## Practical 4- Task Scheduling

**Name:**Tanisha Agrawal

**Roll No**.:A13

## Problem Statement:

Implementation of FCFS, SJF, Round Robin scheduling mechanisms using Cloudsim cloud simulator.

(Output from page 5)

## Abstract:

The process of deciding which task will utilize the cpu time is called task scheduling. The scheduling of the task may be on the basis of their priorities. The priority assignment mechanism for the tasks can be either static or dynamic. In the case of static priority assignment the priority of task is assigned as soon as the task is created, thereafter the priorities cannot be changed. In dynamic assigning the priorities of the task can be changed during the runtime. Efficient provisioning of resources and scheduling of resources as well as tasks will ensure:

1. Resources are easily available on demand.
2. Resources are efficiently utilized under conditions of high/low load.
3. Energy is saved in case of low load (i.e. when usage of cloud resources is below certain threshold).
4. Cost of using resources is reduced.

For measuring the efficiency and effectiveness of Load Balancing algorithms simulation environments are required. CloudSim is the most efficient tool that can be used for modeling of Cloud. During the lifecycle of a Cloud, CloudSim allows VMs to be managed by hosts which in turn are managed by data centers. A simulation of First Come First Serve (FCFS), Round Robin and Shortest Job First (SJF) has been provided using the above mentioned tool.

## Tools and language used:

**Cloud Simulator:** Cloudsim 3.0.3

**IDE:** Netbeans

**Programming Language:** Java 8

**Setup:**

* Cloudsim 3.0.3: <https://github.com/Cloudslab/cloudsim/releases/tag/cloudsim-3.0.3>
* Cloudsim on Netbeans: <https://cloudsim-setup.blogspot.com/2012/01/cloudsim-installation-with-netbeans.html>

## Introduction:

CloudSim is a library for the simulation of cloud scenarios. It provides essential classes for describing data centres, computational resources, virtual machines, applications, users, and policies for the management of various parts of the system such as scheduling and provisioning.

Using these components, it is easy to evaluate new strategies governing the use of clouds, while considering policies, scheduling algorithms, load balancing policies, etc.

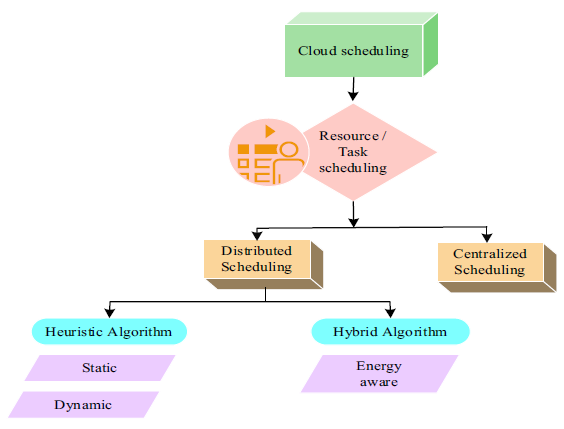
The CloudSim layer provides support for modelling and simulation of cloud environments including dedicated management interfaces for memory, storage, bandwidth and VMs.

By using CloudSim, organisations, R&D centres and industry-based developers can test the performance of a newly developed application in a controlled and easy to set-up environment.

Task scheduling can be done in two modes:

1. Space shared
2. Time shared

* Both hosts and VM can be provisioned to users either in space shared mode or time shared mode.
* In space sharing mode resources are allocated until the task does not undergo complete execution (i.e. resources are not preempted); whereas in time sharing mode resources are continuously preempted till task undergoes completion.



## 

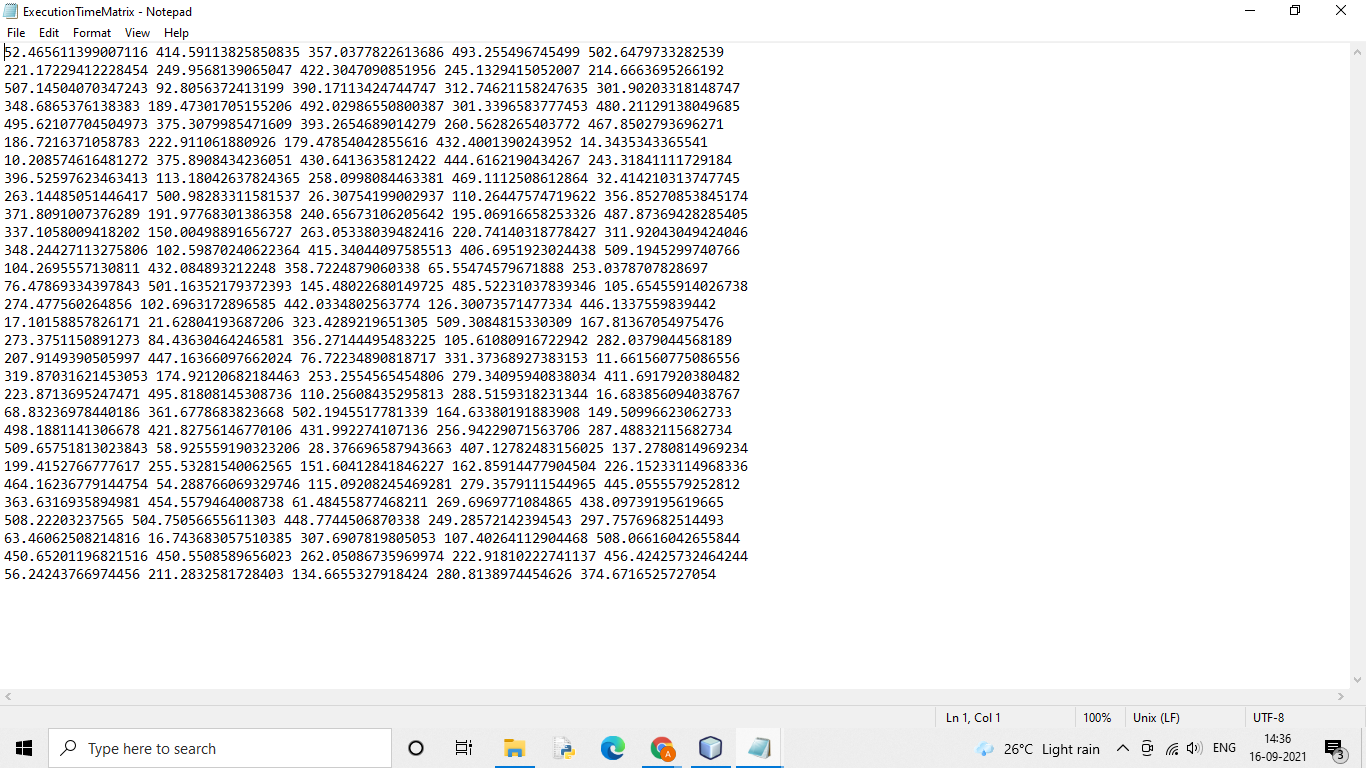
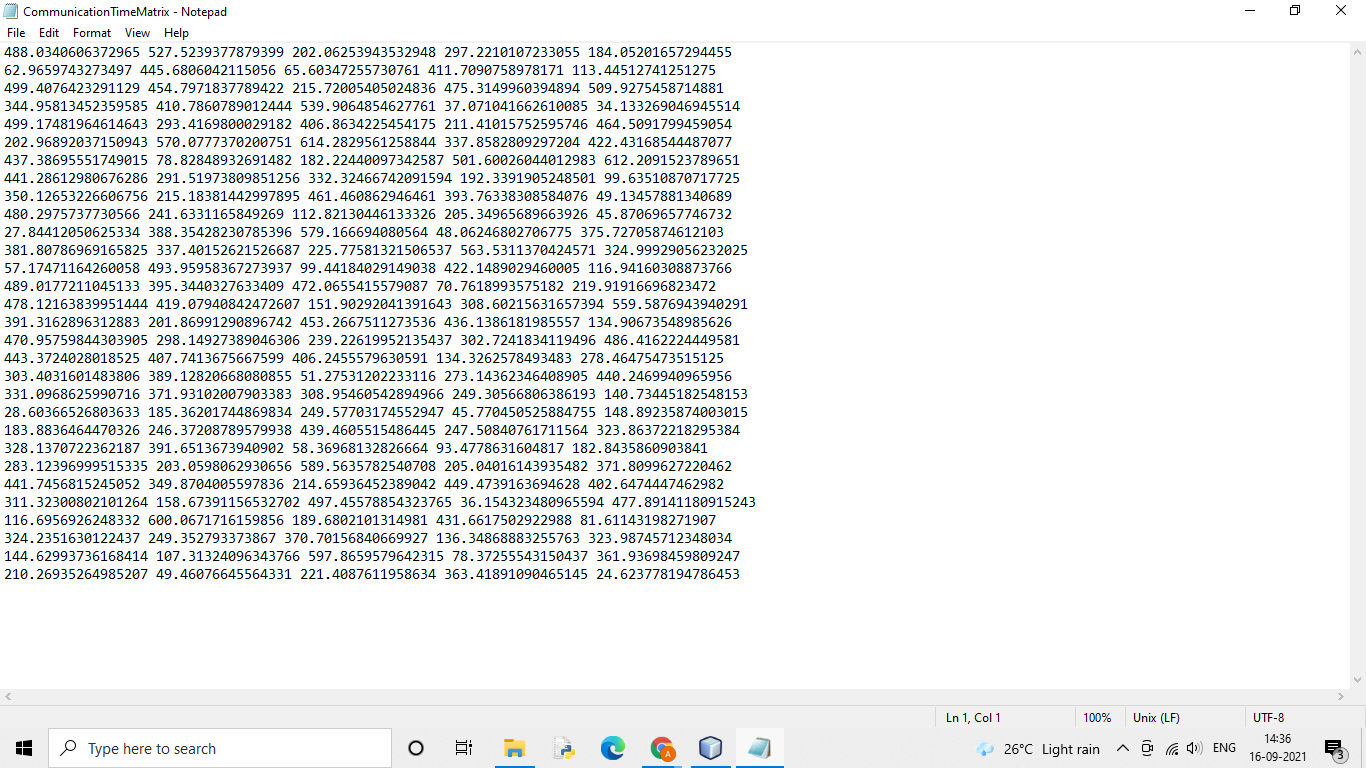
## Implementation:

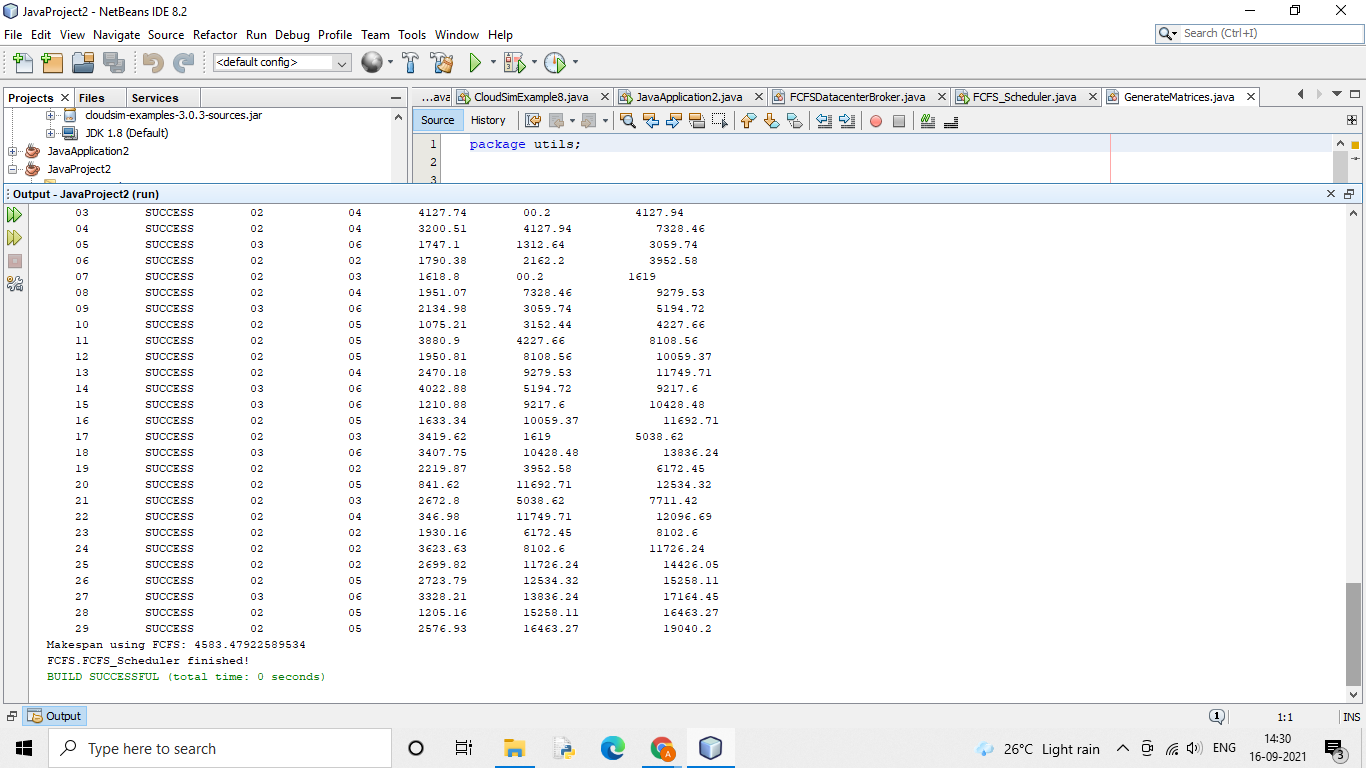
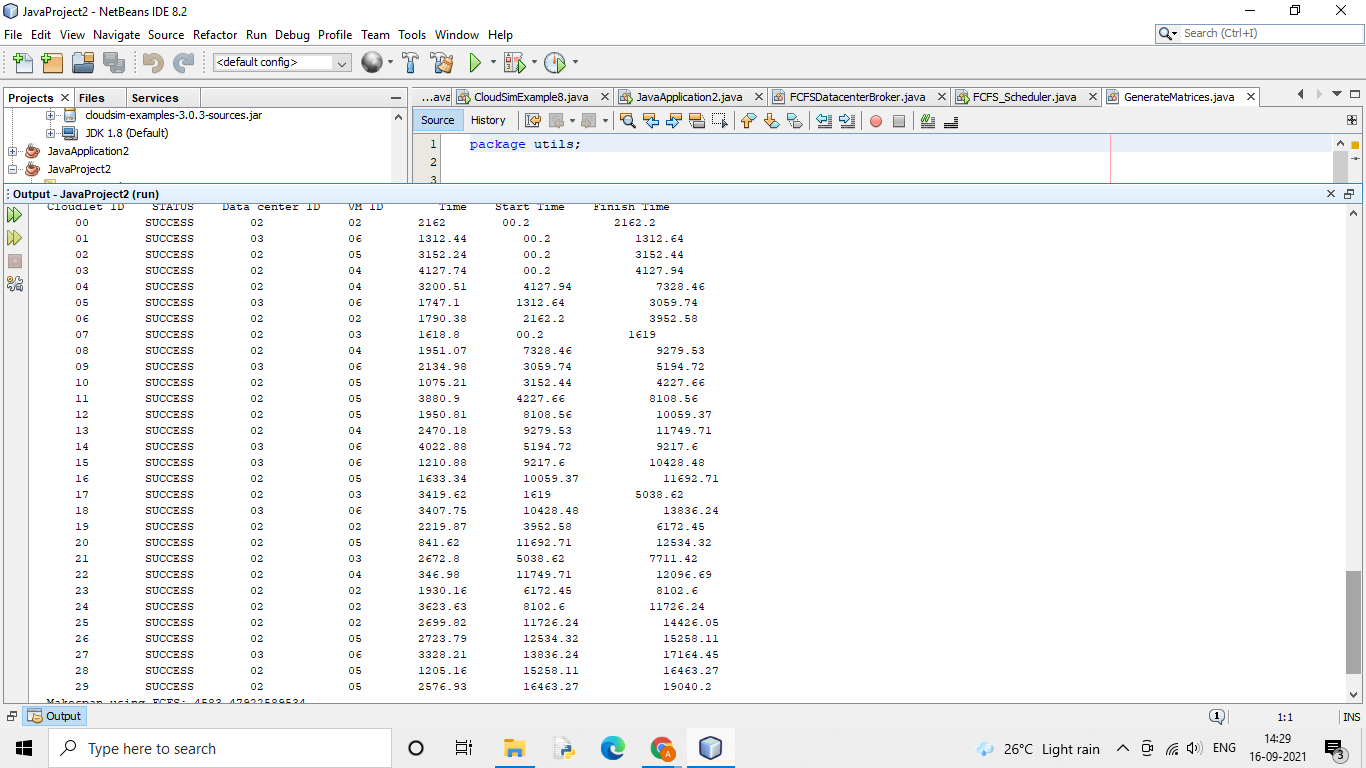
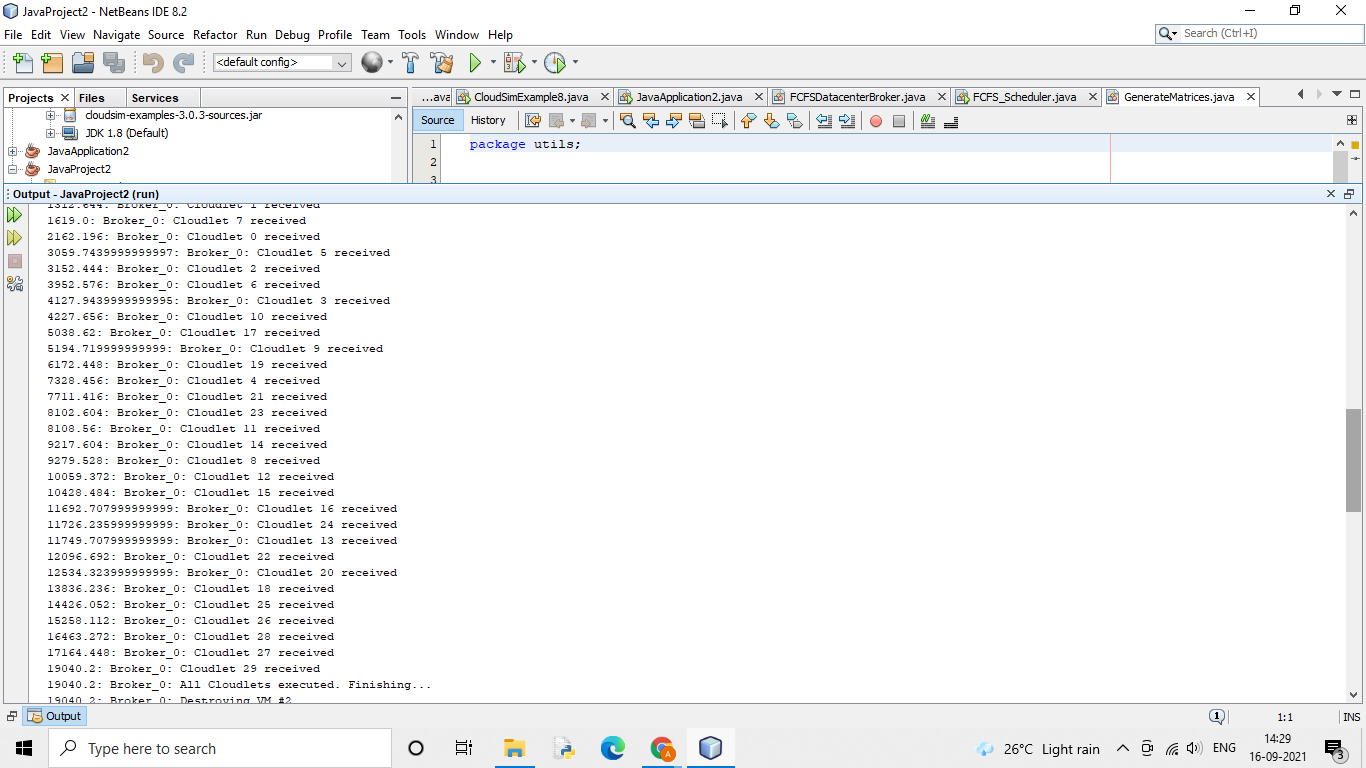
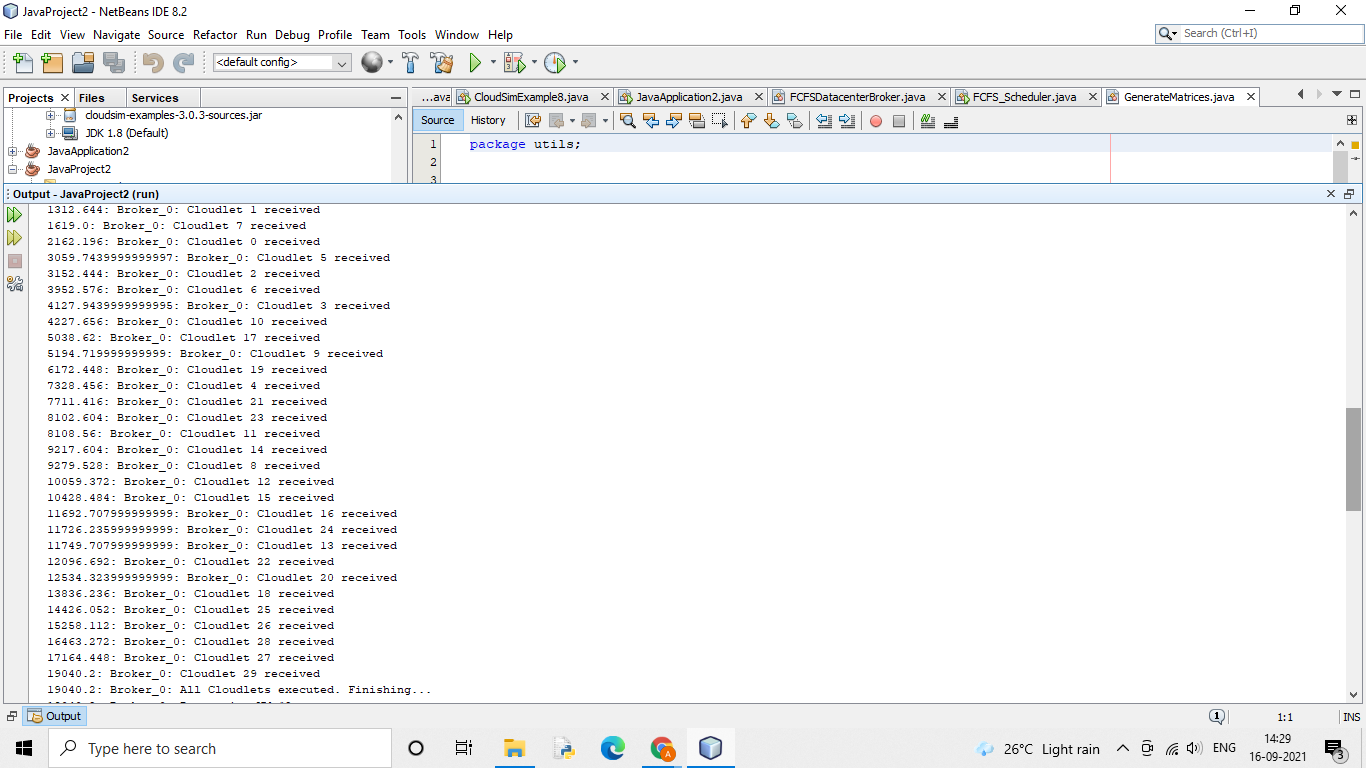
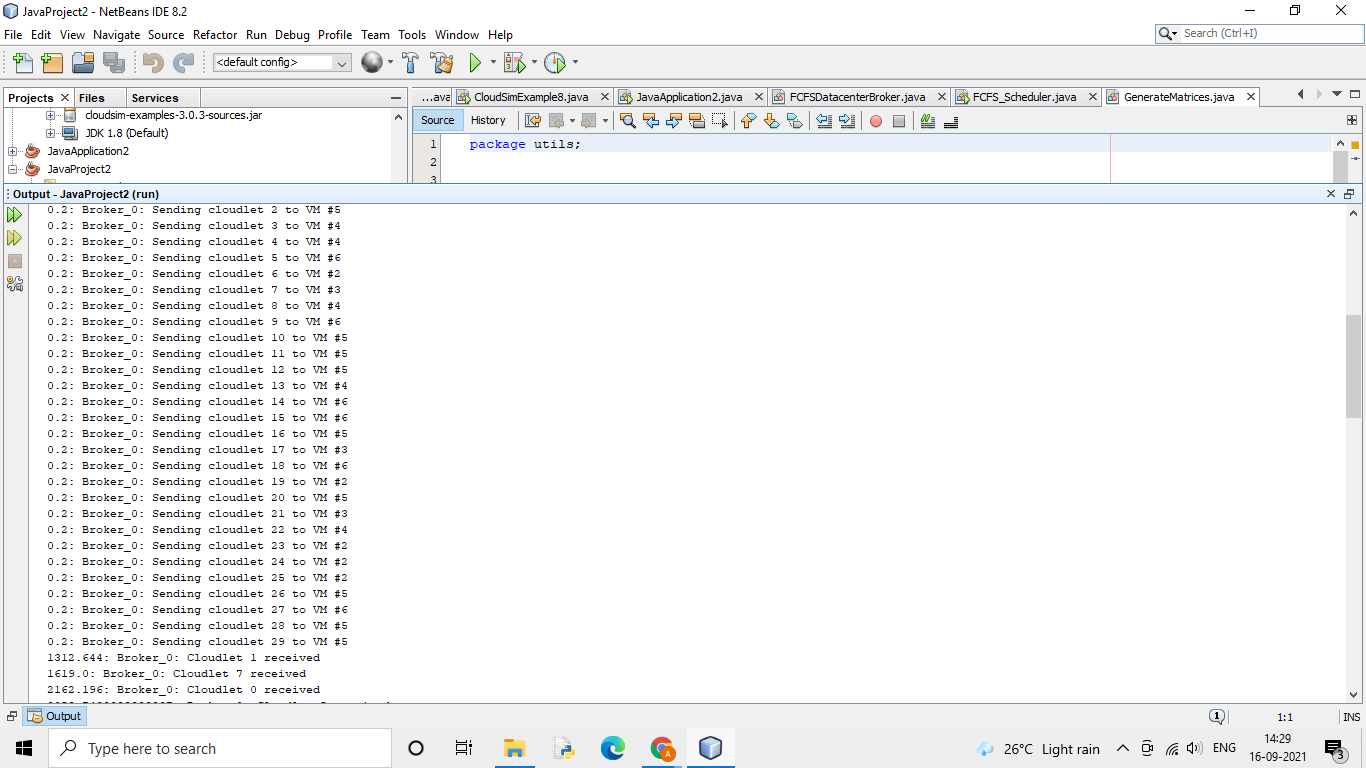
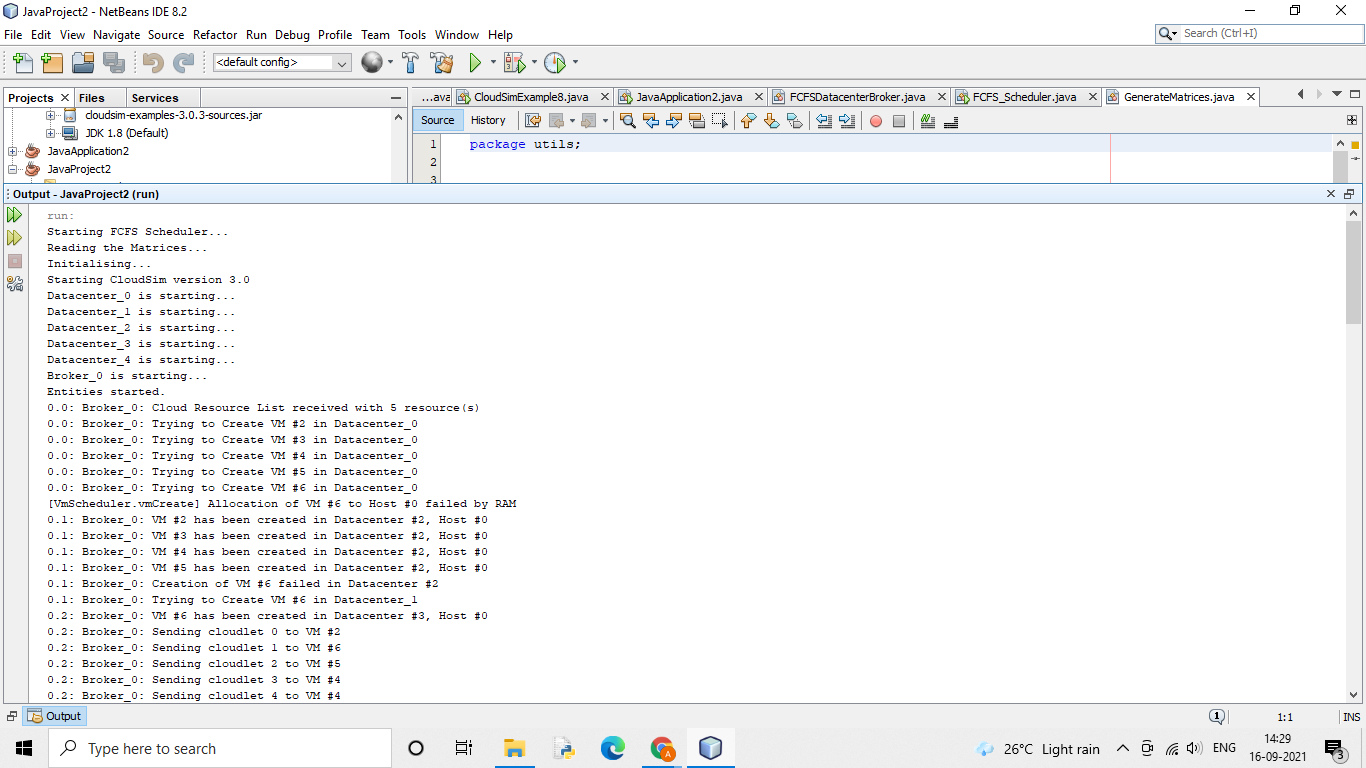
* For implementing Cloudsim Project, we need to set up a project with a Cloudsim .jar on our IDE (Netbeans).
* We are going to implement First Come First Serve (FCFS), Shortest Job First (SJF), Round Robin (RR) Scheduling Algorithms.
* In Cloudsim we need a Scheduler Class and a Specific Broker class for Scheduling Algorithms.
* Scheduler Class is an inbuilt Class in CloudSim library.
* For Scheduler Class -
  + Initialise Lists for VMs, Data Centers, Tasks etc
  + Specify VM parameters
  + Provide Task Matrix containing Task details & Required Time
  + Create Data centers & Initialise them.
  + Create Brokers, specific for each scheduling algorithm.
  + Create & Initialize VM and Cloudlets
  + Provides these to Broker
  + Start Simulation
  + Get the results
  + End the simulation
* Cloud Broker is an entity that manages the use, performance and delivery of cloud services, and negotiates relationships between cloud providers and cloud consumers. The broker for each of the algorithms has been implemented in a different way, according to the algorithm
  + FCFS (First Come First Serve ) Policy
    - Cloudlets are scheduled, as they are supplied to the broker.
    - And execution is performed
  + SJF (Shortest Path First) Policy
    - Cloudlets are simply sorted on the basis of their Job Length
    - And execution is performed
  + Round Robin Policy
    - Each cloudlet is assigned a pre-defined time to execute in a cyclic way.

