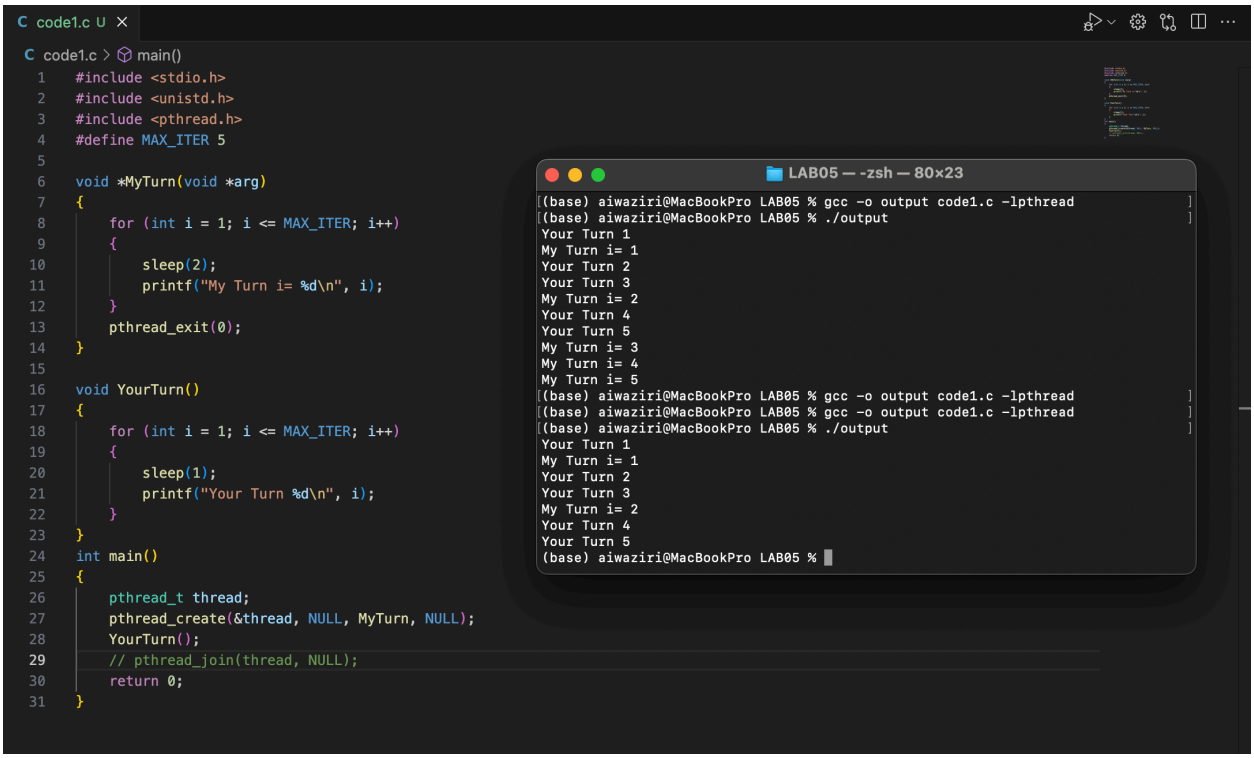
Name: Abubakar Waziri

ID: 4220056

Q1a:



