Name : Nikhil JSK

Reg. No : 11601789

Section : K1615

Roll No. : A26

E-mail : [nikhilj.strikes@gmail.com](mailto:nikhilj.strikes@gmail.com)

Github ID : nikhiljsk

**Question No. 1**

The problem number - 6 is to build an algorithm which can used on a scheduler to process the given processes. The processes arrival time and burst time are entered by the user. The algorithm should be able to handle processes so as assign the CPU to the process for a certain period of time not more than 10-time units in this case.

In problem number – 7, two queues are to be simulated. Where the first queue follows the ‘Priority scheduling algorithm’ and the second queue follows the ‘Round Robin algorithm’ to deal with the processes provided by the user. When a process of higher priority is found then the process which is allotted the CPU is pushed into the next queue. The second queue is executed only if the first one completes all the process.

**Question No. 2**

Algorithm for Problem – 6:

1. Take input from the user about the arrival and burst time.
2. Consider a structure which has attributes about the process and stores all those values.
3. Initialize the status of all the process to zero.
4. Compute the completion time for each process by using the total time variable.
5. If any process takes more than 10 time units to execute, time slice that process and place it back in the queue for further execution
6. Maintain a queue to store all the processes which have executed.
7. Calculate the turnaround time and waiting time for each process
8. Now, calculate the average of times accordingly.

Algorithm for Problem – 7:

1. Take input from the user about the arrival, burst time and also the priority of each process.
2. Consider a structure which has attributes about the process and stores all those values.
3. Initialize the status of all the process to zero.
4. Now start processing the processes in the first queue, if any processes pre-empts the process which is allotted the CPU, push it into the next queue.
5. The processes which are pushed into next queue should be allotted the CPU only if all the processes in the first queue are processed. The processes which are completed execution are pushed into the queue.
6. Start processing the queue using round robin method. The process is only executed for 2 seconds as the time quantum is assumed so. Once the process is executed push it into the queue.
7. Once all the processes are completed execution then display the queue which contains the order of execution.

**Question No. 3**

Problem No. 6

Complexity: O (n log(n))

The complexity specified above is the worst-case scenario.

Where n is the size of input.

Problem No. 7

Complexity: O (n2)

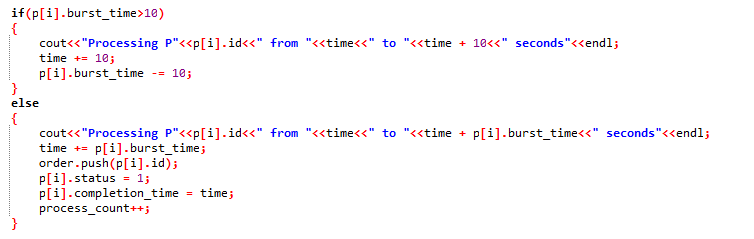
The complexity specified above is the worst-case scenario.

Where n is the size of input.

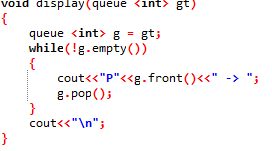
**Question No. 4**

*Problem No 6*

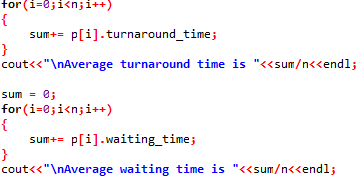
Time quantum of 10 units.



Maintains a queue (Showing the display)

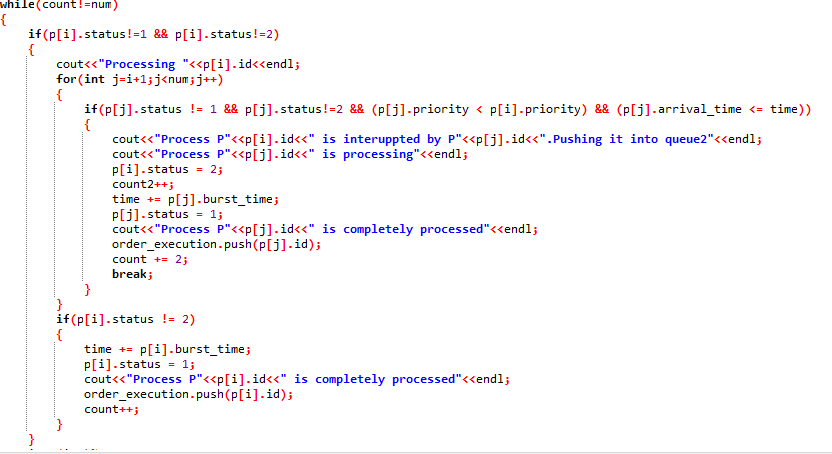


Average turnaround and Average waiting time

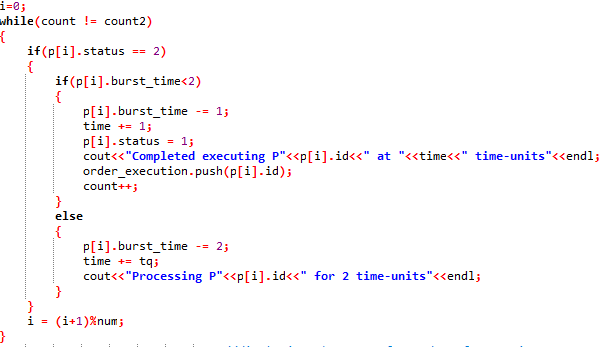


*Problem No 7*

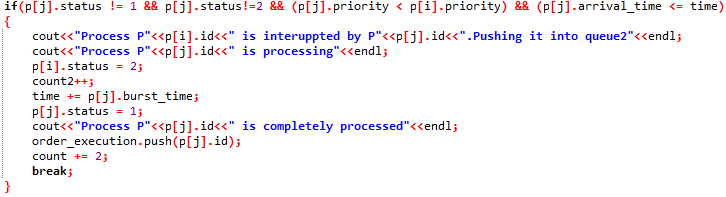
First Queue (Using priority)



Second Queue (Using Round Robin)



Interruptions



**Question No. 5**

The additional algorithm used in the both the problems is to sort the objects of the structures according to the arrival time and burst time.