## 

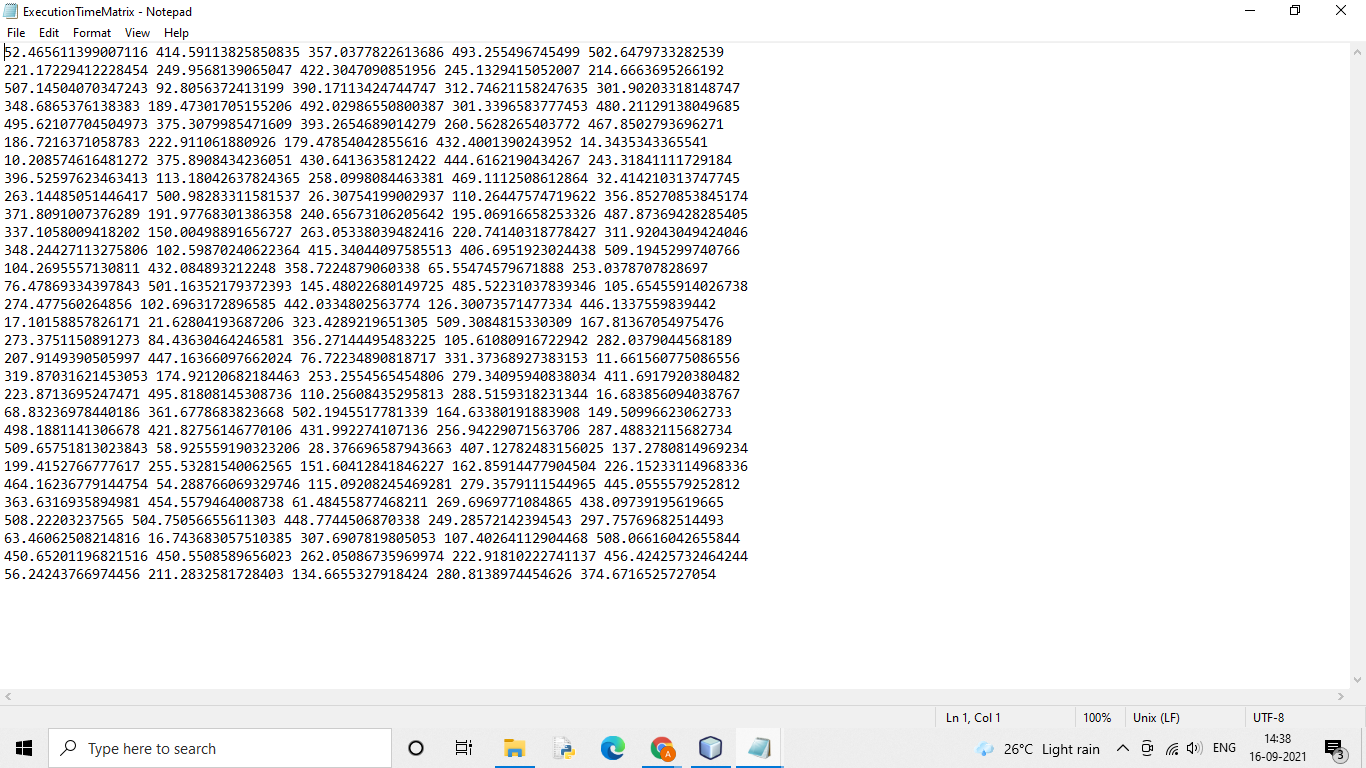
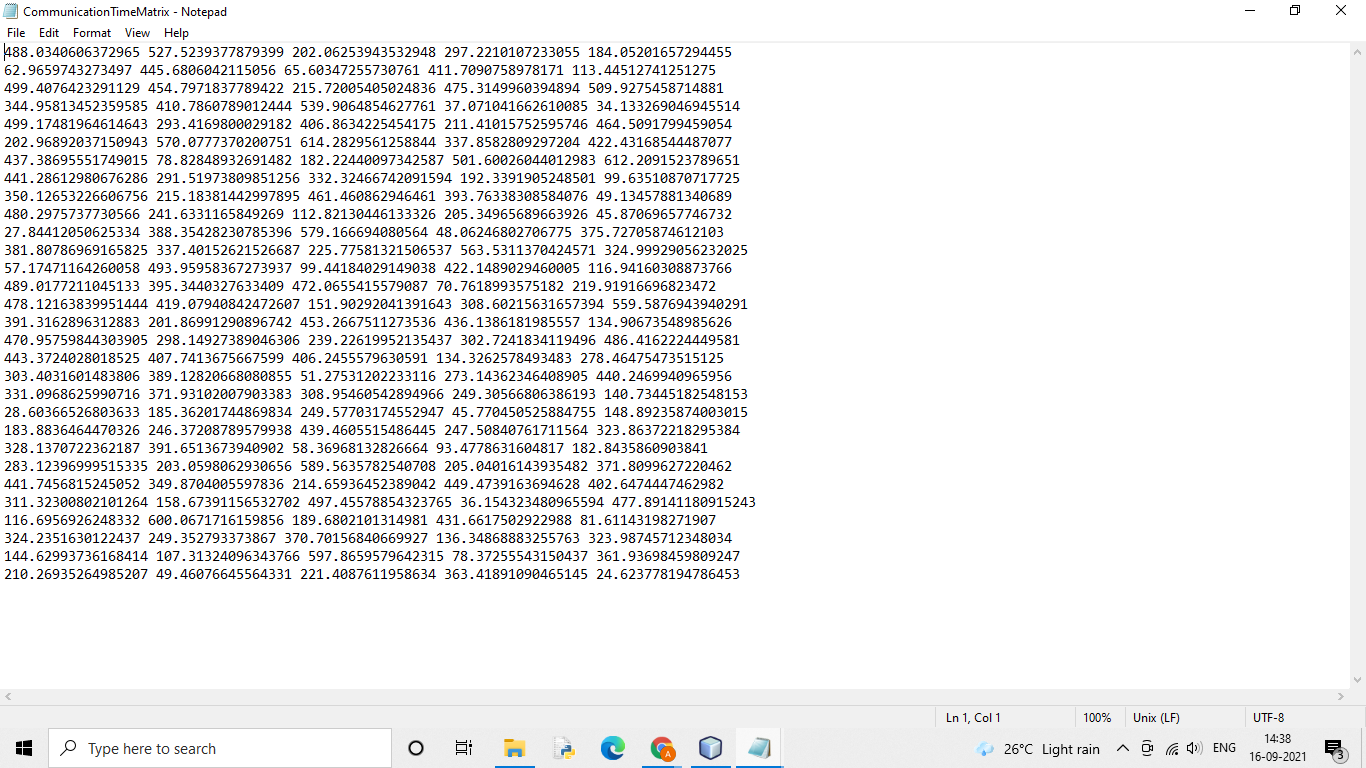
**OUTPUT**

