MILITARY COLLEGE OF SIGNALS, NUST

CPS-410 OPERATING SYSTEM LAB

FINAL EXAM (BESE14 - A&B)

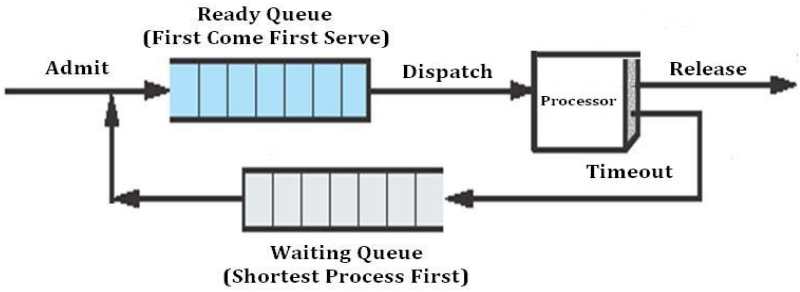
Instructor: Lab Engr. Nausheen **Total:** 20 Marks **Time Allowed:** 2.5 Hrs

---

Instructions:

1. Students are allowed to take help from MSDN and lab handouts.
2. Students are also allowed to have codes developed by them during previous labs.
3. Students must copy their code on LMS at the end of lab exam.
4. Use comments and proper naming scheme in program.
5. Copied code will be given zero mark.
6. No discussion among students is allowed during lab exam.

Consider the following process model:



The process model maintains two queues i.e. ready and waiting queue. Both queues can hold four different processes at a time. The ready queue follows first come first serve order. When a processor is ready, a process is dispatched from front of the ready queue. The processor executes that process for five seconds. If process execution is complete, the process is released from memory; otherwise it is placed in waiting queue. Whenever there is a room in ready queue, the process having shortest execution time will be removed from waiting queue and placed in ready queue. The processes waiting in ready queue will have priority over new processes waiting to get entry in ready queue.

Now, suppose we have five processes named as A, B, C, D and E. The arrival and execution time of each process is given in Table 1. Simulate the above mentioned process model for these five processes.

Table 1: Processes with their Arrival & Execution Time

Process	Arrival Time	Execution Time
A	0	5
B	2	7
C	5	10
D	7	8
E	10	6

Output:

When program will be executed first time, it should display all five processes with their arrival and execution time, the current status of ready queue and waiting queue. After that, whenever a process is removed from ready queue by the dispatcher, show the processes present inside ready and waiting queue with their remaining execution time. Also, display name of the process which was dispatched from ready queue to the processor for execution.