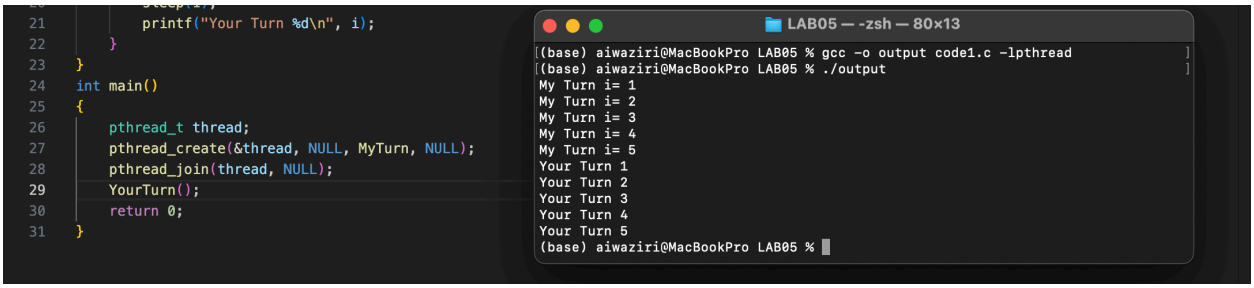
```c
#include <stdio.h>
#include <unistd.h>
#include <pthread.h>
#define MAX_ITER 5

void *MyTurn(void *arg)
{
    for (int i = 1; i <= MAX_ITER; i++)
    {
        sleep(2);
        printf("My Turn i= %d\n", i);
    }
    pthread_exit(0);
}

void YourTurn()
{
    for (int i = 1; i <= MAX_ITER; i++)
    {
        sleep(1);
        printf("Your Turn %d\n", i);
    }
}
int main()
{
    pthread_t thread;
    pthread_create(&thread, NULL, MyTurn, NULL);
    YourTurn();
    // pthread_join(thread, NULL);
    return 0;
}
```

```
LAB05 — -zsh — 80×23
(base) aiwaziri@MacBookPro LAB05 % gcc -o output code1.c -lpthread
(base) aiwaziri@MacBookPro LAB05 % ./output
Your Turn 1
My Turn i= 1
Your Turn 2
Your Turn 3
My Turn i= 2
Your Turn 4
Your Turn 5
My Turn i= 3
My Turn i= 4
My Turn i= 5
(base) aiwaziri@MacBookPro LAB05 % gcc -o output code1.c -lpthread
(base) aiwaziri@MacBookPro LAB05 % gcc -o output code1.c -lpthread
(base) aiwaziri@MacBookPro LAB05 % ./output
Your Turn 1
My Turn i= 1
Your Turn 2
Your Turn 3
My Turn i= 2
Your Turn 4
Your Turn 5
(base) aiwaziri@MacBookPro LAB05 %
```

Explanation: YES

Removing ***pthread_join()*** will lead to incomplete execution of the ***MyTurn*** thread, resulting in missing output from that thread.

Q1b:



```
        printf("Your Turn %d\n", i);
    }
}
int main()
{
    pthread_t thread;
    pthread_create(&thread, NULL, MyTurn, NULL);
    pthread_join(thread, NULL);
    YourTurn();
    return 0;
}
```

```
LAB05 — -zsh — 80×13
(base) aiwaziri@MacBookPro LAB05 % gcc -o output code1.c -lpthread
(base) aiwaziri@MacBookPro LAB05 % ./output
My Turn i= 1
My Turn i= 2
My Turn i= 3
My Turn i= 4
My Turn i= 5
Your Turn 1
Your Turn 2
Your Turn 3
Your Turn 4
Your Turn 5
(base) aiwaziri@MacBookPro LAB05 %
```

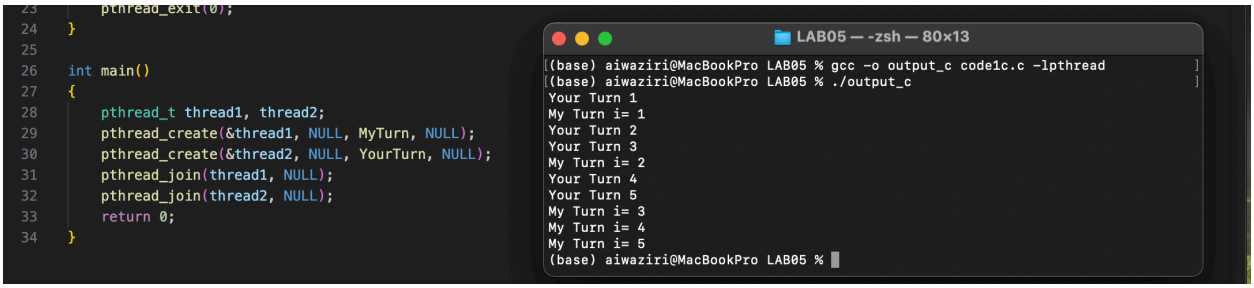
Explanation of Output:

a) The ***MyTurn*** thread starts and prints "*My Turn i= 1" to "My Turn i= 5*", with a 2-second sleep between each print.

b) The *pthread_join(thread, NULL);* ensures that the main thread waits for the ***MyTurn*** thread to complete before proceeding.

c) After the ***MyTurn*** thread completes, the ***YourTurn*** function is called, which prints "Your Turn 1" to "Your Turn 5", with a 1-second sleep between each print.

