**A Task Scheduling Algorithm with Improved Makespan Based on Prediction of Tasks Computation Time algorithm for Cloud Computing**

In this paper new scheduling algorithm called Prediction of Tasks Computation Time algorithm (PTCT) to estimate minimum task execution time/Makespan time for cloud computing environment. Now-a-days all cloud service providers providing all resources to end users at very cheap rate and at the same time by designing scheduling algorithms cloud service providers are ensuring that all users can get response data in quick time. Various scheduling algorithms are implemented in cloud environment such MINMIN, MAXMIN, QOS GUIDE etc.

MINMIN algorithm will schedule all task with less execution time first and then schedule remaining task. In simple terms MINMIN algorithm give priority to less execution time task.

MAXMIN algorithm will schedule all task with more execution time and then schedule small execution time task. In simple terms MAXMIN give priority to high execution time first.

Many more scheduling algorithms are there but above two algorithms are very much popular. This two algorithms will not look for resources which can take minimum execution time and propose PTCT algorithm will look for all resources/processors/machines and then form a matrix which contains estimated execution time for all jobs and then by applying PCA (Principal Component Analysis) algorithm it will predict or choose resource which took minimum execution time and then assign new task to that selected minimum execution time resource. Here resource could be computer or processor or Virtual Machine.

In propose PTCT algorithm we build an array with all task and processors as Directed Acyclic Graph (DAG) and then build a matrix with all processors and task. A matrix will contains estimated execution task time on each processor and all rows of a matrix will filled with all processors execution time for all tasks. On generated matrix we will apply PCA algorithm to choose processor which take less execution time for selected task. This process continues till all task assigned to all processors.

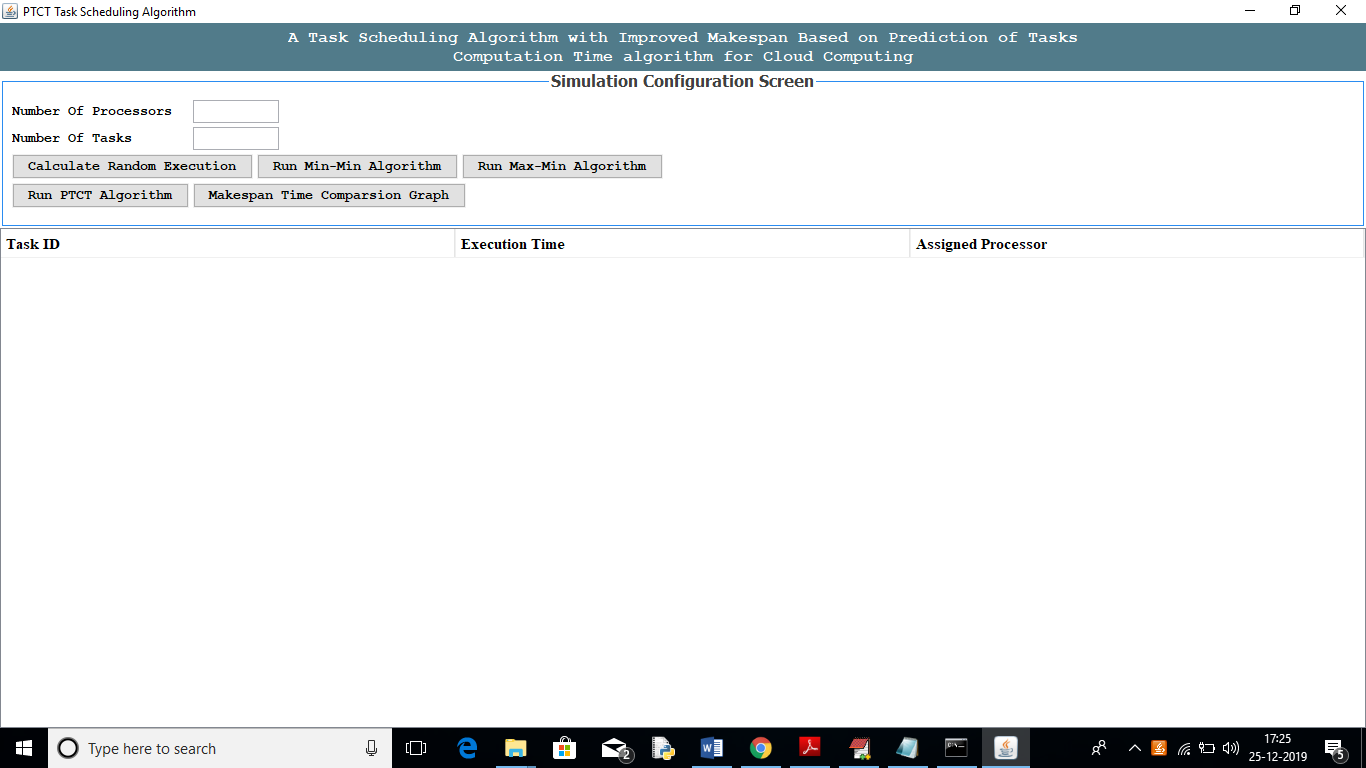
By applying PTCT algorithm we can further decrease computation and communication cost at cloud side.

To implement this paper we design 3 algorithms in the form of simulation and then compare execution/Makespan time between them. In all 3 algorithms PTCT algorithm took less execution time for all tasks.

