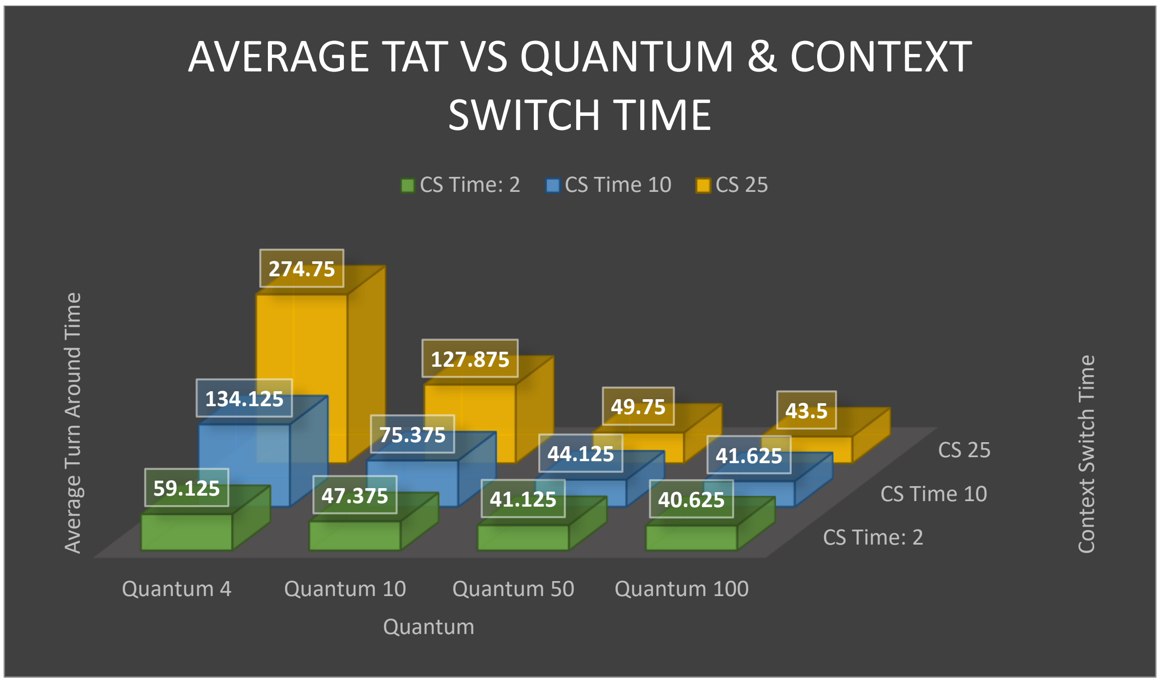
Homework 06

Ricardo Palma Mendoza A01226922

The main issue of this homework is to calculate the execution time of a number of processes using the round robin scheduler algorithm. In the code provided, a process is just represented by an integer that emulates its execution time. The purpose of this homework is to compare the differences of executing round robin with different quantum, context switch time and max size of the processes.

First, we will compare the difference in the round robin algorithm when changing the value of the quantum and the context switch time, and how this impacts the average turn-around time. One knows that the quantum is the period a process is allowed to be in execution mode, while context switch time is the amount of unit of time it takes to save the PCB of a process if such did not finish. For the sake of this experiment, the values 20,15,3, 34,55,23,43,130; will be used.