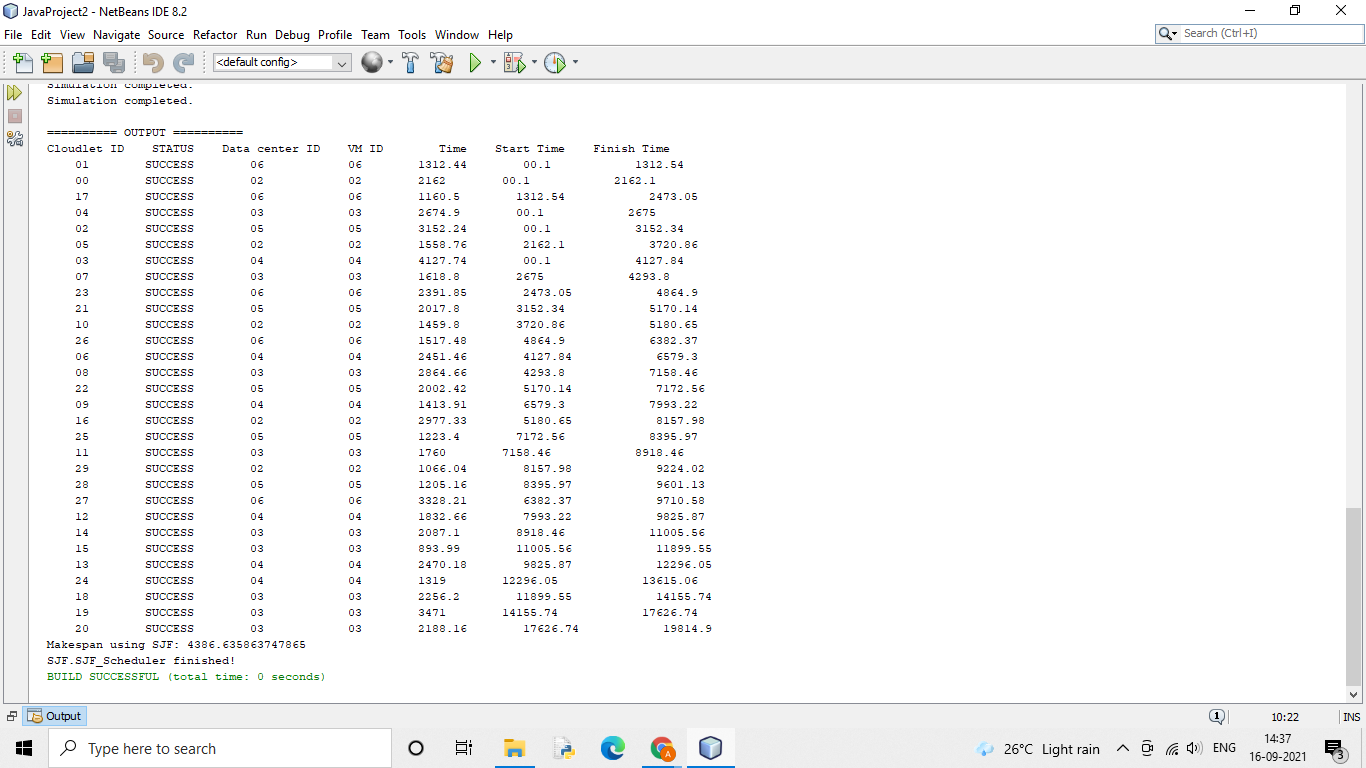
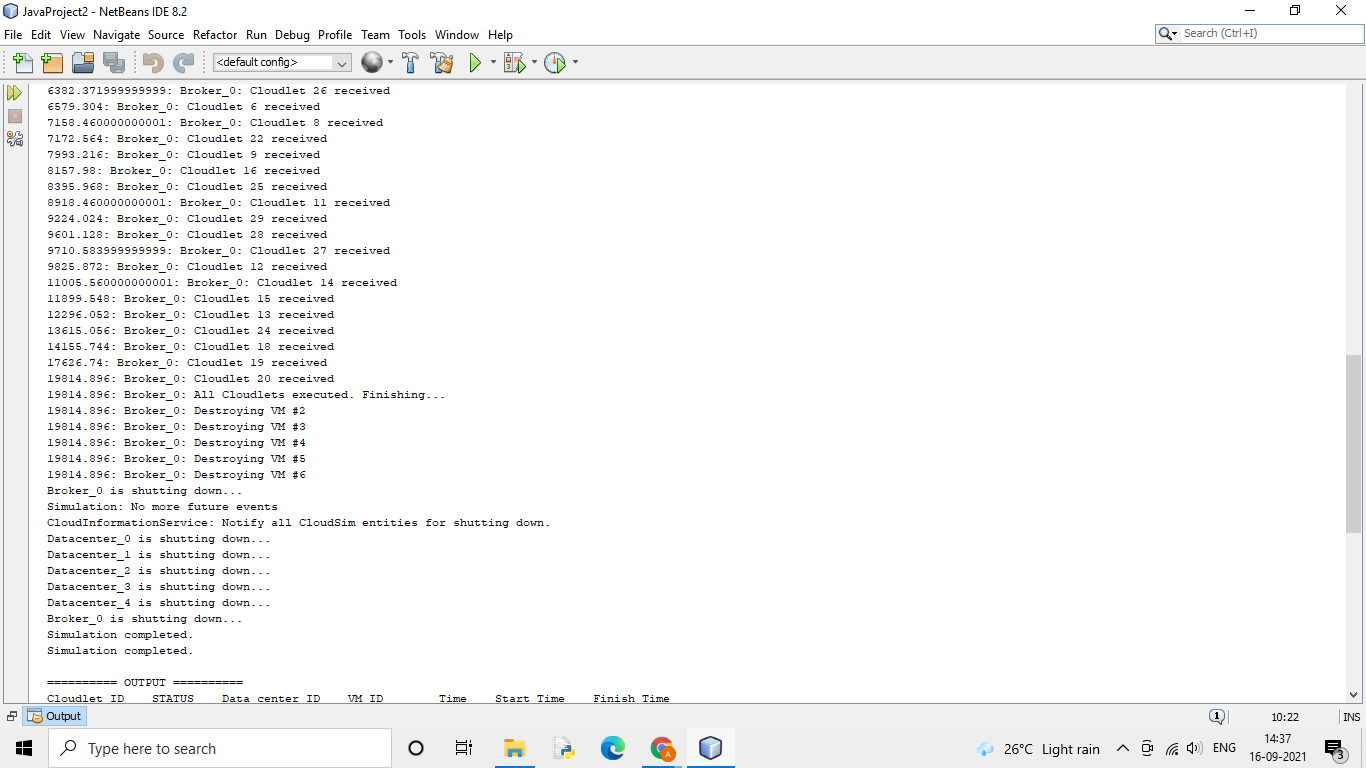