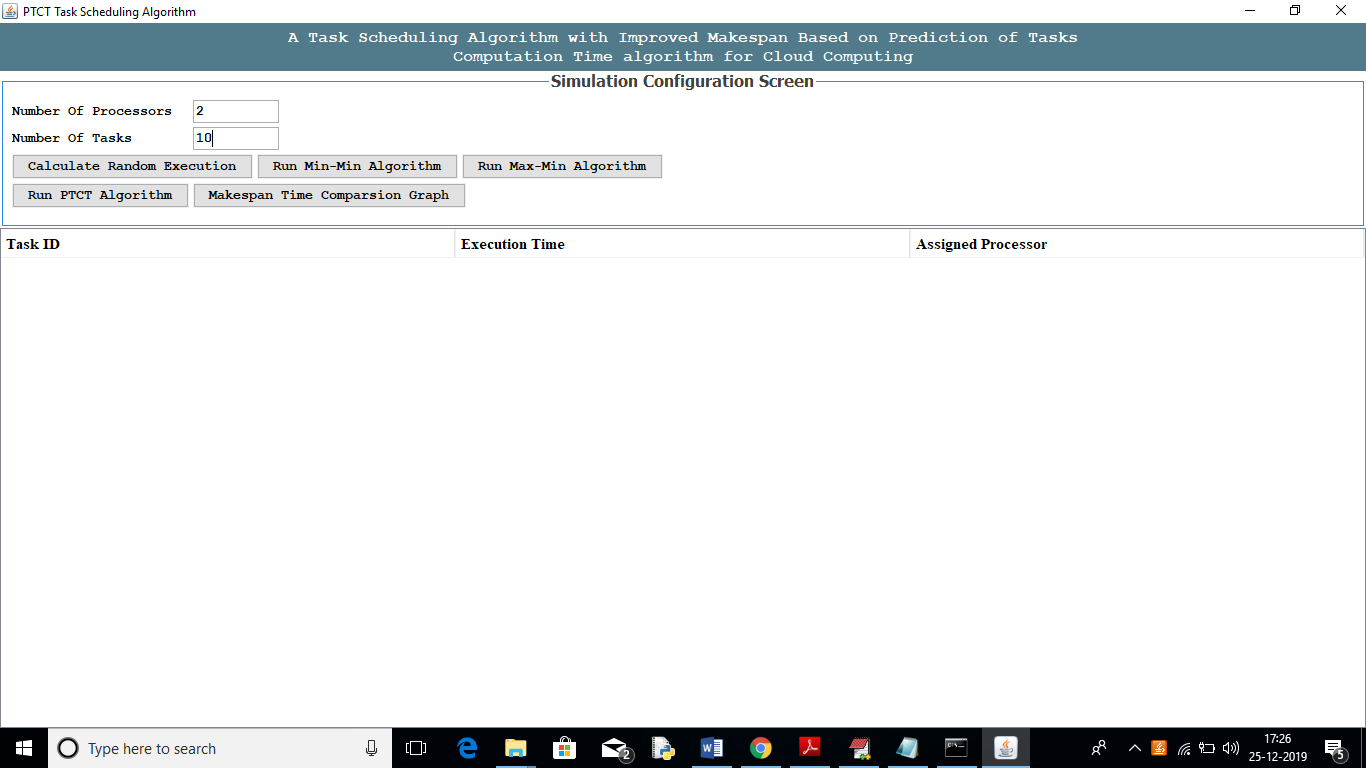
Abstract:

Cloud computing is extensively used in a variety of applications and domains, however task and resource scheduling remains an area that requires improvement. Put simply, in a heterogeneous computing system, task scheduling algorithms, which allow the transfer of incoming tasks to machines, are needed to satisfy high performance data mapping requirements. The appropriate mapping between resources and tasks reduces Makespan and maximises resource utilisation. In this contribution, we present a novel scheduling algorithm using Directed Acyclic Graph (DAG) based on the Prediction of Tasks Computation Time algorithm (PTCT) to estimate the preeminent scheduling algorithm for prominent cloud data. In addition, the proposed algorithm provides a significant improvement with respect to the Makespan and reduces the computation and complexity via employing Principle Components Analysis (PCA) and reducing the Expected Time to Compute (ETC) matrix. Simulation results confirm the superior performance of the algorithm for heterogeneous systems in terms of efficiency, speedup and schedule length ratio, when compared to the state-of-the-art Min-Min, Max-Min etc.

