National Textile University

Department of Computer Science

Subject:

Operating System

Submitted to:

Sir Nasir

Submitted by:

Esha Mubashir Khan

Reg number:

23-NTU-CS-1151

Lab no: **4**

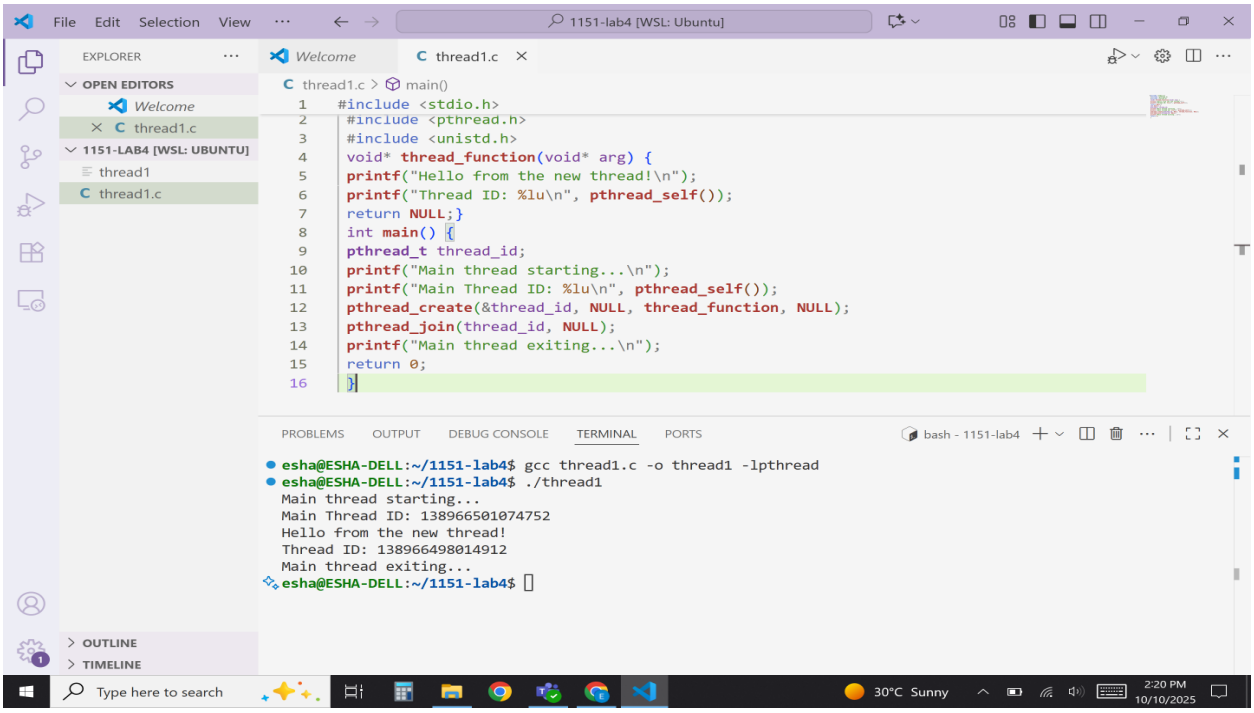
Semester: **5th**

---

3. C Programs with Threads

Program 1: Creating a Simple Thread

```
#include <stdio.h>
#include <pthread.h>
#include <unistd.h>
void* thread_function(void* arg) {
printf("Hello from the new thread!\n");
printf("Thread ID: %lu\n", pthread_self());
return NULL;}
int main() {
pthread_t thread_id;
printf("Main thread starting...\n");
printf("Main Thread ID: %lu\n", pthread_self());
pthread_create(&thread_id, NULL, thread_function, NULL);
pthread_join(thread_id, NULL);
printf("Main thread exiting...\n");
return 0;
}
```



