Assignment 4

Write a program to generate and print Fibonacci series, one thread must generate the series upto number and other thread must print them.
Ensure proper synchronization.

```
import threading
import time
```

```
def generate_fibonacci(n, fib_list, lock, start_time):
    a, b = 0, 1
    for _ in range(n):
        with lock:
            fib_list.append((a, time.time() - start_time))
        a, b = b, a + b
```

```
def print_fibonacci(fib_list, lock):
    with lock:
        for entry in fib_list:
            thread_id = threading.current_thread().ident
            current_time = time.strftime("%Y-%m-%d %H:%M:%S", time.localtime())
            fibonacci_value, time_taken = entry
            print(f"{fibonacci_value}, Thread ID {thread_id} at {current_time}: , Time Taken: {time_taken:.6f} seconds")
```

```
def main():
    n = int(input("Enter the number of Fibonacci numbers to generate: "))

    fib_list = []
    lock = threading.Lock()
    start_time = time.time()

    # Create two threads
    generate_thread = threading.Thread(target=generate_fibonacci, args=(n, fib_list, lock, start_time))
    print_thread = threading.Thread(target=print_fibonacci, args=(fib_list, lock))

    # Start the threads
    generate_thread.start()
    print_thread.start()

    # Wait for both threads to finish
    generate_thread.join()
    print_thread.join()

if __name__ == "__main__":
    main()
```

```
 Enter the number of Fibonacci numbers to generate: 17
 0, Thread ID 138807072519744 at 2024-02-01 11:55:04: , Time Taken: 0.000328 seconds
 1, Thread ID 138807072519744 at 2024-02-01 11:55:04: , Time Taken: 0.000330 seconds
 1, Thread ID 138807072519744 at 2024-02-01 11:55:04: , Time Taken: 0.000331 seconds
 2, Thread ID 138807072519744 at 2024-02-01 11:55:04: , Time Taken: 0.000332 seconds
 3, Thread ID 138807072519744 at 2024-02-01 11:55:04: , Time Taken: 0.000333 seconds
 5, Thread ID 138807072519744 at 2024-02-01 11:55:04: , Time Taken: 0.000335 seconds
 8, Thread ID 138807072519744 at 2024-02-01 11:55:04: , Time Taken: 0.000335 seconds
 13, Thread ID 138807072519744 at 2024-02-01 11:55:04: , Time Taken: 0.000337 seconds
 21, Thread ID 138807072519744 at 2024-02-01 11:55:04: , Time Taken: 0.000338 seconds
 34, Thread ID 138807072519744 at 2024-02-01 11:55:04: , Time Taken: 0.000340 seconds
 55, Thread ID 138807072519744 at 2024-02-01 11:55:04: , Time Taken: 0.000341 seconds
 89, Thread ID 138807072519744 at 2024-02-01 11:55:04: , Time Taken: 0.000343 seconds
 144, Thread ID 138807072519744 at 2024-02-01 11:55:04: , Time Taken: 0.000344 seconds
 233, Thread ID 138807072519744 at 2024-02-01 11:55:04: , Time Taken: 0.000346 seconds
 377, Thread ID 138807072519744 at 2024-02-01 11:55:04: , Time Taken: 0.000347 seconds
 610, Thread ID 138807072519744 at 2024-02-01 11:55:04: , Time Taken: 0.000348 seconds
 987, Thread ID 138807072519744 at 2024-02-01 11:55:04: , Time Taken: 0.000350 seconds
```