Operating Systems
Assignment 3

SUBMITTED BY : **ANANYA KANSAL**
ROLL NO : **2019458**

LOGIC AND IMPLEMENTATION

- In this assignment we are required to edit the completely fair scheduler in our operating system. The CFS scheduler uses the red-black tree to store processes according to their vruntime and extracts processes accordingly.
- We need to focus on the soft real-time requirement of our processes and prioritize them for which we check the rtnice value of that process.
- The higher the value of rtnice the higher its priority.
- We implemented a system call rtnice where we pass 2 arguments: the pid of the process and the rtnice value.
- We find the structure corresponding to our given input of pid, set it's rtnice value and keep updating its rtnice value so that each process has a fair chance of being executed.

OUTPUT

- We fork a process and pass the pid and rtnice value to the syscall. Our output depicts that even if both the child and parent process are given the same load, then the process having soft real time requirements is being executed first.

ERROR HANDLING

- In the test.c program **errno** is used to handle unexpected errors and invalid inputs from the user.
- For example if the process doesn’t exist or the pid isn’t in the valid range then it will return a negative value and exit.

- In the syscall code, I have used the function **EINVAL** for the same purpose of invalid inputs or if the task doesn't exist.