**Process Scheduling Problem**

**(A Data Structure Project Solution in C++ using Queues)**

**Idea:-**

The process scheduling is the activity of the process manager that handles the removal of the running process from the CPU and the selection of another process on the basis of a particular strategy. Process scheduling is an essential part of a Multiprogramming operating systems. Such operating systems allow more than one process to be loaded into the executable memory at a time and the loaded process shares the CPU using time multiplexing.

First Come First Served (FCFS), Shortest Job First (SJF), ROUND-ROBIN algorithm are few algorithms that are used to schedule the execution of different processes. Implement a project that take various processes and their execution times are taken as input, and schedule them according to the selected strategy.

**Solution:-**

The solution is consist of a file named as “PSP Solution using Queues.cpp”. This solution is implemented by using the queues of STL in C++. There are some queues like ready queue and completion queue of CPU process scheduler which built by using the STL of C++. Processes are enters in the ready queue one by one during input and after scheduling according to the selected strategy, removed from ready queue and enters one by one into the completion queue. Run the file in any C++ compiler like Dev-C++ or online and schedule the queue after taking the input data from the user by using any strategy.

**Input Data:-**

* The input data consists of various processes and their execution times are taken as input.
* There are two categories to put input data in queue:

1. By putting sample data which is already defined for testing.
2. By taking the input from the user in console.

**Output Data:-**

* The output data consists of table which list down a set of processes according to their arrival time with many other times like Burst Time, Waiting Time, and Turnaround Time etc.
* The output also show both total & average of the waiting, turnaround times.
* Also output show the Gantt chart which show the order and the actual processing of various processes in the CPU with execution time according to the selected strategy of algorithm.

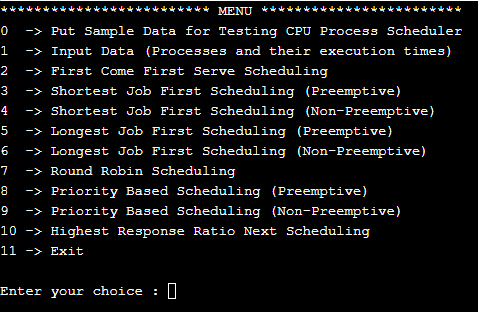
**Working:-**

There are two ways to put input in this solution for working of the code as per requirements, working of both methods are given below:

**a) For Sample Data**

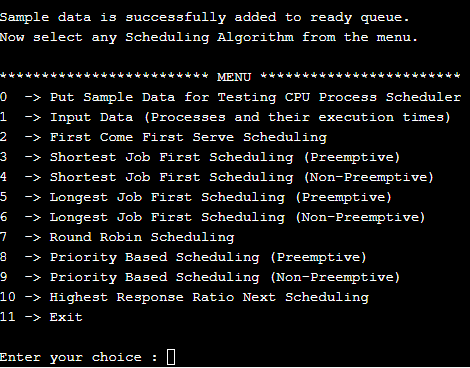
*->Step1*

The working screen show you the menu which looks like this:



