

# OPERATING SYSTEM PROJECT

simulation of a simple multiprogramming batch operating system

## SECTION 46530 | GROUP 3

Student Name	ID
<b>Leader  </b> Juman alyahya	438201355
Shahad alangari	438200180
Rahaf alhubeis	438200748
Renad almtwasi	438201172
Ghaida Alatiah	438201468

## HOW TO EXECUTE THE PROGRAM :

First, we start running the code form Operating System class, then the job.txt will be generated with 100 job and each job has it ID and Expected CPU usage (ECU), where the jobs will be in the hard disk queue.

Second, the jobs will load form the hard disk into the RAM “ready queue” and the state will change from NEW to READY. Then, the process scheduling will start, and it will follow the First-Come, First-Served (FCFS) algorithm policy, where the first job come will be selected by the CPU scheduler and execute from the ready queue. The job state will be changed from READY to RUNNING.

Third, the process will be terminated when the CUT exceeds ECU, and it will be generating an interrupt either normally with probability 10% or abnormally with probability 5% so the state will be changed from RUNNING to TERMINATED.

Finally, when all jobs finish processing a Result.TXT file will be generated and includes the numbers of jobs and the numbers of jobs that have completed their execution normally or abnormally.

## TASK DISTRIBUTION

TASK	Juman	Shahad	Rahaf	Renad	Ghaida
Generate File And Read From File	√				
Create CPU class (Executing the processes)		√			√
Create job and PCB class			√		
Create hardware class				√	
Write The Main	√	√	√	√	√
OUTPUT REPORT	√	√	√	√	√

## STUDENT REFLECTION ON THE SIMULATION

In the beginning of the simulation, we test Short term scheduler by taking jobs from ready queue to CPU and this process invoked in millisecond, we only test 100 jobs so we notice its execute them efficiently and with high speed performance but 100 jobs It's nothing in real life so we are not sure about performance if we increase jobs. We suggest improving performance in realistic operating System by having more than one CPU so the process will be faster than single CPU, also working with multi-threads and replacing the algorithms with round robin so processing will be faster and get the best performance for the CPU.