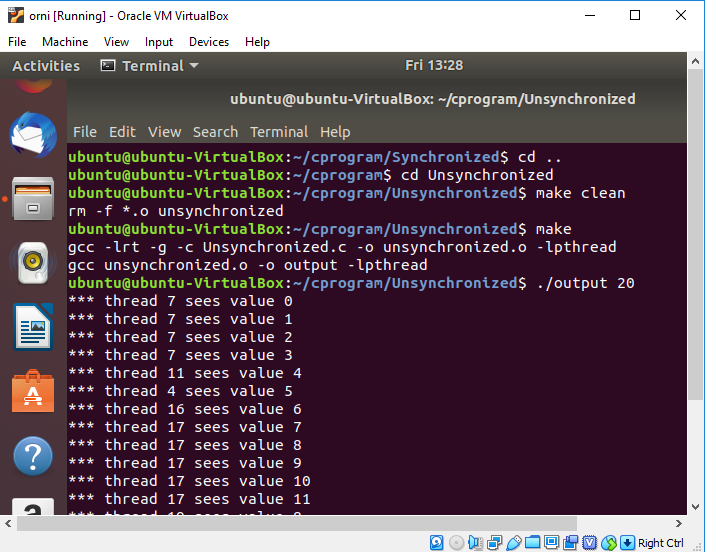
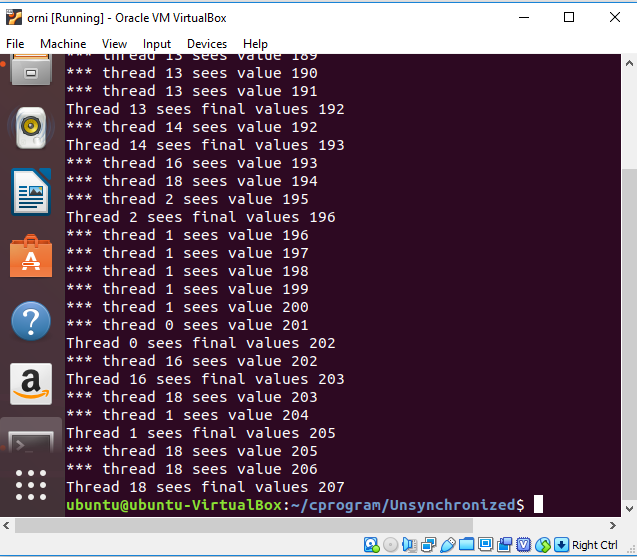
**Project 1: Multithreaded Programming and Synchronization**

**Tasmia Aqila**

**Panther Id: 6144411**

A thread is a path of execution within a process. Thread within the same process run in a shared memory space, while processes run in separate memory spaces. In this project, we are implementing multithread programming to analyze the power of synchronization, which is one of the most important tasks in OS.

**Results of Pthreads without synchronization:** Here we can see from the screenshot that, when no synchronization mechanism is used in the program, the task i.e increasing the shared variable value is not done properly. Because when a thread gets access to the shared variable, at the same time the OS can make a switch to some other threads which is scheduled at that time. So, the first scheduled thread may not be able to complete it’s execution of the loop.



Some threads may end earlier without waiting for others to complete their task. As a result, all threads can’t see the same value at the end of the execution.

**Results of Pthreads with synchronization:**  Here we can see from the screenshot that, when synchronization mechanism is used in the program, the task i.e increasing the shared variable value is done properly. I have used **pthread\_barrier** here so when one thread is sent to **sleep()**,after it wakes up from sleep it can restart it’s execution exactly from that point where it was gone to sleep. I also used the **sleep()** after unlocking the lock variable otherwise the locking thread will block other threads to get access to that shared variable. **pthread\_join()** function is used so that all threads wait for each other’s execution to terminate. As a result, all the threads see the same value of the shared variable at the end of the execution. At the end**, pthread\_exit()** is used in **main()** function to allow other threads to continue execution as the main thread should terminate by calling **pthread\_exit()** . Results are shown below:

