Task#1:

Create 4 threads in the main function, with each thread printing a different sequence. Thread#1 will print integers (1-5), Thread#2 will print integers (6-10) etc… Perform this without using pthread_join

Task#2

Create 4 threads in the main function, with each thread printing a different sequence. Thread#1 will print integers (1-5), Thread#2 will print integers (6-10) etc… Perform this using pthread_join so that the order of execution is controlled in sequence of numbers.

Task#3

Take a number as input from the user. Create 2 threads. One thread should multiply this number by 2 and update the global value. The other thread should increment the number by 1 and update the global value.

Task#4

Perform the same as Task#3 but now use mutex locks and unlocks to prevent the other thread from accessing the global variable while one thread is accessing and updating it.