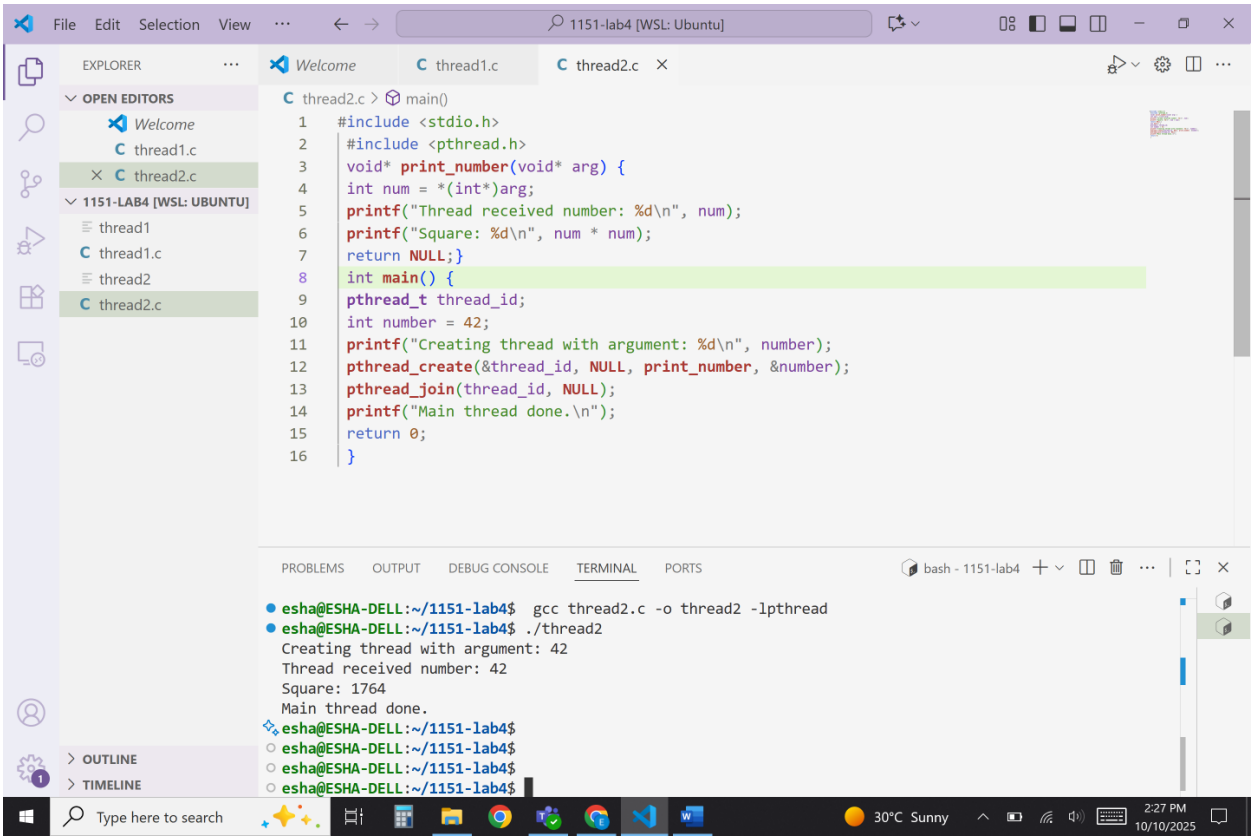
Program 2: Passing Arguments to Threads

```
#include <stdio.h>

#include <pthread.h>

void* print_number(void* arg) {

int num = *(int*)arg;

printf("Thread received number: %d\n", num);

printf("Square: %d\n", num * num);

return NULL;}

int main() {

pthread_t thread_id;

int number = 42;

printf("Creating thread with argument: %d\n", number);

pthread_create(&thread_id, NULL, print_number, &number);

pthread_join(thread_id, NULL);
```

printf("Main thread done.\n");

return 0;

}



Program 3: Passing Multiple Data

#include <stdio.h>

#include <pthread.h>

typedef struct {

int id;

char* message;

} **ThreadData;**

void* **printData**(void* arg) {

ThreadData* data = (**ThreadData***)arg;

printf("Thread %d says: %s\n", data->id, data->message);

return **NULL**; }

int **main**() {

pthread_t t1, t2;

ThreadData data1 = {1, "Hello"};

ThreadData data2 = {2, "World"};

pthread_create(&t1, **NULL**, **printData**, &data1);

pthread_create(&t2, **NULL**, **printData**, &data2);

pthread_join(t1, **NULL**);

pthread_join(t2, **NULL**);

printf("All threads done.\n");

return 0; }