Write up Mannual

Question 1:

Part 1.1

Linux pthreads and their scheduling

In this question what I have done is that I make a loop that run upto 10 times and in each loop what I does is that I will create 3 pthreads and each three calls their respective Thr_A(),Thr_B(),Thr_C() functions in pthread_create();

In Thr_A() what I have done is that this will make an integer variable set_priority=0 as SCHED_OTHER have 0 nice value only and then using structure sched_param and pthread_setschedparam() we have set the scheduling policy that the thread will follow with sched_priority.

In Thr_B() what I have done is that this will make an integer variable set_priority=1 as SCHED_RR have 1 to 10 nice values which will increase in every loop and then using structure sched_param and pthread_setschedparam() we have set the scheduling policy that the thread will follow with sched_priority

In Thr_C() what I have done is that this will make an integer variable set_priority=1 as SCHED_FIFO have 1 to 10 nice values which will increase in every loop and then using structure sched_param and pthread_setschedparam() we have set the scheduling policy that the thread will follow with sched_priority

And then using gettimeofday function belonging to time.h file that I have included in each cases I have calculate the time taken to count 0 to 2^32-1 in each case and stored the values in microseconds in the ans.txt file.

And then in Graph_q1.ipynb using this generated ans.txt file values I have created a bar graph showing the time taken for each pthreads with their respective scheduling policies and different nice values.

Question 1:
Part 1.2
Process scheduling:
In this question what I have done is that I have create 3 nested child processes using fork command in order to run all these child processes in congruency.

First here I have open a file ans2.txt in which in future I am going to put the values of time taken in each of three processes of compiling the kernel and then I have used 3 nested forks and each fork what I have done is just like the 1st question I have set diff scheduling_policies for different processes to execute the same kernel compilation and then just like in the 1st question I have calculated the time taken using the time.h file's functions.

And then in Graph_q1.ipynb using this generated ans2.txt file values I have created a bar graph showing the time taken for each processes with different scheduling_policies with their respective scheduling policies and fixed nice values.