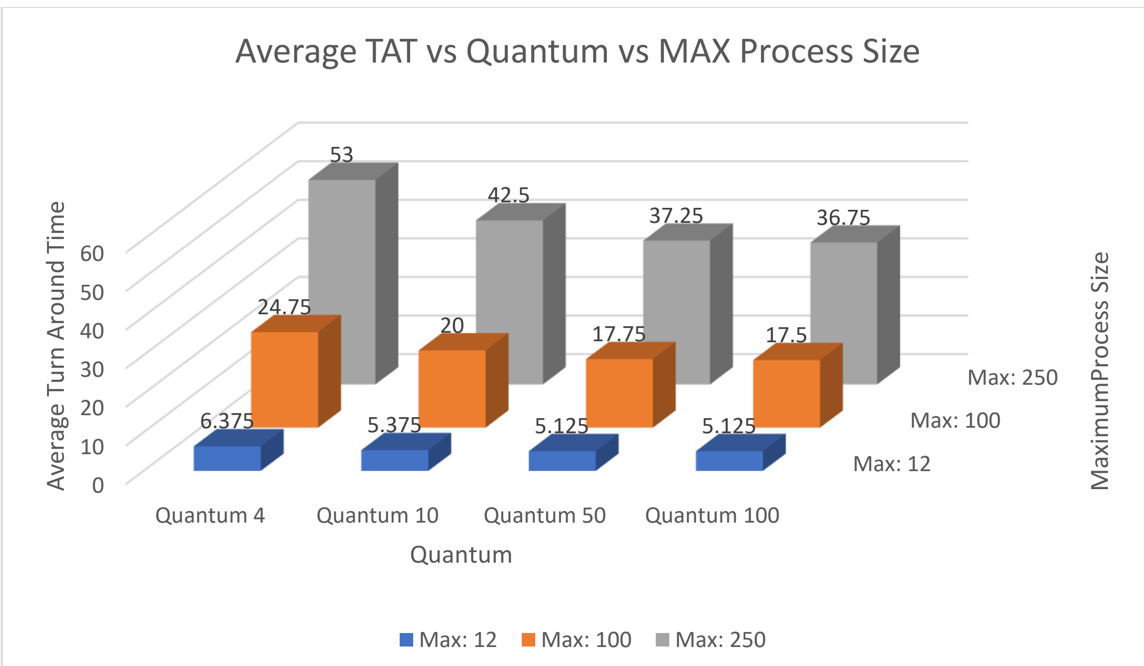


Quantum	Number of context switch
4	75
10	28
50	3
100	1

An impact in the Average Turn-Around Time is noticed when a higher Quantum value is used for the array. Since the array consist in processes with extreme values of execution time, the quantum would cut the same process for several occasions if such quantum has a small value. As noticed in the table, the lesser the quantum is, the more context switches are given. Now, since there is a higher number of context switches, this would mean how much would the context switch time would impact in the ATAT. If the context switch time takes an amount of 25 units of time, the ATAT with Quantum 4 would suffer from a peak in Total Turn Around Time.

As a conclusion, one could mention the importance of handling processes with long durations. The main purpose of the round robin algorithm is to assure long lasting processes do not take over the CPU for long, so that short lasting processes can be executed in between. However, as the experiment demonstrated, there is backfire in such algorithm. The importance of the quantum highly depends of the size of the processes and the time it takes to perform a context switch. If a quantum of such size performs several context switches this will decrease the performance of the algorithm, thus it will not be the best choice. On the other side, if an oversized quantum is used, there may be occasions when no process will be cut, hence there would be no point in such algorithm. That is why there needs to be a balance between quantum and the expected Burst Time of a process taking in consideration the cost of a context switch.

The next data use for testing consist in an array of 8 elements, with the values: 2,3,4,12,8,5,6,1; in that order. For the sake of the experiment, the last element, in this case 1, will be changed to a max value defined in the graphs (except for the first case, since 12 is already in the array). This is done because the experiment consists in testing different maximum process sizes. In addition, the context switch between one process to another, if such process was not finished, takes a total of 2 units of time. The first graph compares the changes between the average turnaround time, quantum and maximum process size:



From this graph one can notice the enormous difference between running a set of processes when the maximum length of such set is 12 units, compared to a set with a maximum of 250 units of time. One of the biggest impacts when handling the set of processes, is the quantum defined for such round robin algorithm. If one defines a small quantum for a set with processes of enormous Burst Time, the Average Turn Around Time will increase peak in extreme values. The main reason of such increase is caused by the context switch and how much units of time this cost. Because a process is to large to be finished in 1 or even 50 cuts, a context switch will be in constant execution; therefore, it will cause a major impact in the ATAT.