

Question1:

Question1.1:

### Linux pthreads and their scheduling

In this question we have to create three thread which count  $2^0$  to  $2^{32}$ ;

This exercise is to show you how to use Linux's scheduling policies for three threads. You need to launch three threads, each of which relies on three different functions, countA(), countB() and countC(). Each function does the same thing, i.e. counts from 1 –  $2^{32}$

1. Thread 1 (call it ta()): Uses SCHED\_OTHER scheduling discipline with standard priority=0
2. Thread 2 (call it tb()): Uses SCHED\_RR scheduling discipline with default priority.
3. Thread 3 (call it tc()): Uses SCHED\_FIFO scheduling discipline with default priority.

Basically i used pthread\_t and create three pid and call it to it's respective function:

```
pthread_t a;  
pthread_create(&a, NULL, ta, NULL);
```

Here , ta is why above define function in which , i call it here and we countA() a function to count 1 to 4294967296 using pthread\_setschedparam()

```
struct sched_param parameters;  
parameters.sched_priority = prioty1;  
int works = pthread_setschedparam(pthread_self(), SCHED_OTHER,  
&parameters);  
// printf("/n%d\n",works);
```

And stored time by using clock\_gettime() and stored the time in a file "threadtime.txt"

And this is similarly for thread b and thread c.

Afterthat using **plot.py** to plot the graph by using "threadtime.txt" data .

In question 1.2

### Process scheduling:

This part of the exercise, involves creating three process, instead of the three threads.

Each of these process should involve compiling a copy of the Linux kernel source (with the minimal config shared by the TAs earlier).

The three processes are in parallel by calling wait at the end of all fork() and i used **execvp()** function to call the part of script.sh and used **sched\_set\_scheduler()** to set priority of the fifo , rr ==50,

Script.sh is a file where bash command is written to compile the kernel three times to respected directory(by using chdir()).

And clock\_gettime() function is used to get the time of compiler of kernel and stored the Time in file data.txt and at the end plot2.py used to plot the graph.

```
parameters.sched_priority = 50;  
int works = sched_setscheduler(getpid(), SCHED_FIFO, &parameters);  
char * test[]={"/home/vickey/Desktop/linux1/script.sh",NULL};  
execvp(test[0],test);
```

