

Operating System

LAB – 5

Name : **Om Santosh Gore**

Div : **AI – 'A'**

Roll No. : **56**

PRN : **12110548**

➤ Problem Statement : .

Implementation of Classical problem Reader Writer using Threads and Mutex

➤ Code :

```
➤ #include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
#include <unistd.h>

#define NUM_READERS 3
#define NUM_WRITERS 2

pthread_mutex_t mutex = PTHREAD_MUTEX_INITIALIZER;
pthread_mutex_t rw_mutex = PTHREAD_MUTEX_INITIALIZER;
int readers_count = 0;

void *reader(void *arg) {
    int reader_id = *((int *)arg);

    while (1) {
        pthread_mutex_lock(&mutex);
        readers_count++;
        if (readers_count == 1) {
            pthread_mutex_lock(&rw_mutex);
        }
        pthread_mutex_unlock(&mutex);

        // Reading
        printf("Reader %d is reading...\n", reader_id);
        sleep(1);

        pthread_mutex_lock(&mutex);
        readers_count--;
        if (readers_count == 0) {
            pthread_mutex_unlock(&rw_mutex);
        }
        pthread_mutex_unlock(&mutex);

        sleep(2);
    }
    return NULL;
}

void *writer(void *arg) {
    int writer_id = *((int *)arg);

    while (1) {
```

```

        pthread_mutex_lock(&rw_mutex);

        // Writing
        printf("Writer %d is writing...\n", writer_id);
        sleep(2);

        pthread_mutex_unlock(&rw_mutex);

        sleep(1);
    }
    return NULL;
}

int main() {
    pthread_t reader_threads[NUM_READERS];
    pthread_t writer_threads[NUM_WRITERS];
    int reader_ids[NUM_READERS];
    int writer_ids[NUM_WRITERS];

    for (int i = 0; i < NUM_READERS; i++) {
        reader_ids[i] = i + 1;
        pthread_create(&reader_threads[i], NULL, reader, &reader_ids[i]);
    }

    for (int i = 0; i < NUM_WRITERS; i++) {
        writer_ids[i] = i + 1;
        pthread_create(&writer_threads[i], NULL, writer, &writer_ids[i]);
    }

    for (int i = 0; i < NUM_READERS; i++) {
        pthread_join(reader_threads[i], NULL);
    }
    for (int i = 0; i < NUM_WRITERS; i++) {
        pthread_join(writer_threads[i], NULL);
    }

    return 0;
}

```

➤ Output :

```

omi@LIGHT: ~/Desktop
omi@LIGHT:~/Desktop$ ./reader_writer
Reader 2 is reading...
Reader 1 is reading...
Reader 3 is reading...
Writer 1 is writing...
Writer 2 is writing...
Reader 3 is reading...
Reader 2 is reading...
Reader 1 is reading...
Writer 1 is writing...
Writer 2 is writing...
Reader 1 is reading...
Reader 3 is reading...
Reader 2 is reading...
Writer 1 is writing...
Writer 2 is writing...
^C
omi@LIGHT:~/Desktop$

```