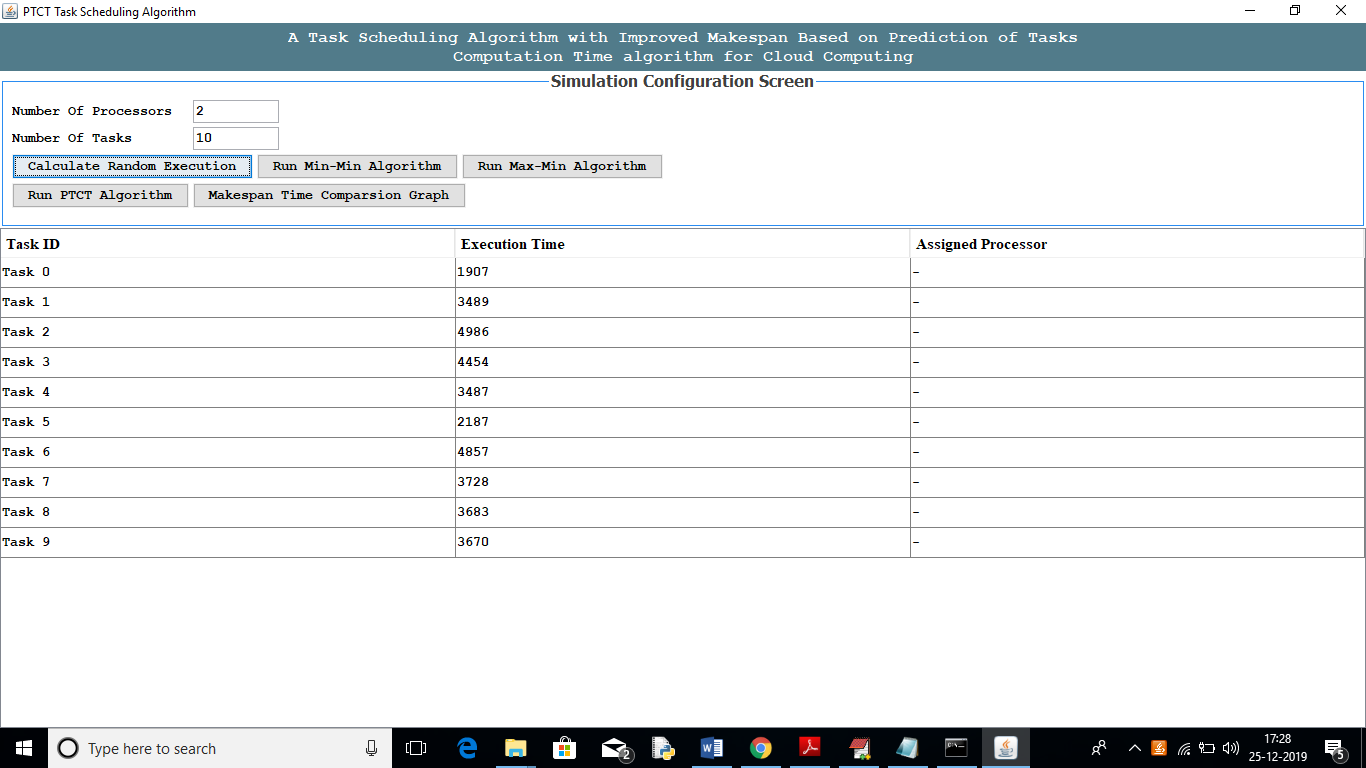
Screen shots



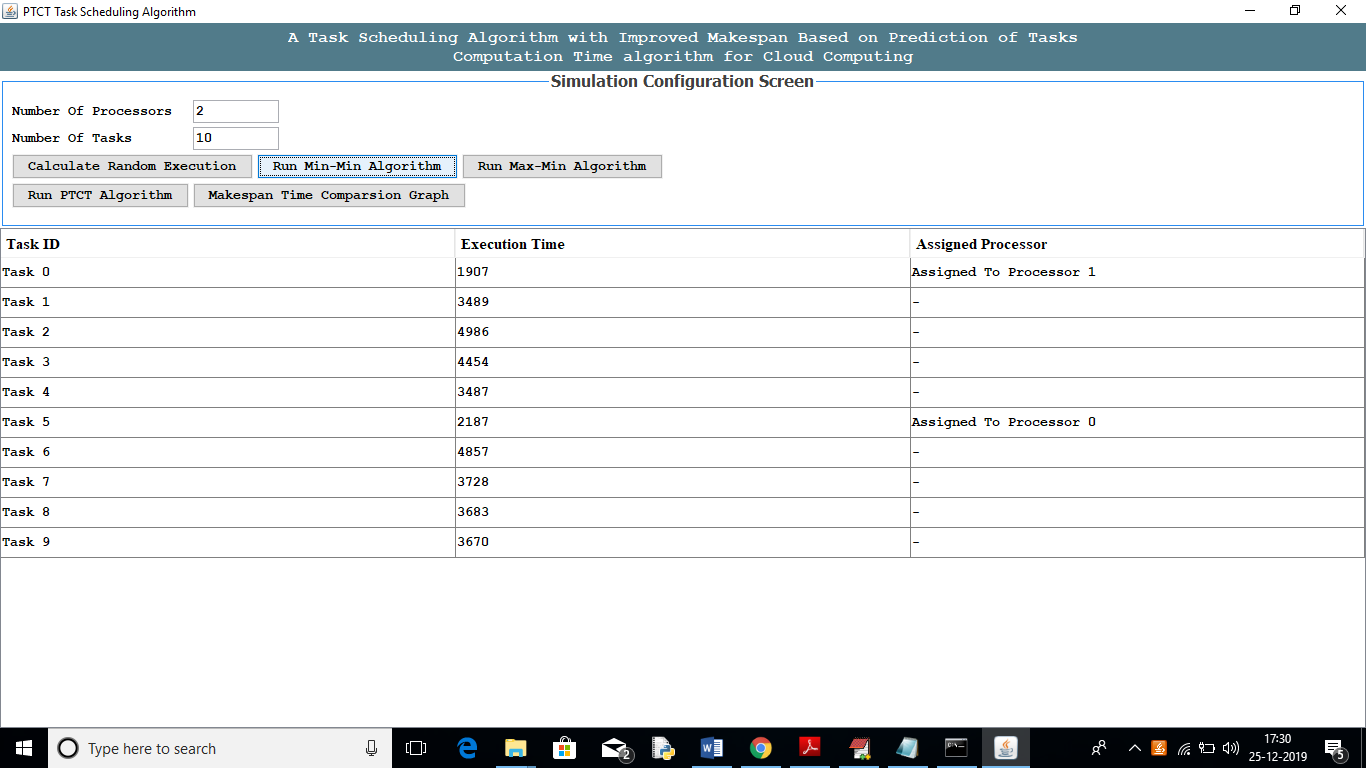
In above screen enter number of processors and number of tasks

