TITLE :

Write the C program to demonstrate mutual exclusion using mutex locks, solving the critical section problem.

NAME : Shinde Shubham Dnyandev,   **DIV :** SY-B,   **ROLL NO. :** 23107121.

PROGRAM :

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

int counter = 0;
pthread_mutex_t mutex;

void* increment_counter(void* arg)
{
    pthread_mutex_lock(&mutex);
    counter++;
    printf("Counter Value : %d \n", counter);

    pthread_mutex_unlock(&mutex);
    return NULL;
};

int main()
{
    if(pthread_mutex_init(&mutex, NULL) != 0)
    {
        printf("Mutex Initialization Failed");
```

```c
    return 1;
}


pthread_t thread1, thread2, thread3;


pthread_create(&thread1, NULL, increment_counter, NULL);

pthread_create(&thread2, NULL, increment_counter, NULL);

pthread_create(&thread3, NULL, increment_counter, NULL);


pthread_join(thread1, NULL);

pthread_join(thread2, NULL);

pthread_join(thread3, NULL);


pthread_mutex_destroy(&mutex);


return 0;
}
```

OUTPUT :

```
shubham@ShubhsPC:~$ gcc -pthread -o mutex threads.c
shubham@ShubhsPC:~$ ./mutex
Counter Value : 1
Counter Value : 2
Counter Value : 3
shubham@ShubhsPC:~$
```