9, 1]
Consumer 3501061888 consumed 1 at position 4    [_, 4, 7, 9, _]
Producer 3576596224 produced 2 at position 4    [_, 4, 7, 9, 2]
Consumer 3492669184 consumed 2 at position 4    [_, 4, 7, 9, _]
Producer 3543025408 produced 3 at position 4    [_, 4, 7, 9, 3]
Producer 3584988928 produced 2 at position 0    [2, 4, 7, 9, 3]
Consumer 3526240000 consumed 2 at position 0    [_, 4, 7, 9, 3]
Producer 3568203520 produced 1 at position 0    [1, 4, 7, 9, 3]
Consumer 3517847296 consumed 1 at position 0    [_, 4, 7, 9, 3]
Producer 3534632704 produced 1 at position 0    [1, 4, 7, 9, 3]
Consumer 3509454592 consumed 1 at position 0    [_, 4, 7, 9, 3]
Producer 3559810816 produced 6 at position 0    [6, 4, 7, 9, 3]
Consumer 3501061888 consumed 6 at position 0    [_, 4, 7, 9, 3]
Producer 3551418112 produced 1 at position 0    [1, 4, 7, 9, 3]
Consumer 3492669184 consumed 1 at position 0    [_, 4, 7, 9, 3]
Producer 3576596224 produced 7 at position 0    [7, 4, 7, 9, 3]
Consumer 3526240000 consumed 7 at position 0    [_, 4, 7, 9, 3]
Consumer 3517847296 consumed 3 at position 4    [_, 4, 7, 9, _]
Consumer 3509454592 consumed 4 at position 1    [_, _, 7, 9, _]
Consumer 3501061888 consumed 9 at position 3    [_, _, 7, _, _]
Producer 3559810816 produced 2 at position 3    [_, _, 7, 2, _]
Producer 3576596224 produced 3 at position 1    [_, 3, 7, 2, _]
Consumer 3492669184 consumed 3 at position 1    [_, _, 7, 2, _]
Producer 3534632704 produced 2 at position 1    [_, 2, 7, 2, _]
Producer 3543025408 produced 9 at position 4    [_, 2, 7, 2, 9]
Producer 3584988928 produced 3 at position 0    [3, 2, 7, 2, 9]
kash@kash-VirtualBox:~/Desktop/4320/HW3$
```

Output second time:




kash@kash-VirtualBox: ~/Desktop/4320/HW3
Producer 3042658048 produced 4 at position 0 [4, 2, 7, 4, 1]
Consumer 3009087232 consumed 4 at position 0 [_, 2, 7, 4, 1]
Producer 3017479936 produced 8 at position 0 [8, 2, 7, 4, 1]
Consumer 3000694528 consumed 8 at position 0 [_, 2, 7, 4, 1]
Producer 3051050752 produced 6 at position 0 [6, 2, 7, 4, 1]
Consumer 2992301824 consumed 6 at position 0 [_, 2, 7, 4, 1]
Producer 3059443456 produced 3 at position 0 [3, 2, 7, 4, 1]
Consumer 2983909120 consumed 3 at position 0 [_, 2, 7, 4, 1]
Producer 3034265344 produced 3 at position 0 [3, 2, 7, 4, 1]
Consumer 2975516416 consumed 3 at position 0 [_, 2, 7, 4, 1]
Producer 3067836160 produced 2 at position 0 [2, 2, 7, 4, 1]
Consumer 3009087232 consumed 2 at position 0 [_, 2, 7, 4, 1]
Producer 3025872640 produced 3 at position 0 [3, 2, 7, 4, 1]
Consumer 3000694528 consumed 3 at position 0 [_, 2, 7, 4, 1]
Producer 3042658048 produced 8 at position 0 [8, 2, 7, 4, 1]
Consumer 2992301824 consumed 8 at position 0 [_, 2, 7, 4, 1]
Producer 3017479936 produced 7 at position 0 [7, 2, 7, 4, 1]
Consumer 2975516416 consumed 7 at position 0 [_, 2, 7, 4, 1]
Producer 3051050752 produced 0 at position 0 [_, 2, 7, 4, 1]
Consumer 2983909120 consumed 0 at position 0 [_, 2, 7, 4, 1]
Producer 3059443456 produced 6 at position 0 [6, 2, 7, 4, 1]
Consumer 3009087232 consumed 6 at position 0 [_, 2, 7, 4, 1]
Consumer 3000694528 consumed 1 at position 4 [_, 2, 7, 4, _]
Producer 3034265344 produced 8 at position 4 [_, 2, 7, 4, 8]
Producer 3067836160 produced 4 at position 0 [4, 2, 7, 4, 8]
Consumer 2992301824 consumed 4 at position 0 [_, 2, 7, 4, 8]
Consumer 2975516416 consumed 8 at position 4 [_, 2, 7, 4, _]
Producer 3042658048 produced 3 at position 4 [_, 2, 7, 4, 3]
Consumer 2983909120 consumed 3 at position 4 [_, 2, 7, 4, _]
Producer 3025872640 produced 2 at position 4 [_, 2, 7, 4, 2]
Producer 3076228864 produced 7 at position 0 [7, 2, 7, 4, 2]
kash@kash-VirtualBox:~/Desktop/4320/HW3$