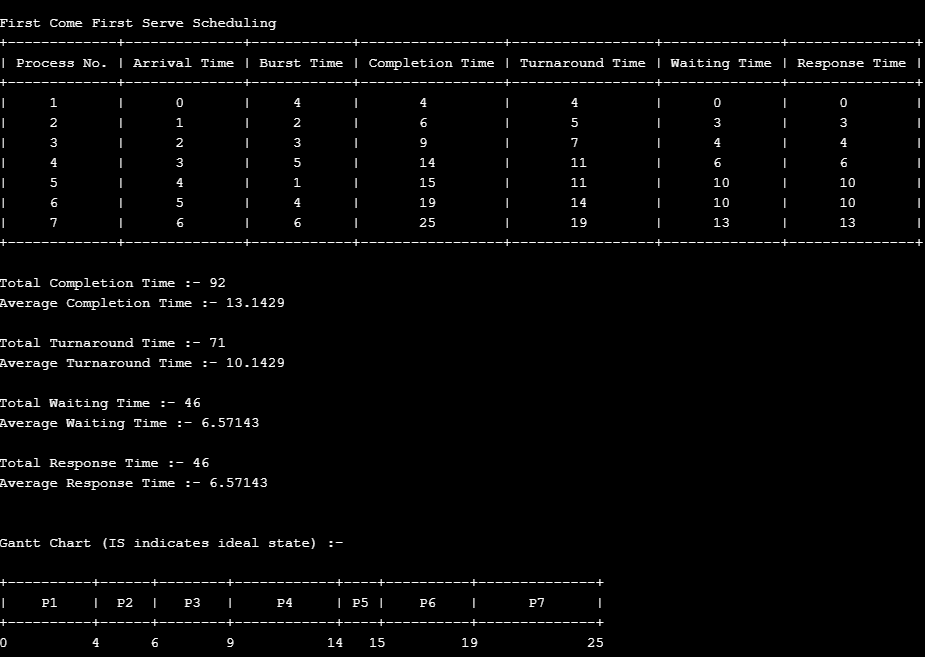
*->Step2*

First of all put the data in the ready queue so select the option 0 or option 1 for the input. Let first put the sample data by selecting option 0. After enter 0 the screen looks like this:



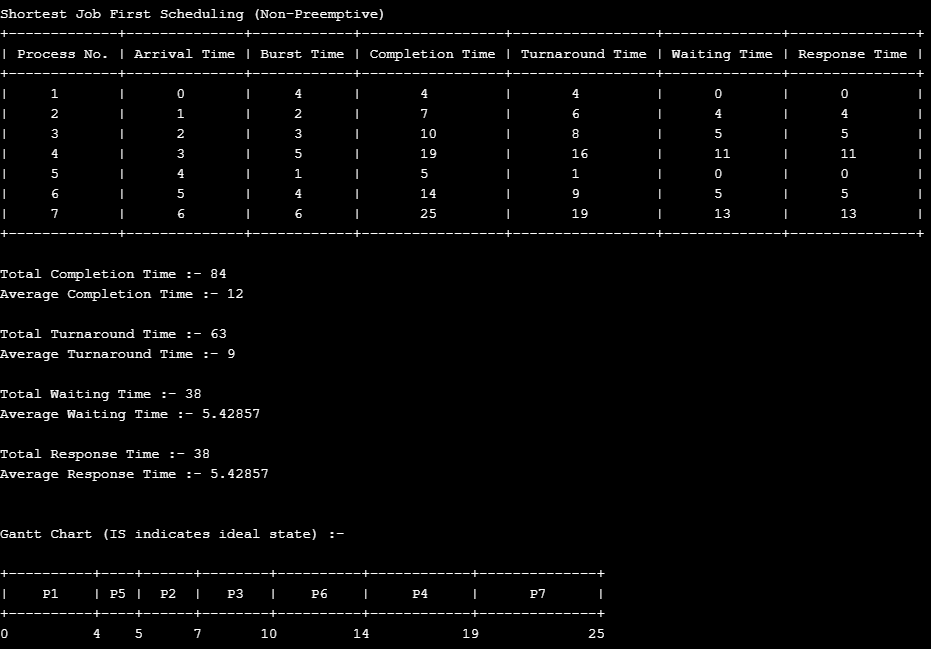
*->Step3*

Now select any option from 2 to 10 for scheduling according to the selected algorithm. Let select the option 2 for FCFS algorithm. The result screen looks like this:



