Write up for Question 1

Steps followed:

1. I have written a C language program where three threads CountA_thread, CountB_thread, CountC_thread are being created.
2. All these three threads invoke their respective count functions countA(), countB(), countC(), that I have created.
3. Each of these functions does the job of counting from 1 to 2^{32}.
4. I have also used clock_gettime() function for obtaining the actual time ticks that is being used to compute the time it took to complete a function. (clock() instead of gettime() as I found it better working)
5. Thread for CountA uses SCHED_OTHER scheduling discipline with standard priority.
6. Thread for CountB uses SCHED_RR scheduling discipline with default priority
7. Thread for CountC uses SCHED_FIFO scheduling discipline with default priority.
8. Then I have tested my code for 6 different priorities.
9. On running the code, timings for every scheduling policy is printed.
10. Added histogram for the comparison between different scheduling policies are displayed.