Task 6: Write a Program to Implementation of Classical problems using Threads and Mutex (input Takes by user).

Here's an example of a simple program that uses threads and mutex for a basic problem, like incrementing a shared variable safely:

```
#include <stdio.h>
#include <pthread.h>

#define NUM_THREADS 3

int counter = 0;
pthread_mutex_t mutex = PTHREAD_MUTEX_INITIALIZER;

void *incrementCounter(void *arg) {
    pthread_mutex_lock(&mutex);

    int thread_id = *((int *)arg);
    printf("Thread %d: Incrementing counter.\n", thread_id);

    counter++;

    printf("Thread %d: Counter value after increment: %d\n", thread_id, counter);

    pthread_mutex_unlock(&mutex);
    pthread_exit(NULL);
}

int main() {
    pthread_t threads[NUM_THREADS];
    int thread_ids[NUM_THREADS];

    for (int i = 0; i < NUM_THREADS; i++) {
```

```
        thread_ids[i] = i;
        pthread_create(&threads[i], NULL, incrementCounter, (void *)&thread_ids[i]);
    }

    for (int i = 0; i < NUM_THREADS; i++) {
        pthread_join(threads[i], NULL);
    }

    printf("Final counter value: %d\n", counter);

    return 0;
}
```

Output-