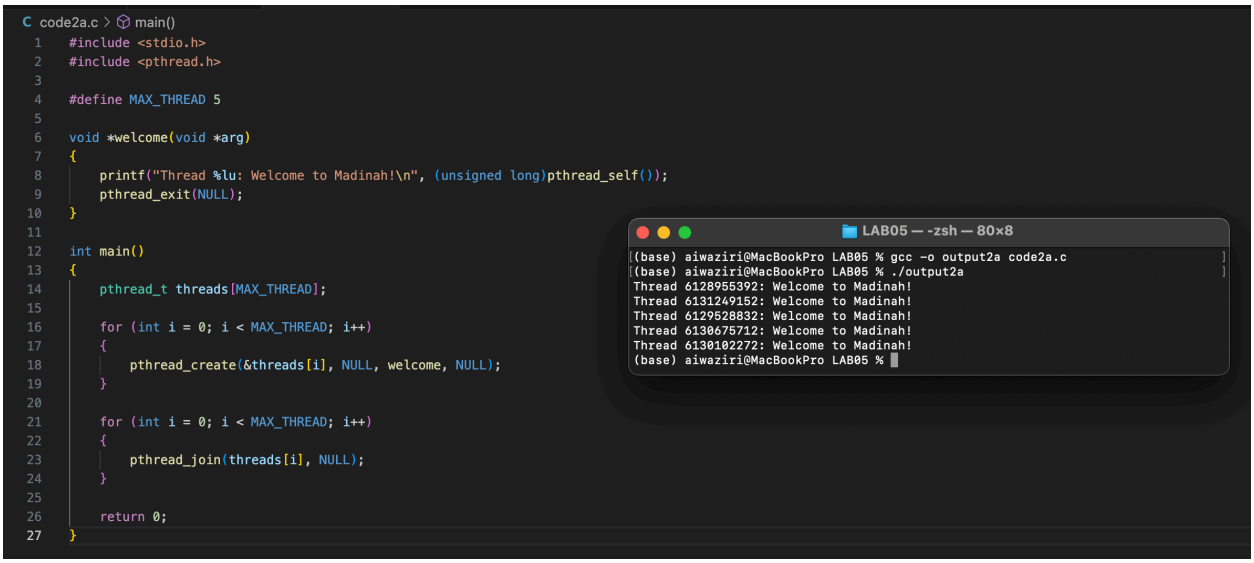
Q1c:



```
    pthread_exit(0);
}

int main()
{
    pthread_t thread1, thread2;
    pthread_create(&thread1, NULL, MyTurn, NULL);
    pthread_create(&thread2, NULL, YourTurn, NULL);
    pthread_join(thread1, NULL);
    pthread_join(thread2, NULL);
    return 0;
}
```

```
LAB05 — -zsh — 80×13
(base) aiwaziri@MacBookPro LAB05 % gcc -o output_c code1c.c -lpthread
(base) aiwaziri@MacBookPro LAB05 % ./output_c
Your Turn 1
My Turn i= 1
Your Turn 2
Your Turn 3
My Turn i= 2
Your Turn 4
Your Turn 5
My Turn i= 3
My Turn i= 4
My Turn i= 5
(base) aiwaziri@MacBookPro LAB05 %
```

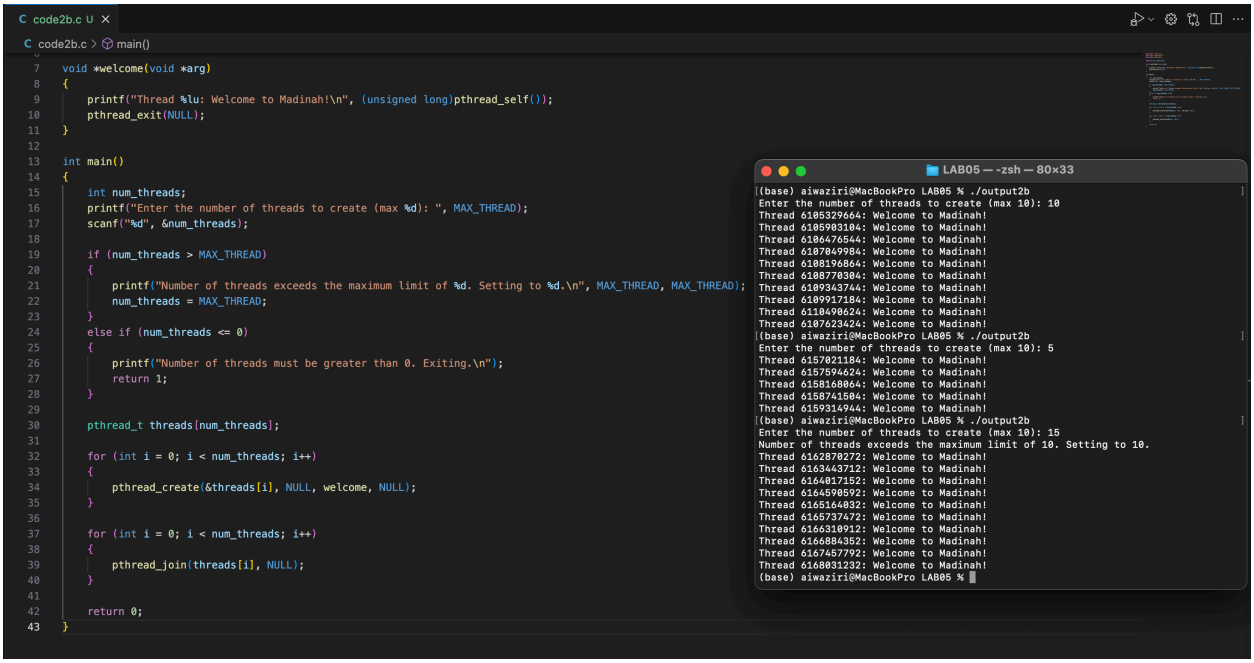
Q2a:



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

#define MAX_THREAD 5

void *welcome(void *arg)
{
    printf("Thread %lu: Welcome to Madinah!\n", (unsigned long)pthread_self());
    pthread_exit(NULL);
}

int main()
{
    pthread_t threads[MAX_THREAD];

    for (int i = 0; i < MAX_THREAD; i++)
    {
        pthread_create(&threads[i], NULL, welcome, NULL);
    }

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

    return 0;
}
```

```
LAB05 — -zsh — 80×8
(base) aiwaziri@MacBookPro LAB05 % gcc -o output2a code2a.c
(base) aiwaziri@MacBookPro LAB05 % ./output2a
Thread 6128955392: Welcome to Madinah!
Thread 6131249152: Welcome to Madinah!
Thread 6129528832: Welcome to Madinah!
Thread 6130675712: Welcome to Madinah!
Thread 6130102272: Welcome to Madinah!
(base) aiwaziri@MacBookPro LAB05 %
```

Q2b:



```c
void *welcome(void *arg)
{
    printf("Thread %lu: Welcome to Madinah!\n", (unsigned long)pthread_self());
    pthread_exit(NULL);
}

int main()
{
    int num_threads;
    printf("Enter the number of threads to create (max %d): ", MAX_THREAD);
    scanf("%d", &num_threads);

    if (num_threads > MAX_THREAD)
    {
        printf("Number of threads exceeds the maximum limit of %d. Setting to %d.\n", MAX_THREAD, MAX_THREAD);
        num_threads = MAX_THREAD;
    }
    else if (num_threads <= 0)
    {
        printf("Number of threads must be greater than 0. Exiting.\n");
        return 1;
    }

    pthread_t threads[num_threads];

    for (int i = 0; i < num_threads; i++)
    {
        pthread_create(&threads[i], NULL, welcome, NULL);
    }

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

    return 0;
}
```

```
LAB05 — -zsh — 80×33
(base) aiwaziri@MacBookPro LAB05 % ./output2b
Enter the number of threads to create (max 10): 10
Thread 6105329664: Welcome to Madinah!
Thread 6105903104: Welcome to Madinah!
Thread 6106476544: Welcome to Madinah!
Thread 6107049984: Welcome to Madinah!
Thread 6108196864: Welcome to Madinah!
Thread 6108770304: Welcome to Madinah!
Thread 6109343744: Welcome to Madinah!
Thread 6109917184: Welcome to Madinah!
Thread 6110490624: Welcome to Madinah!
Thread 6107623424: Welcome to Madinah!
(base) aiwaziri@MacBookPro LAB05 % ./output2b
Enter the number of threads to create (max 10): 5
Thread 6157021184: Welcome to Madinah!
Thread 6157594624: Welcome to Madinah!
Thread 6158168064: Welcome to Madinah!
Thread 6158741504: Welcome to Madinah!
Thread 6159314944: Welcome to Madinah!
(base) aiwaziri@MacBookPro LAB05 % ./output2b
Enter the number of threads to create (max 10): 15
Number of threads exceeds the maximum limit of 10. Setting to 10.
Thread 6162870272: Welcome to Madinah!
Thread 6163443712: Welcome to Madinah!
Thread 6164017152: Welcome to Madinah!
Thread 6164590592: Welcome to Madinah!
Thread 6165164032: Welcome to Madinah!
Thread 6165737472: Welcome to Madinah!
Thread 6166310912: Welcome to Madinah!
Thread 6166884352: Welcome to Madinah!
Thread 6167457792: Welcome to Madinah!
Thread 6168031232: Welcome to Madinah!
(base) aiwaziri@MacBookPro LAB05 %
```