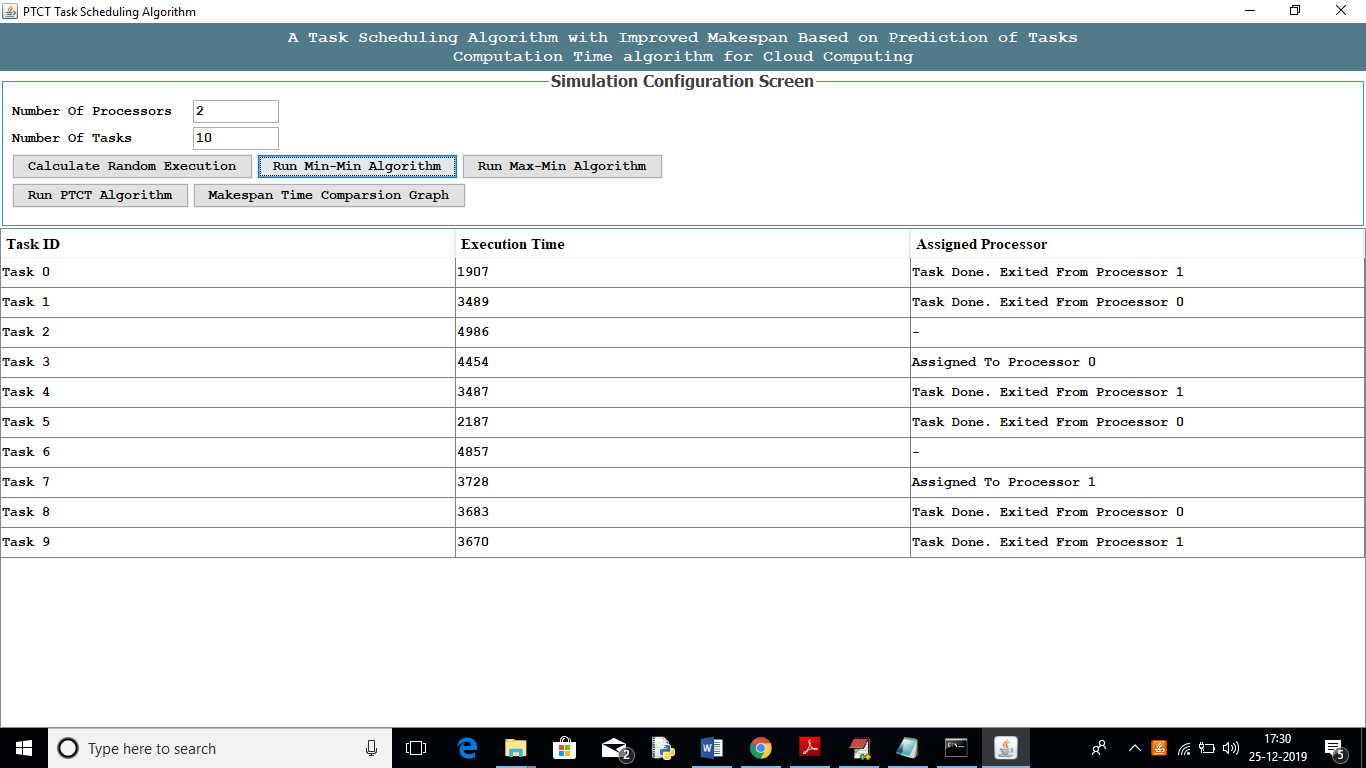


In above screen I entered number of processors as 2 and number of task as 10 which means all 10 tasks has to schedule and run in given 2 processors. Now click on ‘Calculate Random Execution Time’ button to assign some execution time to each task and based on this execution time algorithms will schedule tasks to processors. See below screen

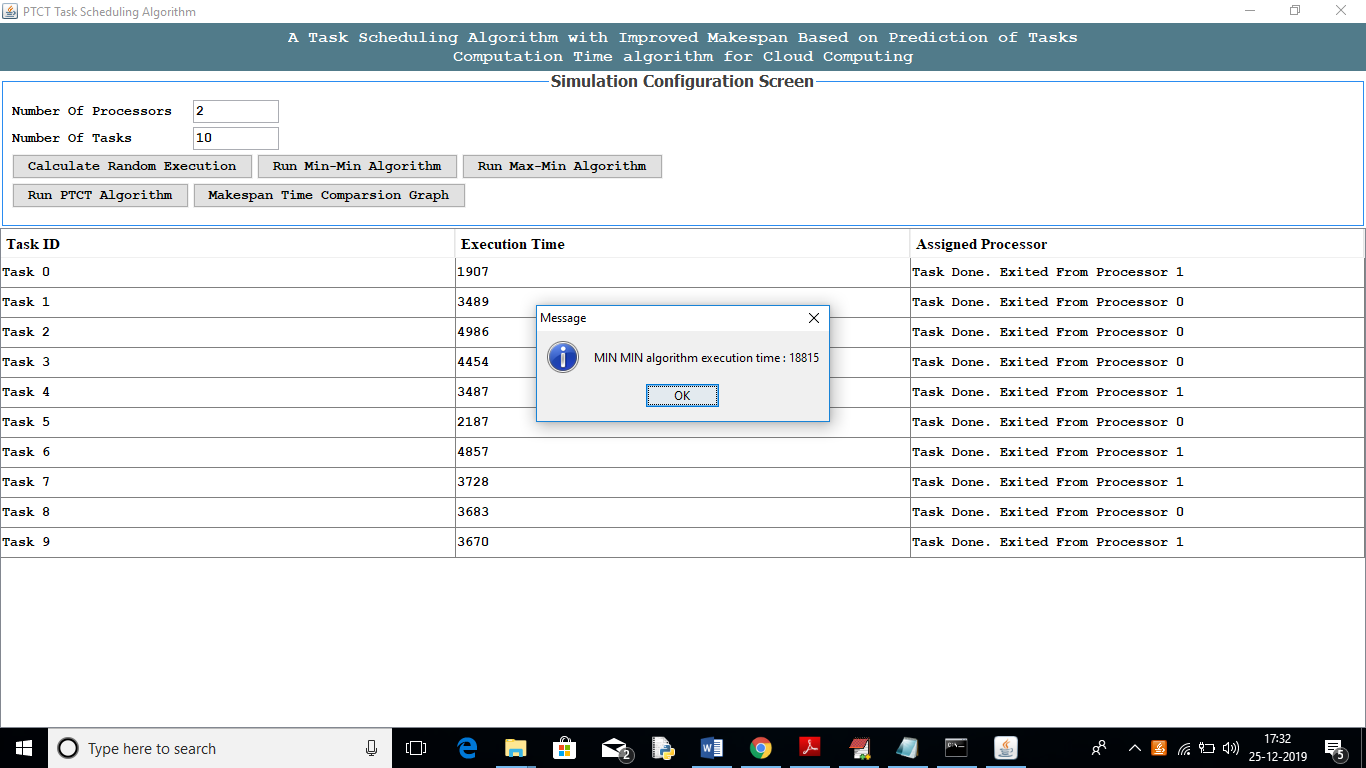


In above screen all 10 tasks got some random execution time and now click on ‘Run Min-Min Algorithm’ button to schedule this 10 tasks to 2 processors. We can see MINMIN will schedule less execution time task first. In third column empty value is there as processor not yet assign to task.

