



NATIONAL TEXTILE

UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE

SUBMITTED BY:

Farah Naz

23-NTU-CS-1152

SECTION SE: 5th(A)

LAB MANUAL

SUBMITTED TO:

Sir Nasir Mehmood

SUBMISSION DATE:

10-10-2025

Question 1:

Program 1: Creating a Simple Thread

Objective: Create a thread and print messages from both main thread and new thread.

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

// Thread function - this will run in the new thread
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());

    // Create a new thread
    pthread_create(&thread_id, NULL, thread_function, NULL);
```

```
    // Wait for the thread to finish
    pthread_join(thread_id, NULL);

    printf("Main thread exiting...\n");
    return 0;
}
```

Create Thread:

Code:

```
#include <stdio.h>
#include <pthread.h>
#include <unistd.h>
// Thread function - this will run in the new thread
```

```

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());
    // Create a new thread
    pthread_create(&thread_id, NULL, thread_function, NULL);
    // Wait for the thread to finish
    pthread_join(thread_id, NULL);
    printf("Main thread exiting...\n");
    return 0;
}

```

Output:

The screenshot shows the Visual Studio Code interface with a file named `create_thread.c` open. The code defines a `thread_function` that prints a message and its thread ID, and a `main` function that creates and joins this thread. The terminal output shows the following sequence of events:

```

farah@DESKTOP-LI8S698:~/lab4-1152$ gcc create_thread.c -o create.o
farah@DESKTOP-LI8S698:~/lab4-1152$ ./create.o
Main thread starting...
Main Thread ID: 139821950109504
Hello from the new thread!
Thread ID: 139821950043840
Main thread exiting...
farah@DESKTOP-LI8S698:~/lab4-1152$

```

Question 2:

Program 2: Passing Arguments to Threads

Objective: Pass data to a thread function.

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

void* print_number(void* arg) {
```

```
    // We know that we've passed an integer pointer
    int num = *(int*)arg; // Cast void* back to int*
    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);

    // Pass address of 'number' to thread
    pthread_create(&thread_id, NULL, print_number, &number);

    pthread_join(thread_id, NULL);

    printf("Main thread done.\n");
    return 0;
}
```

Pass Arguments to the thread:

Code:

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

void* print_number(void* arg) {
    // We know that we've passed an integer pointer
    int num = *(int*)arg; // Cast void* back to int*
    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);
    // Pass address of 'number' to thread
    pthread_create(&thread_id, NULL, print_number, &number);
    ;pthread join will block main process until the created one completes and
then resume the main thread
    pthread_join(thread_id, NULL);
    printf("Main thread done.\n");
    return 0;
}

```

Output:

The screenshot shows the Visual Studio Code interface with a C program named `arguments.c` open. The program creates a thread that prints the square of the number 42. The terminal output shows the execution results.

```

lab4-1152 [WSL: Ubuntu-24.04]
EXPLORER
  LAB4-1152 [WSL: UBUNTU-24.04]
    arguments_o.o
    C arguments.c
    C create_thread.c
    create.o
  OUTLINE
  TIMELINE
  PROBLEMS
  OUTPUT
  TERMINAL
    bash - lab4-1152
    farah@DESKTOP-LI8S698:~/lab4-1152$ ./arguments_o.o
    Creating thread with argument: 42
    Thread received number: 42
    Square: 1764
    Main thread done.
    farah@DESKTOP-LI8S698:~/lab4-1152$

```