Assignment #4
Saad Ahmad
(20P-0051)

Task 1

Part A) We had created two functions named as thread1 and thread2 and in main function we have created the threads using the pthread_create function and then pass it the arguments and it calls the function thread1 and thread2 respectively and print the string.
The pthread_join suspend the execution of the calling thread until the target *thread* terminates. After that it will continue its execution.

Code:

```c
1    #include<stdio.h>
2    #include<pthread.h>
3    #include<stdlib.h>
4
5    void* thread1()
6    {
7        for (int c = 0; c < 10; c++)
8        {
9            printf("Hello\n");
10       }
11   }
12
13   void* thread2()
14   {
15       for (int c = 0; c < 10; c++)
16       {
17           printf("World\n");
18       }
19   }
20
21   int main()
22   {
23       int status;
24       pthread_t tid1 , tid2;
25
26       pthread_create(&tid1 , NULL , thread1 , NULL);
27       pthread_create(&tid2 , NULL , thread2 , NULL);
28
29       pthread_join(tid1 , NULL);
30       pthread_join(tid2 , NULL);
31
32       return 0;
33   }
34
35
```

Output:

```
ayyzenn@Saad-A:~/Desktop/OS$ gcc task1.c -lpthread
ayyzenn@Saad-A:~/Desktop/OS$ ./a.out
Hello
Hello
Hello
Hello
Hello
Hello
Hello
Hello
Hello
Hello
World
World
World
World
World
World
World
World
World
World
ayyzenn@Saad-A:~/Desktop/OS$
```

Part B)
Code:

```c
C task1.c  ×   C task2.c

os > C task1.c > ...
   1    #include<stdio.h>
   2    #include<pthread.h>
   3    #include<stdlib.h>
   4
   5    void* thread1()
   6    {
   7        for (int c = 0; c < 10; c++)
   8        {
   9            printf("Hello\n");
  10        }
  11    }
  12
  13    void* thread2()
  14    {
  15        for (int c = 0; c < 10; c++)
  16        {
  17            printf("World\n");
  18        }
  19    }
  20
  21    int main()
  22    {
  23        int status;
  24        pthread_t tid1 , tid2 , tid3 , tid4;
  25
  26        pthread_create(&tid1 , NULL , thread1 , NULL);
  27        pthread_create(&tid2 , NULL , thread2 , NULL);
  28
  29        pthread_create(&tid3 , NULL , thread1 , NULL);
  30        pthread_create(&tid4 , NULL , thread2 , NULL);
  31
  32        pthread_join(tid1 , NULL);
  33        pthread_join(tid2 , NULL);
  34
  35        pthread_join(tid3 , NULL);
  36        pthread_join(tid4 , NULL);
  37
  38        return 0;
  39    }
```

Output:

```
ayyzenn@Saad-A:~/Desktop/OS$ gcc task1.c -lpthread
ayyzenn@Saad-A:~/Desktop/OS$ ./a.out
Hello
Hello
Hello
Hello
Hello
Hello
Hello
World
World
World
World
World
World
World
Hello
Hello
Hello
Hello
Hello
Hello
Hello
Hello
Hello
Hello
World
World
World
World
World
World
World
World
World
World
Hello
Hello
Hello
World
World
World
ayyzenn@Saad-A:~/Desktop/OS$ 
```

Task 2

Part A)
Code:

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

#define NUM_RUNS 1000000

void handler(void *ptr);

int counter;

int main()
{
    int i[2];
    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" , (NUM_RUNS*2 - counter));

    exit(0);

}

void handler(void *ptr)
{
    int iter = 0;
    int thread_num;
    thread_num = *((int *) ptr);

    printf("String thread: %d \n ",  thread_num);

    while (iter < NUM_RUNS)
    {
        counter++;
        iter += 1;
    }

    printf("Thread %d, counter = %d \n" , thread_num , counter);
    pthread_exit(0);

}
```

Output:

```
ayyzenn@Saad-A:~/Desktop/OS$ gcc task2.c -lpthread
ayyzenn@Saad-A:~/Desktop/OS$ ./a.out
String thread: 0
 String thread: 1
 Thread 0, counter = 1244159
Thread 1, counter = 1469110
------------------------------------------------------------
Final counter value:      1469110
Error:                     530890
ayyzenn@Saad-A:~/Desktop/OS$
```

Part B)

  I.   The value of counter variable should be **2000000**
 II.   The value I get is **1469110**
III.   Error is **530890** and it is varied by different values like 540443, 956441, 848464, 531317.
 IV.   User time is **0m0.024s**