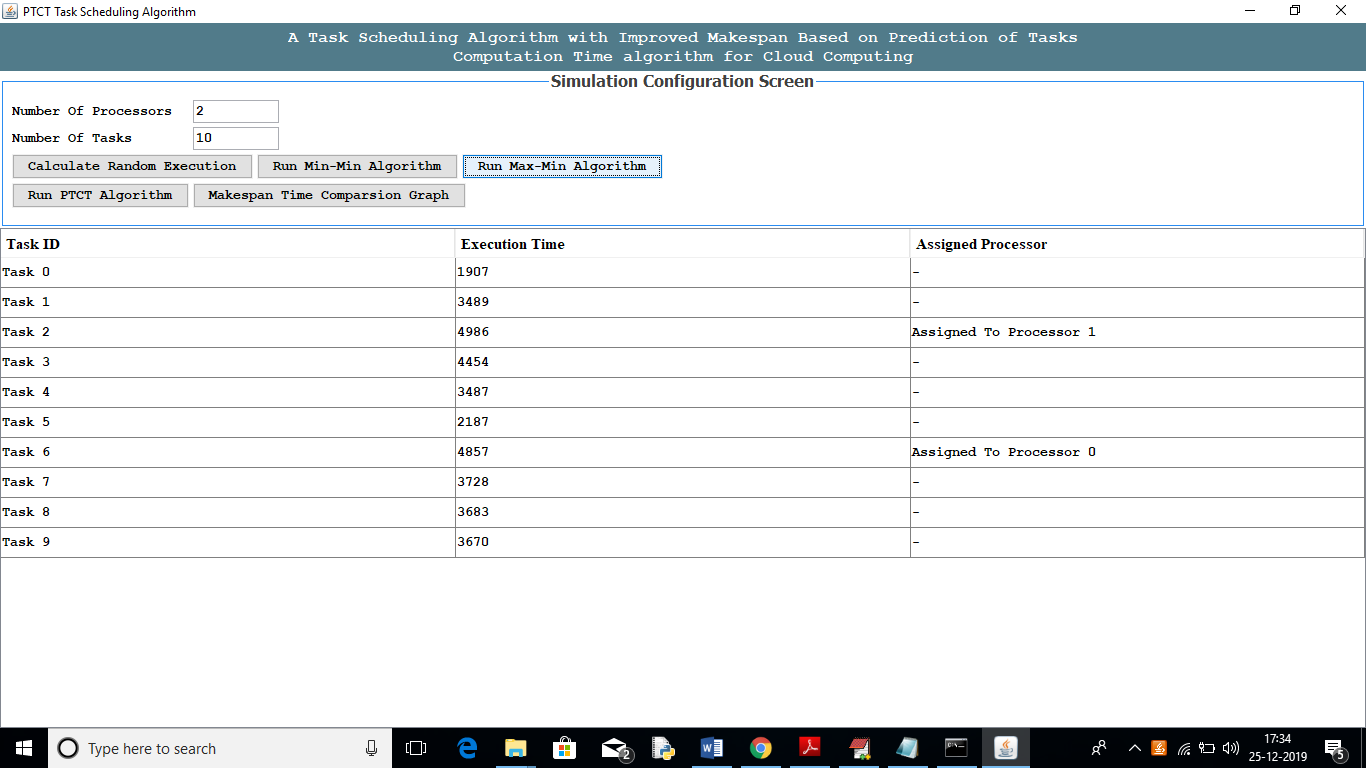


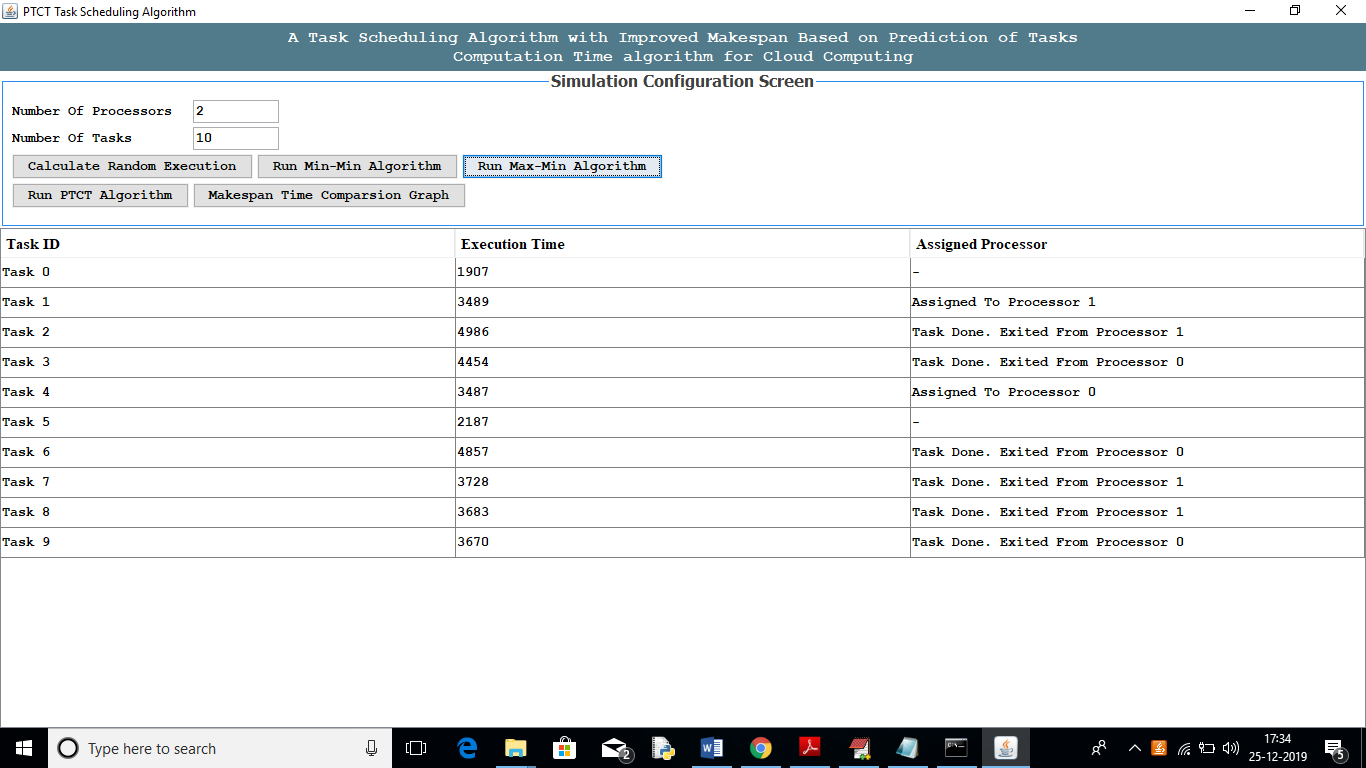


In above 2 screen we can see MINMIN scheduling task based on freeness of resource and execution time. In third column we can see task is assign to which processor and after task completion we will get message as task done on which processor. After all task execution we will get total execution time for all tasks. See below screen

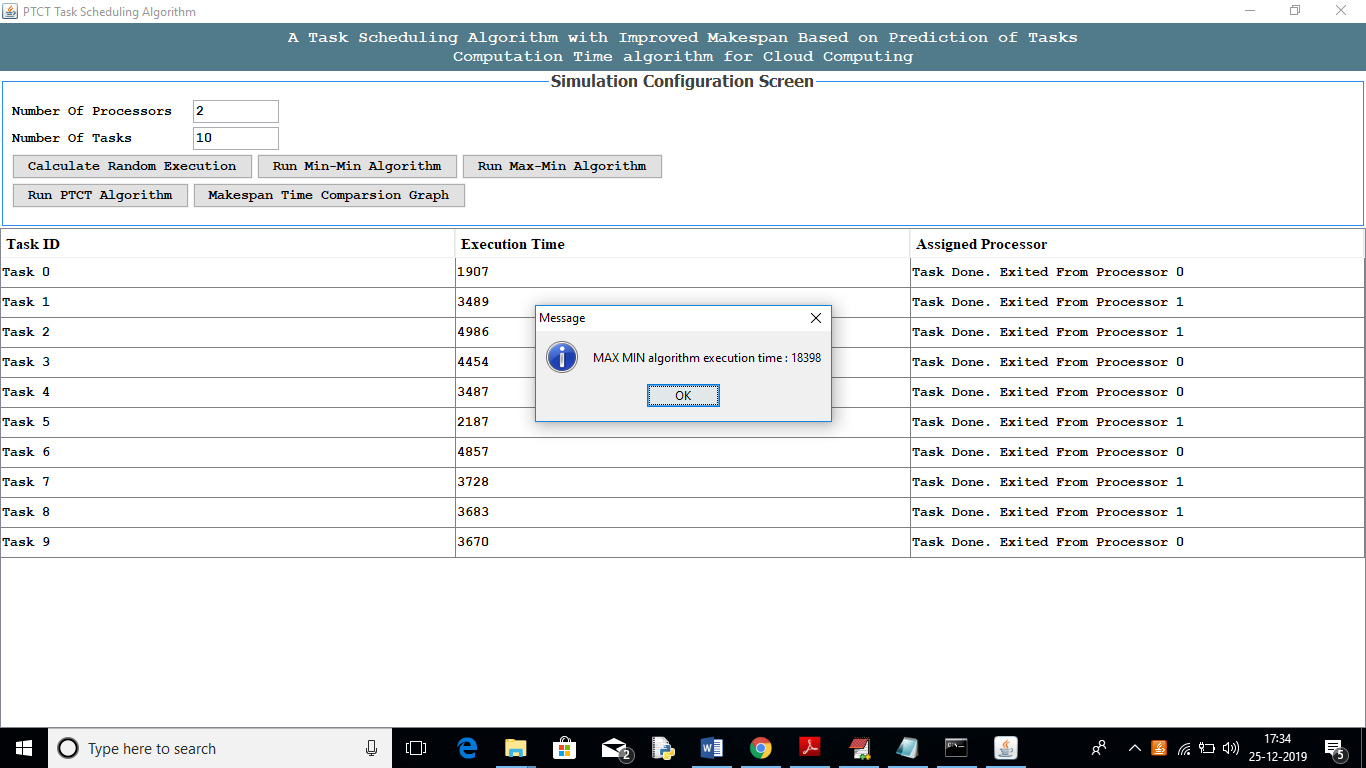


In above screen we can see MINMIN took 18815 milli seconds to complete all tasks. Similarly click on ‘Run Max-Min Algorithm’ button to schedule all tasks based on MAXMIN algorithm

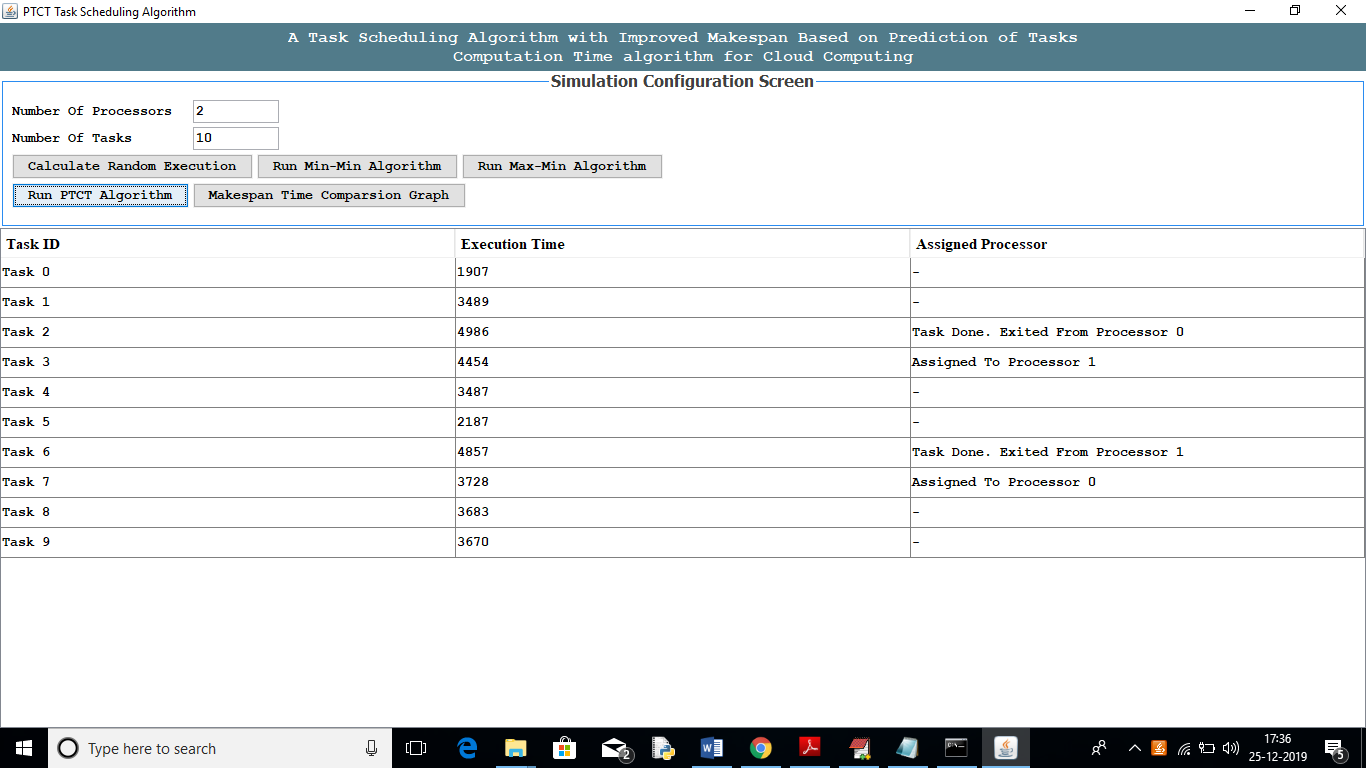


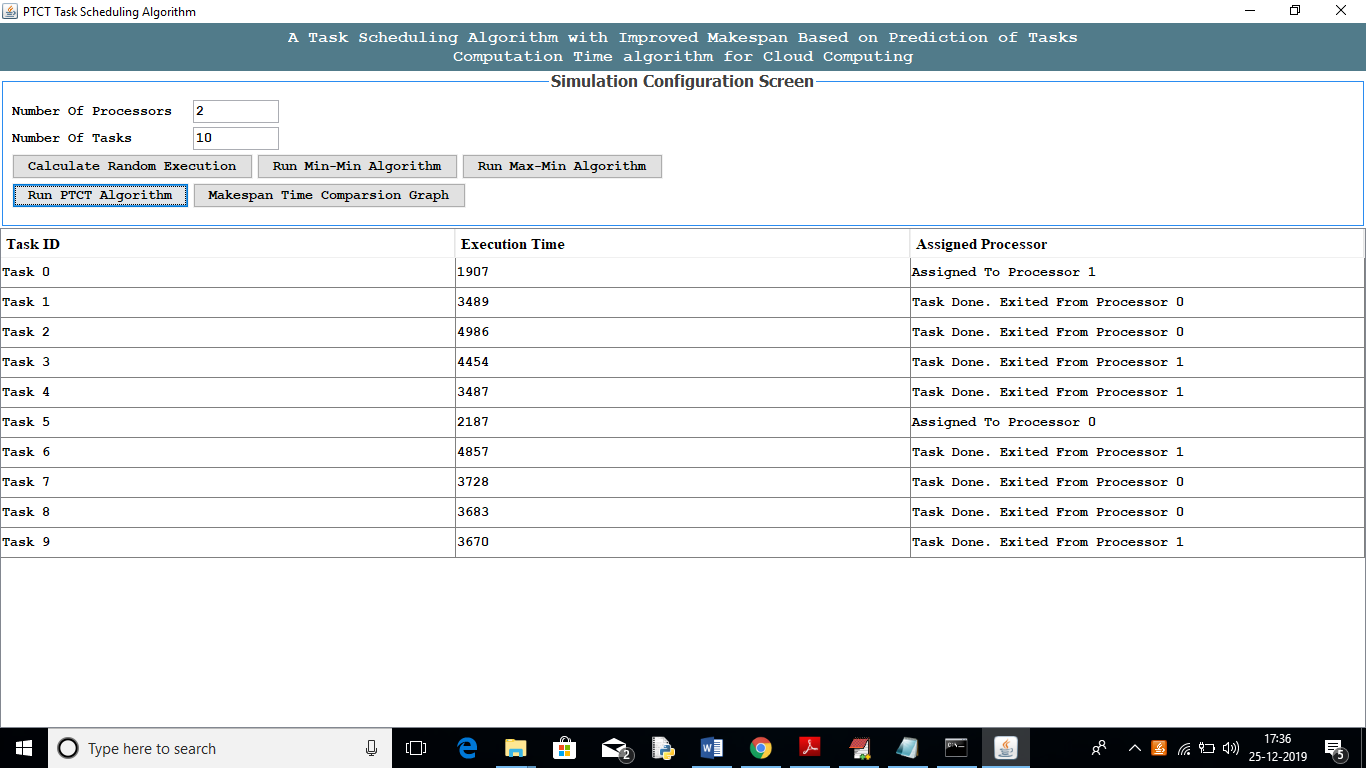


Above two screen showing scheduling process of MAXMIN algorithm and below is MAXMIN algorithm total execution time

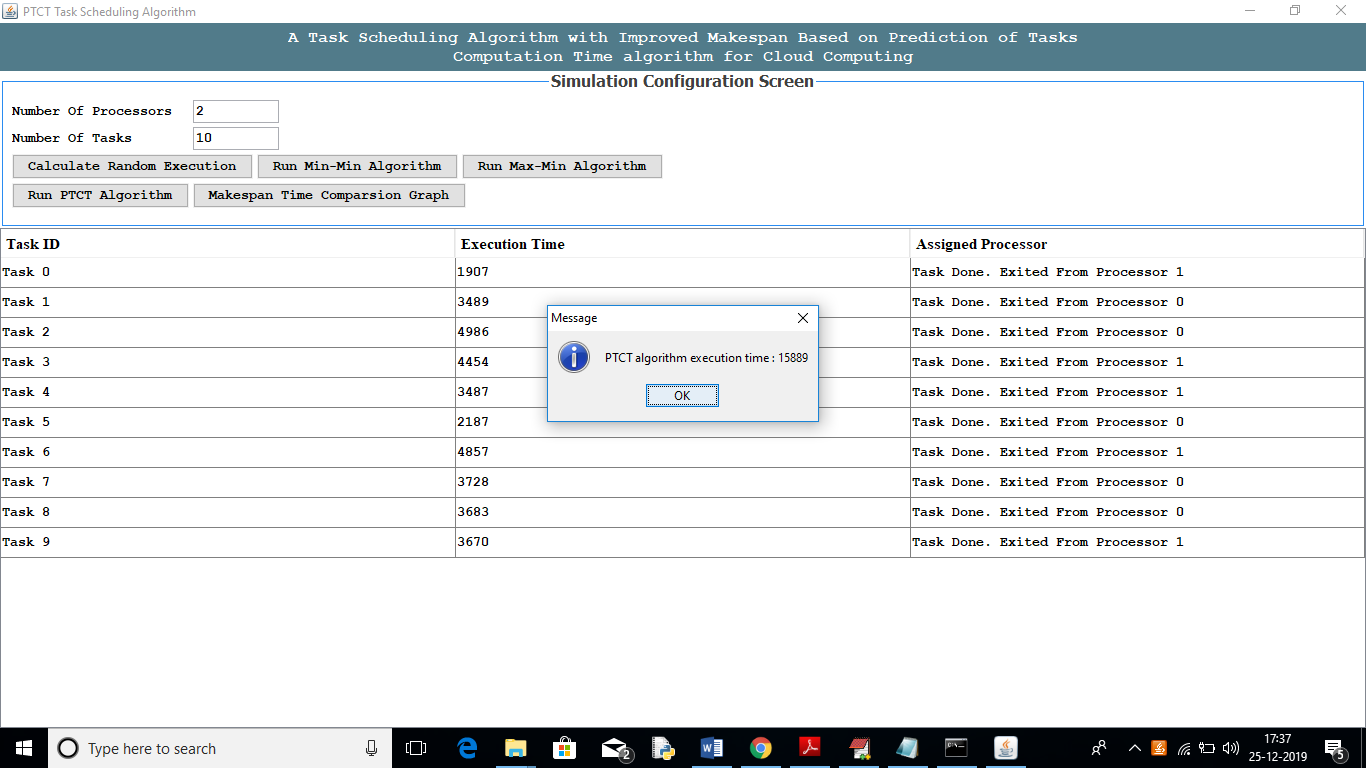


In above screen we can see MAXMIN took 18398 Milli Seconds to complete all tasks and we can say MAXMIN took less time compare to MINMIN. Similarly click on ‘Run PTCT Algorithm’ button to schedule tasks based on PTCT algorithm concept.

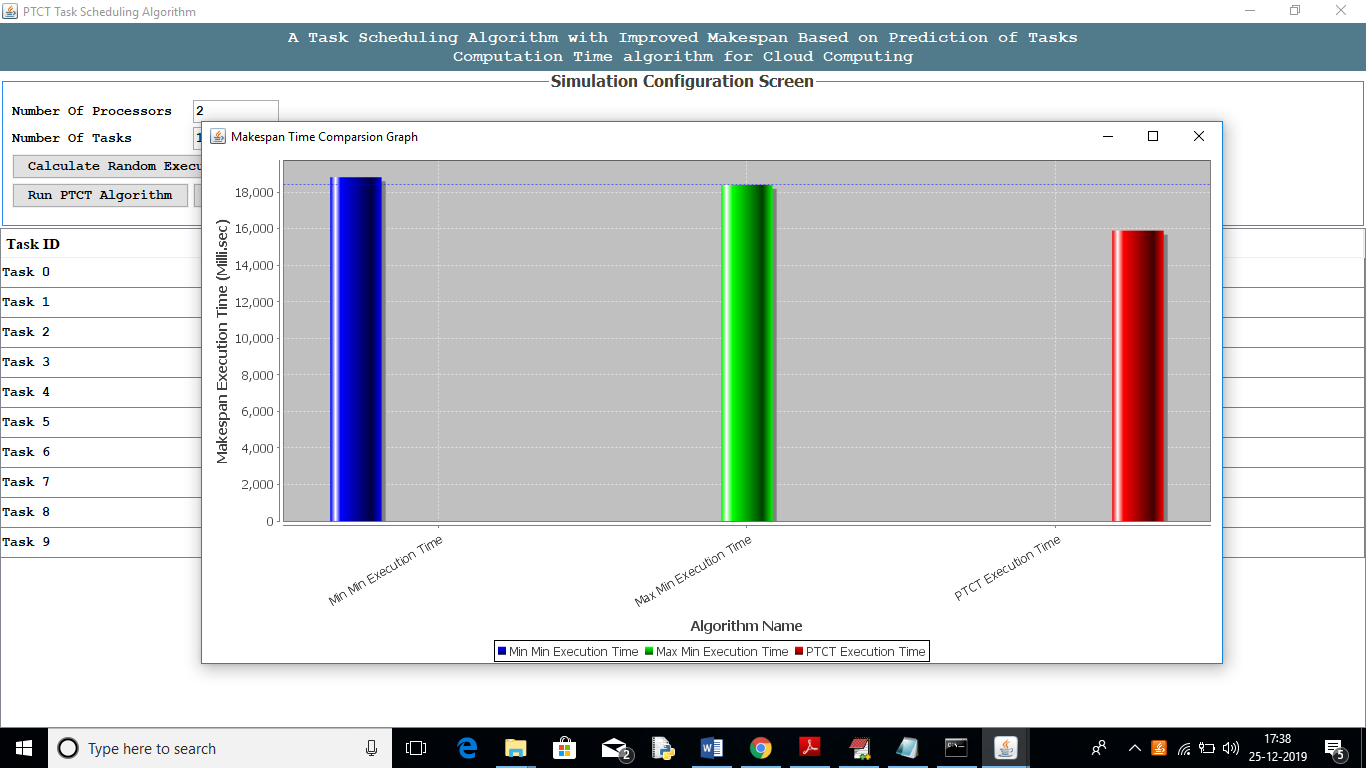




Above two screen showing PTCT scheduling output and below is total PTCT execution time



In above screen we can see PTCT took 15889 Milli Seconds to complete all tasks execution and is better than other 2 algorithms. Now click on ‘Makespan Time Comparison Graph’ button to see all algorithms execution time graph



In above graph x-axis represents algorithm name and y-axis represents execution time in Milli seconds. From above graph we can conclude PTCT propose algorithm better than other 2 algorithms.

This code is dynamic so u can give any number of tasks and processors