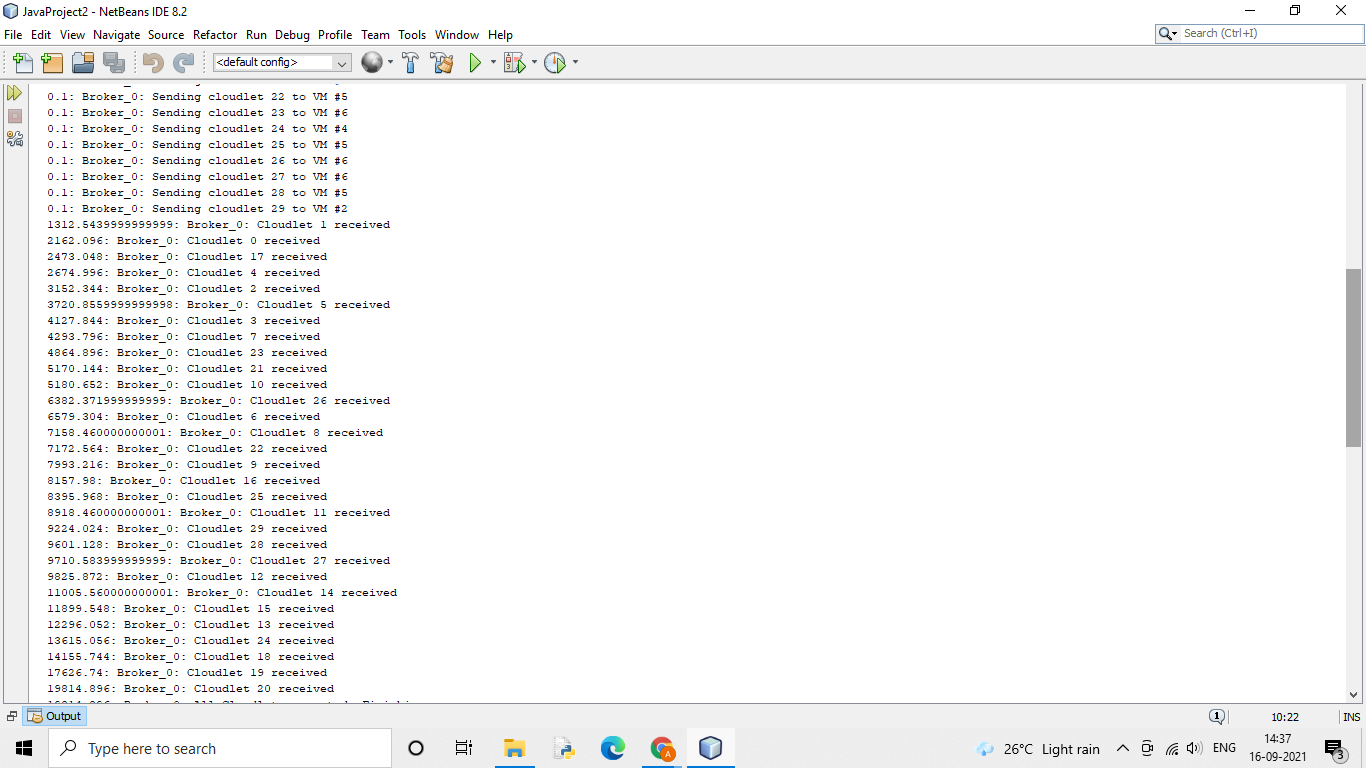
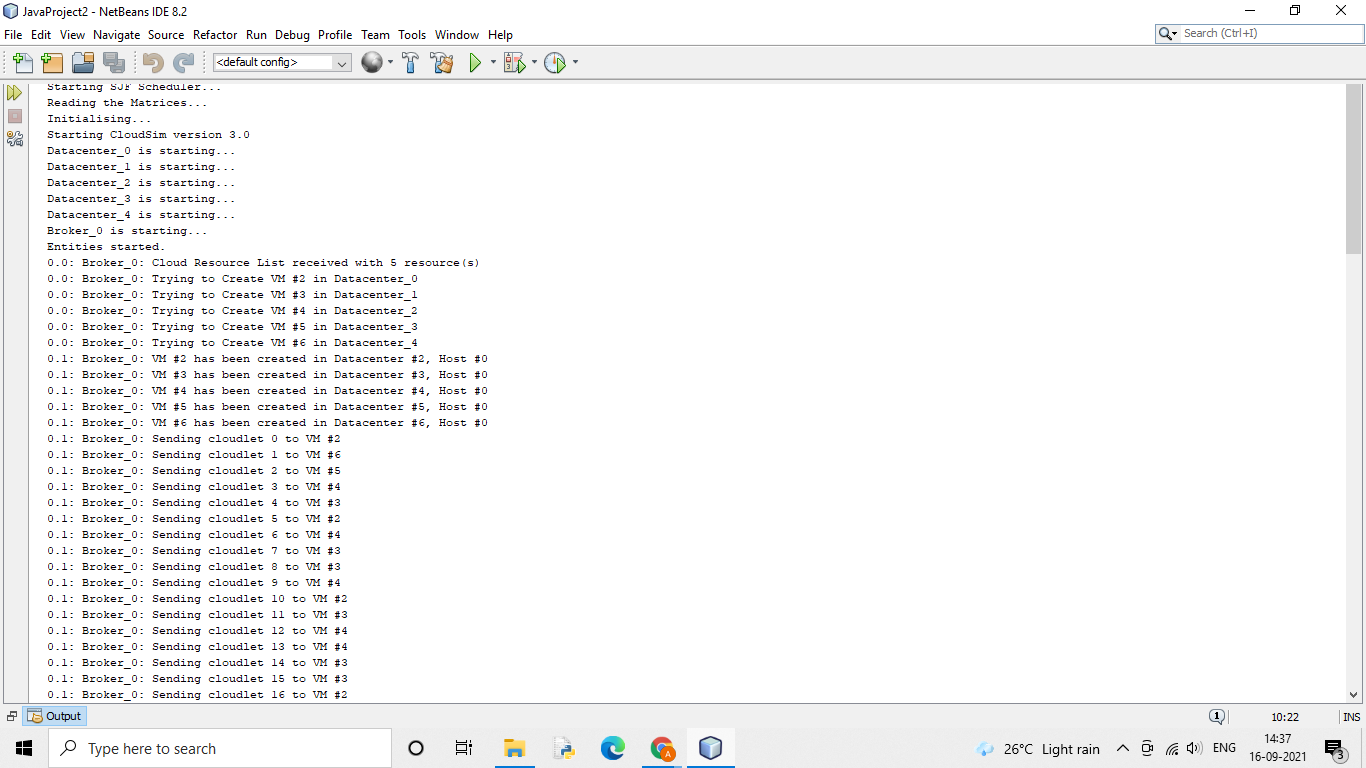
**FCFS**