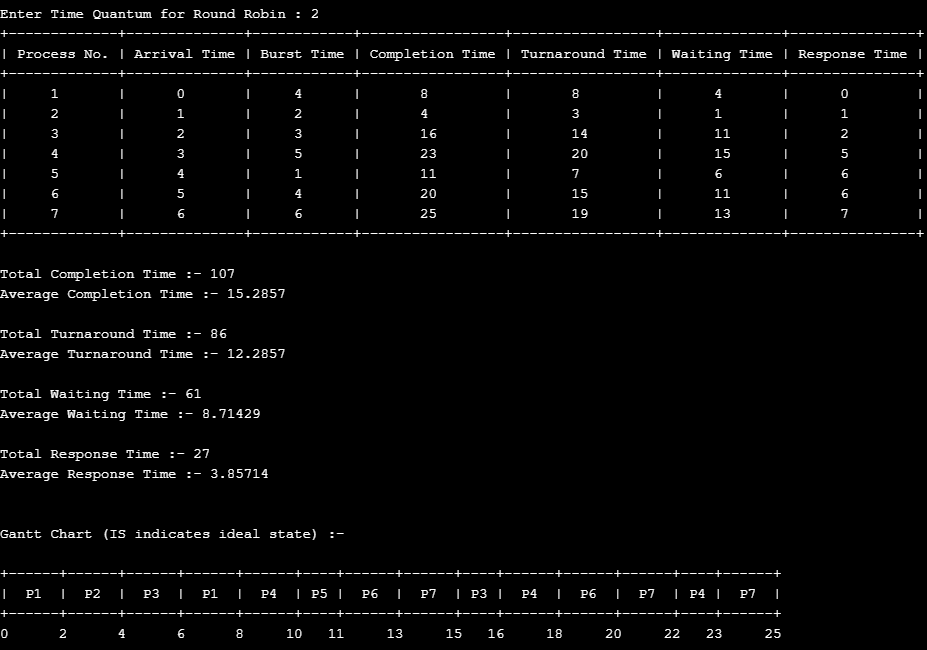
*->Step4*

Let again select some different option and select the option 4 for SJF (NP) algorithm. The result screen looks like this:



*->Step5*

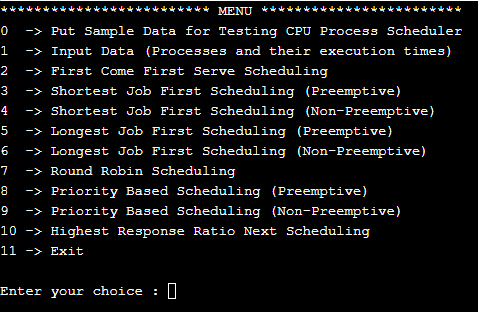
Let again select some different option and select the option 7 for RR algorithm. The round robin also take the time quantum as input. Put the time quantum 2. The result screen looks like this:



**a) For Input Data**

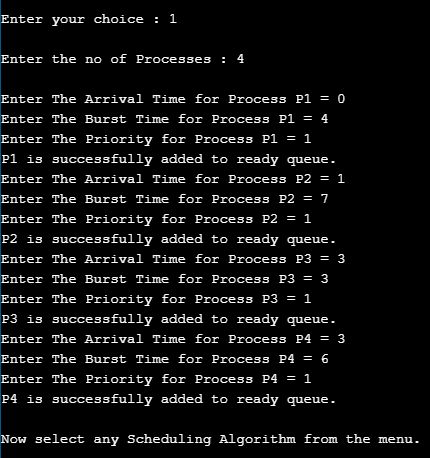
*->Step1*

The working screen show you the menu which looks like this:



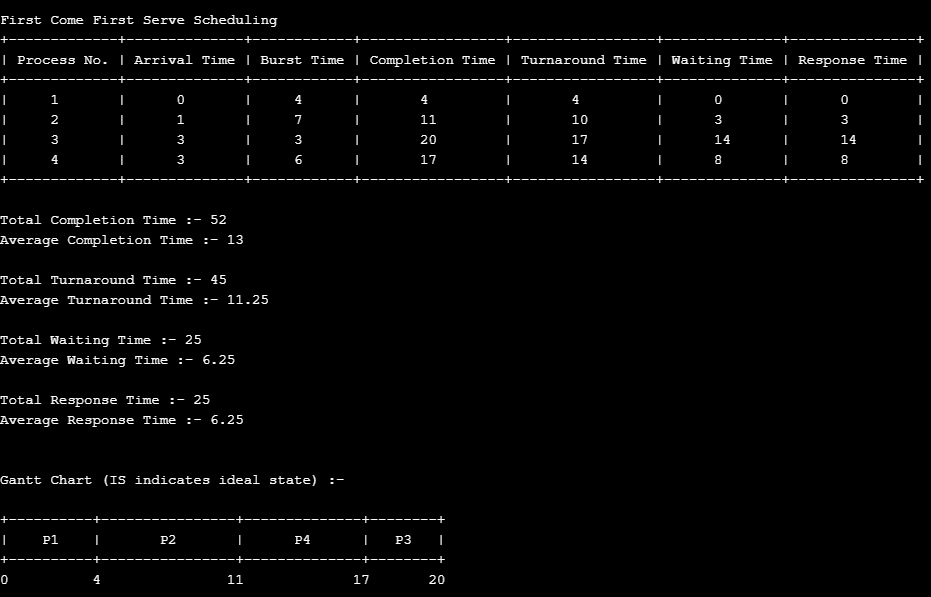
*->Step2*

First of all put the data in the ready queue so select the option 0 or option 1 for the input. Let take input the data from the user by selecting option 1. After enter 1 then select the no of processes you want to push in the ready queue. After this input the detail of each process which comprises of Arrival Time, Burst Time and the priority which is used in priority scheduling. The screen after taken the input from the user looks like this:



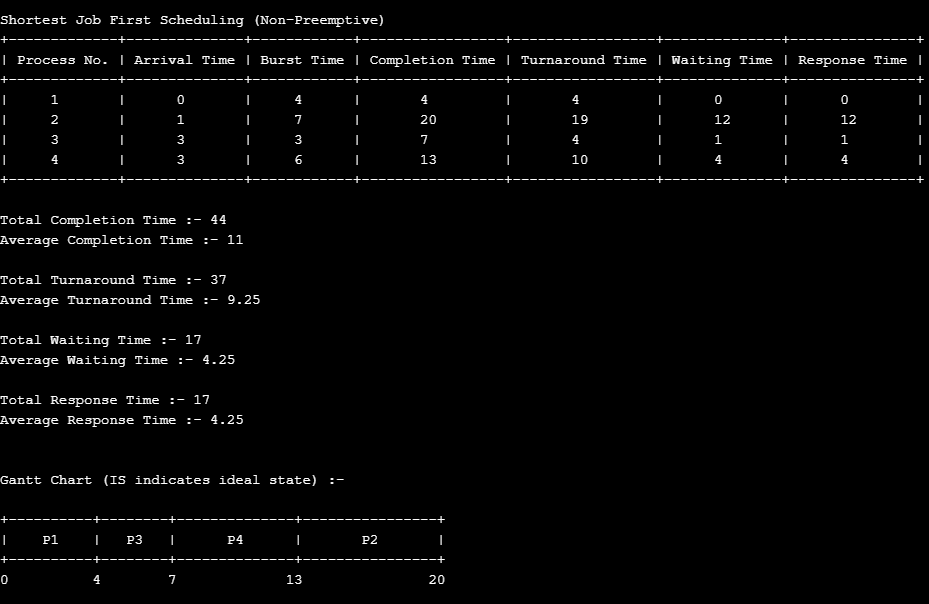
*->Step3*

Now select any option from 2 to 10 for scheduling according to the selected algorithm. Let select the option 2 for FCFS algorithm. The result screen looks like this:



*->Step4*

Let again select some different option and select the option 4 for SJF (NP) algorithm. The result screen looks like this:



*->Step5*

Let again select some different option and select the option 7 for RR algorithm. The round robin also take the time quantum as input. Put the time quantum 2. The result screen looks like this:

