Name : Hassan Ali
Roll No : 20P-0149
Section : A
Assignment : 05

Code:

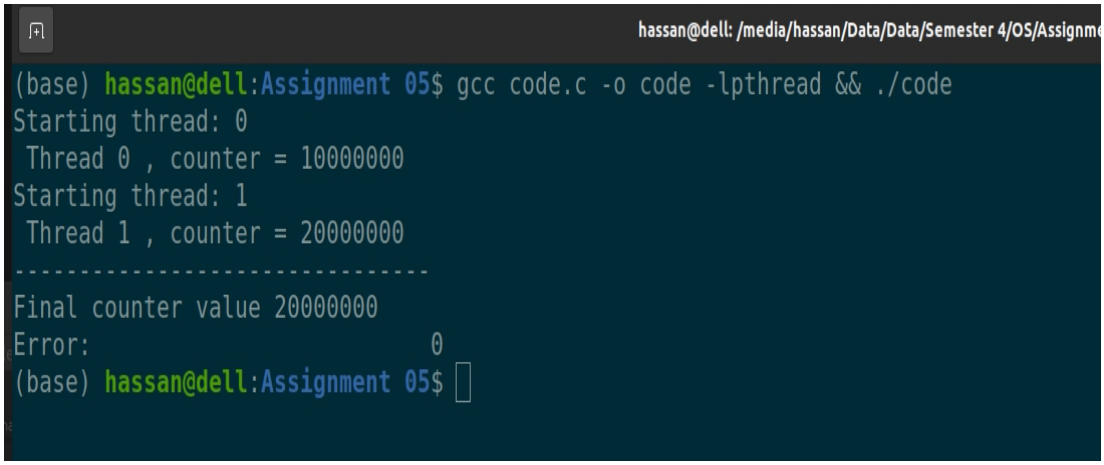
media > hassan > Data > Data > Semester 4 > OS > Assignments > Assignment 05 > C code.c > main()

```c
#include<unistd.h>
#include<sys/types.h>
#include<errno.h>
#include<stdio.h>
#include<stdlib.h>
#include<pthread.h>
#include<string.h>
#include<semaphore.h>

#define NUMS_RUNS 10000000

void handler(void *ptr);

int counter;
sem_t mutex;

int main(){
    int i[2];
    sem_init(&mutex ,0, 1);
    pthread_t thread_a;
    pthread_t thread_b;
    i[0] = 0;
    i[1] = 1;

    pthread_create(&thread_a , NULL , (void *) &handler,(void *) &i[0]);
    pthread_create(&thread_b , NULL , (void *) &handler,(void *) &i[1]);

    pthread_join(thread_a , NULL);
    pthread_join(thread_b , NULL);

    printf("--------------------------------\n");
    printf("Final counter value %d\n",counter);
    printf("Error:                          %d\n",(NUMS_RUNS*2-counter));
    exit(0);
}
```

media > hassan > Data > Data > Semester 4 > OS > Assignments > Assignment 05 > C code.c > handler(void *)

```c
void handler(void *ptr){
    sem_wait(&mutex);
    int iter = 0 ;
    int thread_num;
    thread_num = *((int *) ptr);
    printf("Starting thread: %d \n ", thread_num);

    while(iter <NUMS_RUNS){
        counter++;
        iter += 1;
    }
    printf("Thread %d , counter = %d \n", thread_num , counter);
    sem_post(&mutex);
    sem_destroy(&mutex);
    pthread_exit(0);
}
```

Output:


hassan@dell: /media/hassan/Data/Data/Semester 4/OS/Assignm
(base) hassan@dell:Assignment 05$ gcc code.c -o code -lpthread && ./code
Starting thread: 0
Thread 0 , counter = 10000000
Starting thread: 1
Thread 1 , counter = 20000000
Final counter value 20000000
Error: 0
(base) hassan@dell:Assignment 05$