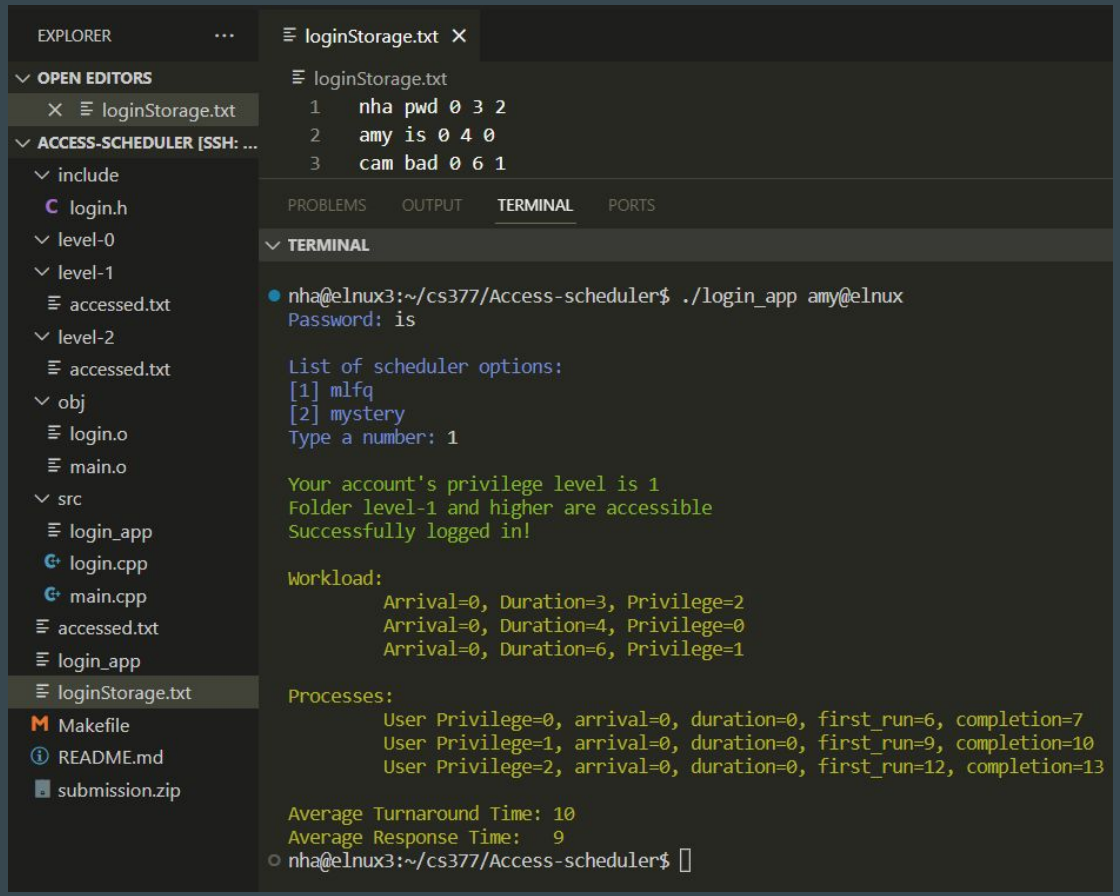
Access Scheduler

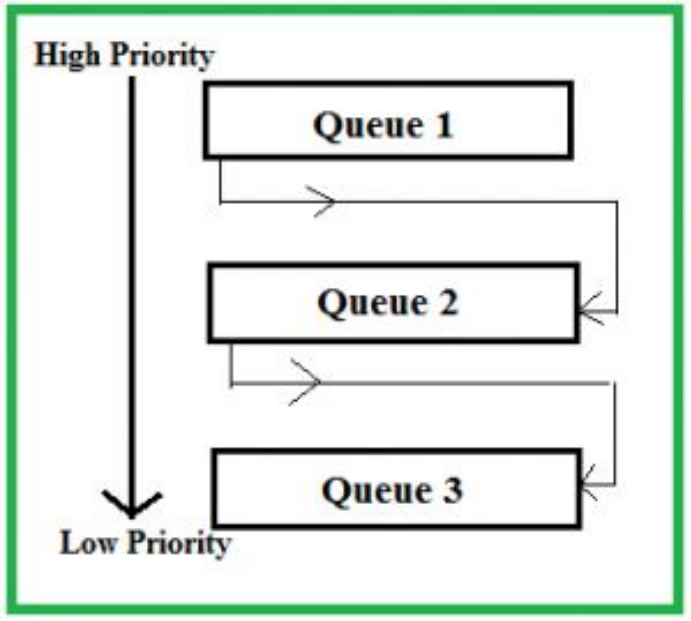
Nhi Ha
CS 377
Final Project

What is this project about?

