Write-up

Code Description

I have added soft_runtime as an attribute in sched_entity and initialized it to zero in the file 'core.c'. I have made changes in the existing CFS class. In fair.c I have modified two functions: entity_before and update_curr.

- Time accounting in CFS is done by virtual runtime. Updates to the virtual runtime are performed in update_curr(). I have reduced soft_runtime by the delta_exec value in case soft_runtime is greater than delta_exec. Else, soft_runtime is set to zero.
- When a task is woken up or migrated, it is added to runqueue in the enqueue_entity function. entity_before is called in this function. I have compared two sched_entity instances in entity_before. In case, soft_runtime of sched_entity a is greater than sched_entity b, it is given higher priority.

I added a syscall rtnice in the syscall table (syscall number: 441). In sys.c, I have implemented the syscall. It takes pid and time as inputs find the task with the given pid and sets its soft_runtime to time.

How to run?

Commands:

```
$make compile
```

$make run

Expected Output

Time taken by child process followed by the string “(child)”.
Time taken by parent process followed by the string “(parent)”.

- Use command $dmesg to see messages print by syscall. Syscall prints messages to show the status (called, task not found and soft_runtime changed).

Errors

- ESRCH: Returned if no process or process group can be found corresponding to the given PID.
- Positive time