It is used to implement the shortest job first or the First com first serve algorithm

**Question No. 6**

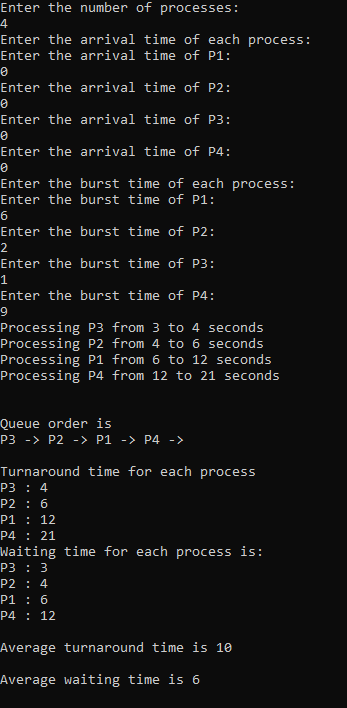
The boundary conditions of the algorithm used in both the question is the number of input cases. Which in this case is given 20 input-size.

**Question No. 7**

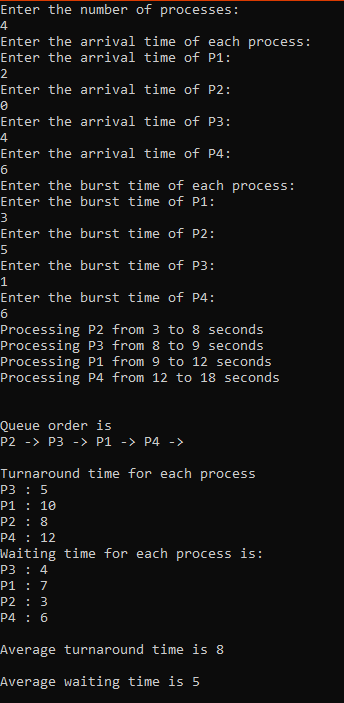
Test – Cases

*Problem No. 6*

Same Arrival Time

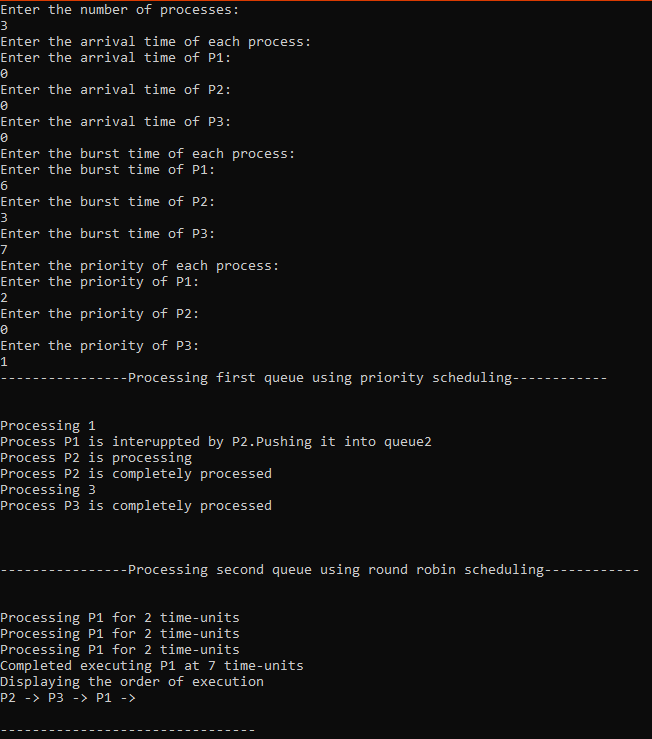


Different Arrival Time

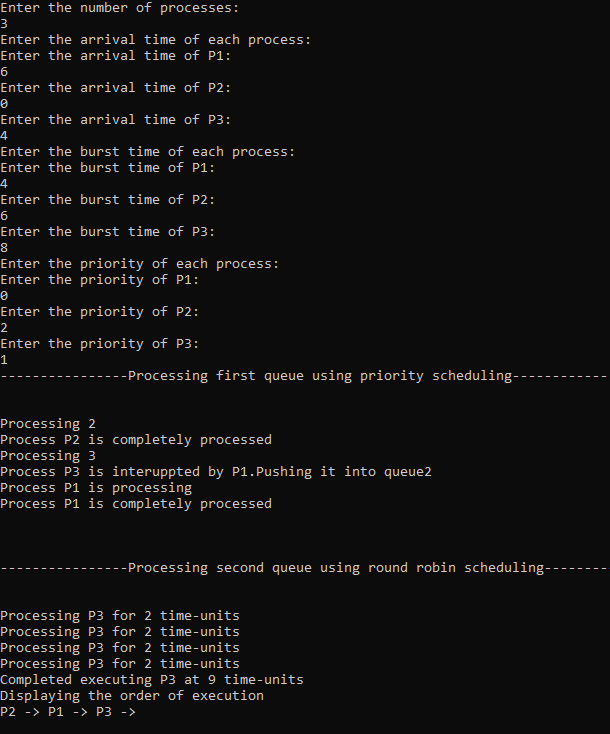


*Problem No. 7*

Same Arrival Time



Different Arrival Time



**Question No. 8**

No, I’ve uploaded the project but did not create 5 repositories of it previously.