- Extension of schedulers with a MLFQ and a mystery scheduler
- Integration of schedulers with user login authentication and their privilege levels
- Traverses directories based on user's privilege of each most efficient processes



MLFQ Explanation



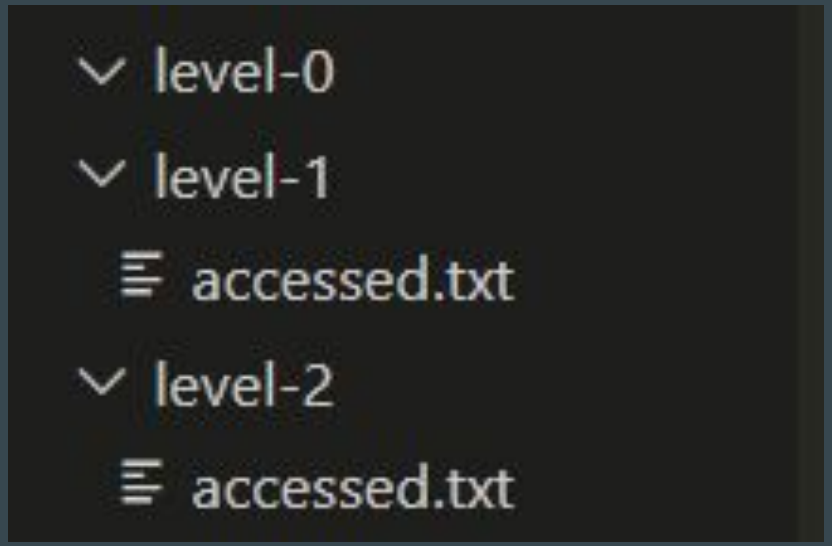
- **Rule 1:** If Priority(A) > Priority(B), A runs (B doesn't).
- **Rule 2:** If Priority(A) = Priority(B), A & B run in round-robin fashion using the time slice (quantum length) of the given queue.
- **Rule 3:** When a job enters the system, it is placed at the highest priority (the topmost queue).
- **Rule 4:** Once a job uses up its time allotment at a given level (regardless of how many times it has given up the CPU), its priority is reduced (i.e., it moves down one queue).
- **Rule 5:** After some time period S, move all the jobs in the system to the topmost queue.

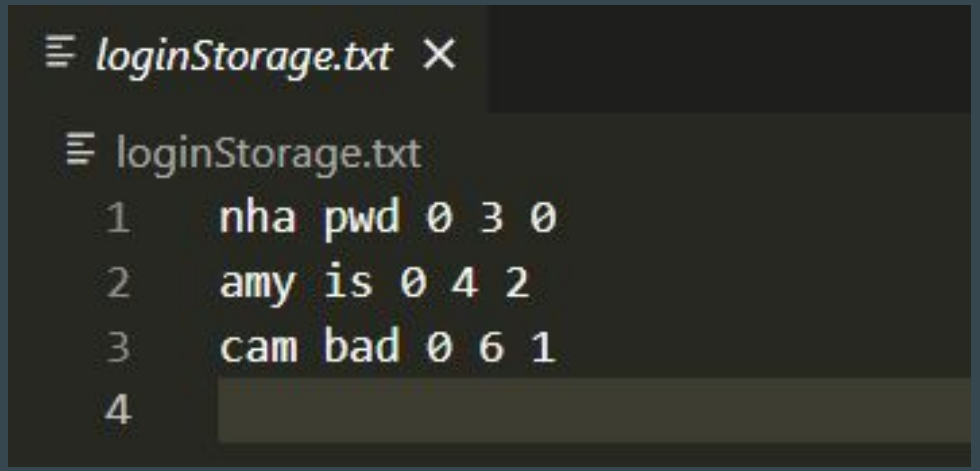
Sources:

https://www.geeksforgeeks.org/multilevel-feedback-queue-scheduling-mlfq-cpu-scheduling/

https://pages.cs.wisc.edu/~remzi/OSTEP/cpu-sched-mlfq.pdf

Privilege Levels Accessibility and Workload





- The lower the number -> The higher the privilege level of the folder.

- The txt means that the folder has already been accessed by the user based on privilege level

Column 1 -> Username

Column 2 -> Password

Column 3 -> Start time

Column 4 -> Duration

Column 6 -> Privilege levels (holder)

Why integrate privilege level and schedulers

- Can have process priorities and privilege levels go hand-to-hand

- assign priority levels to processes with any user privilege number, then adjust priorities based on the processes behavior using the scheduling algorithm

- A better user interaction to schedulers using login authentication and user's privilege levels