****

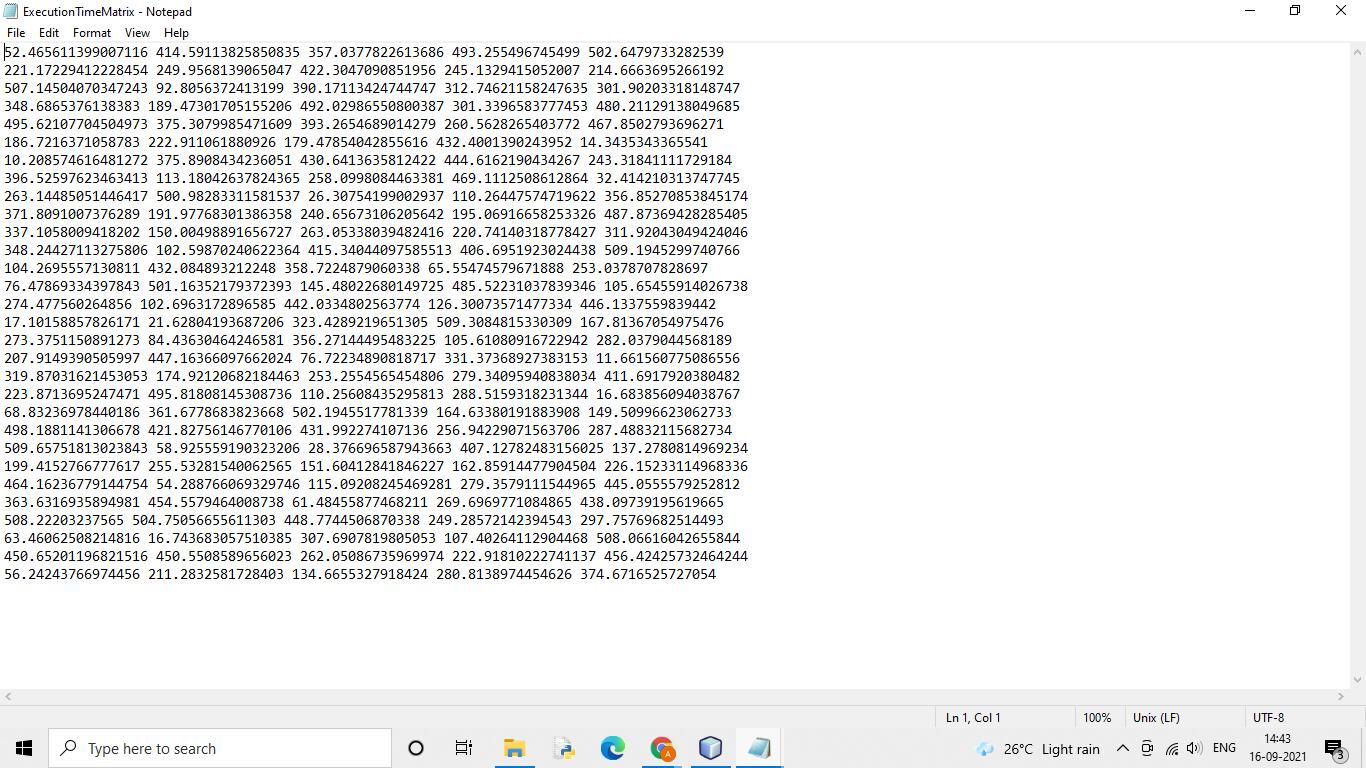
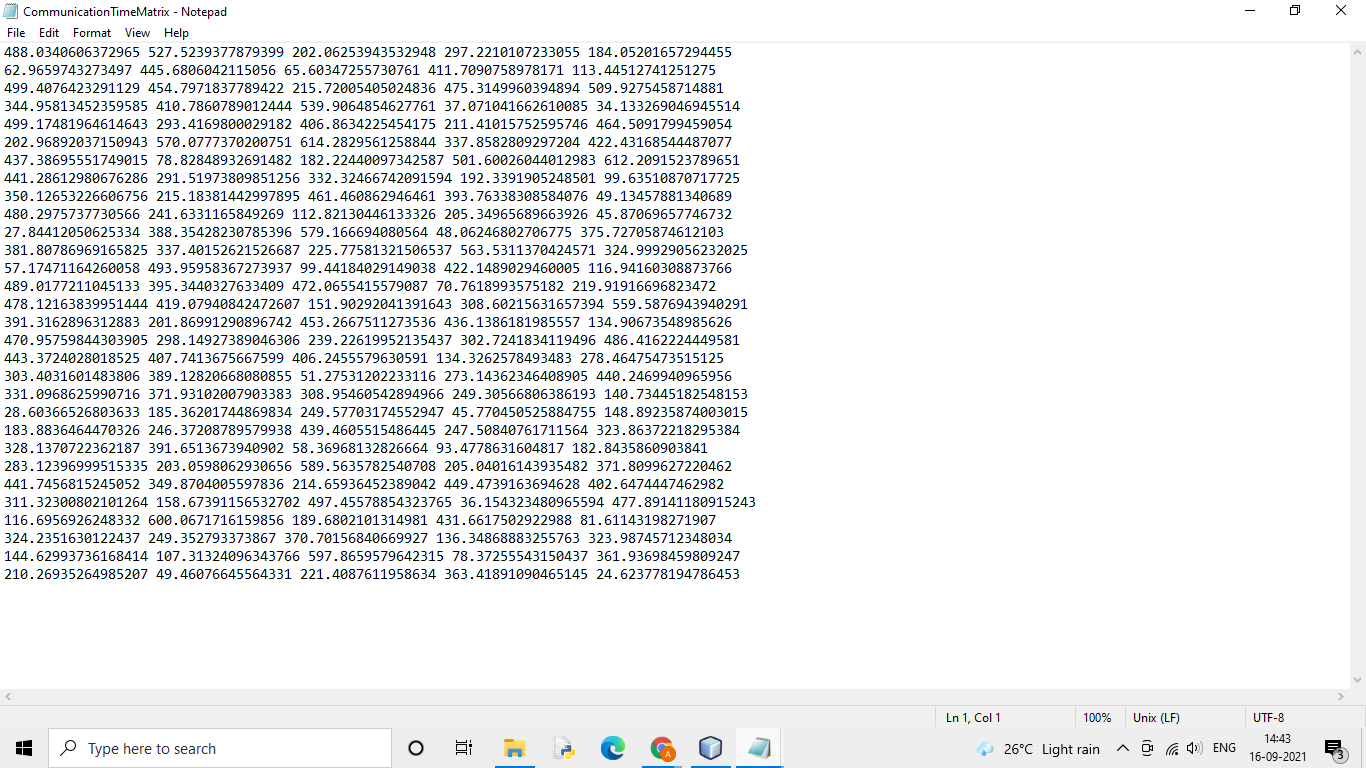
****

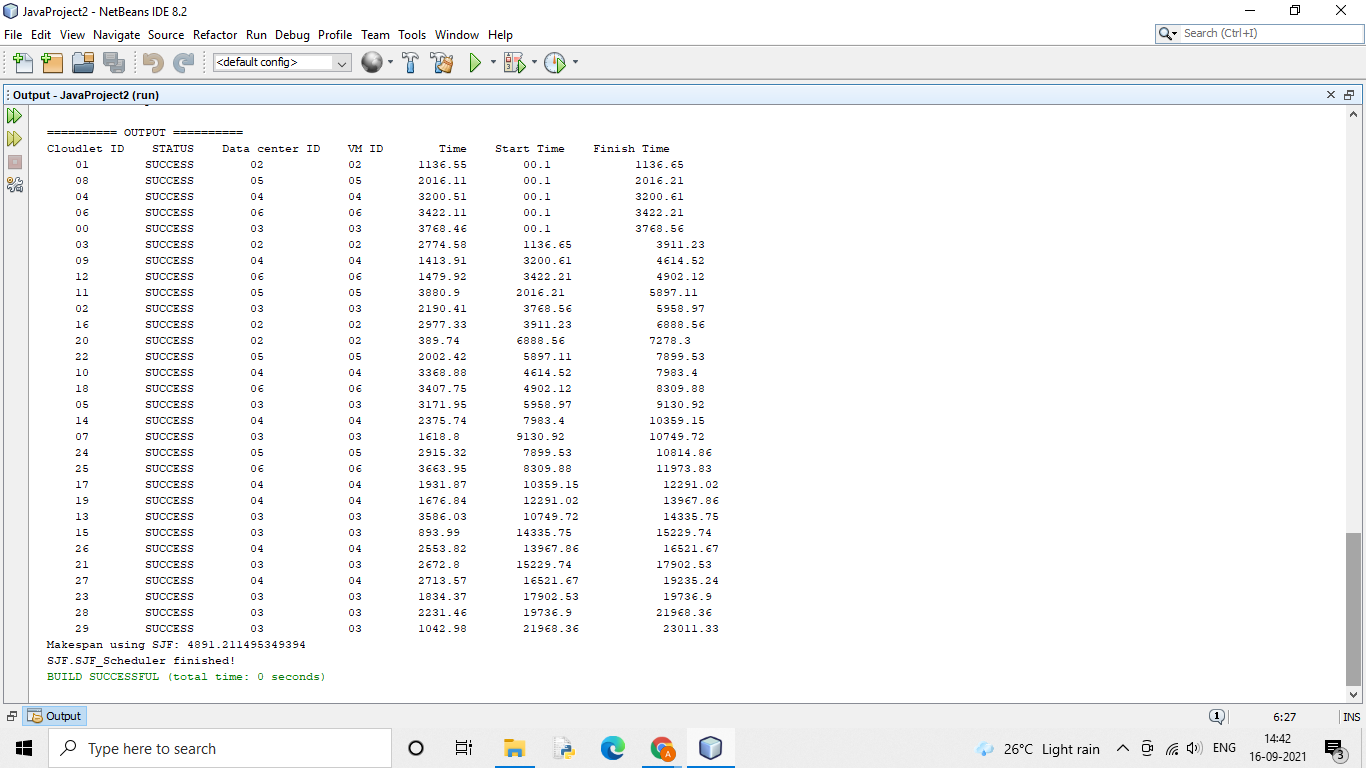
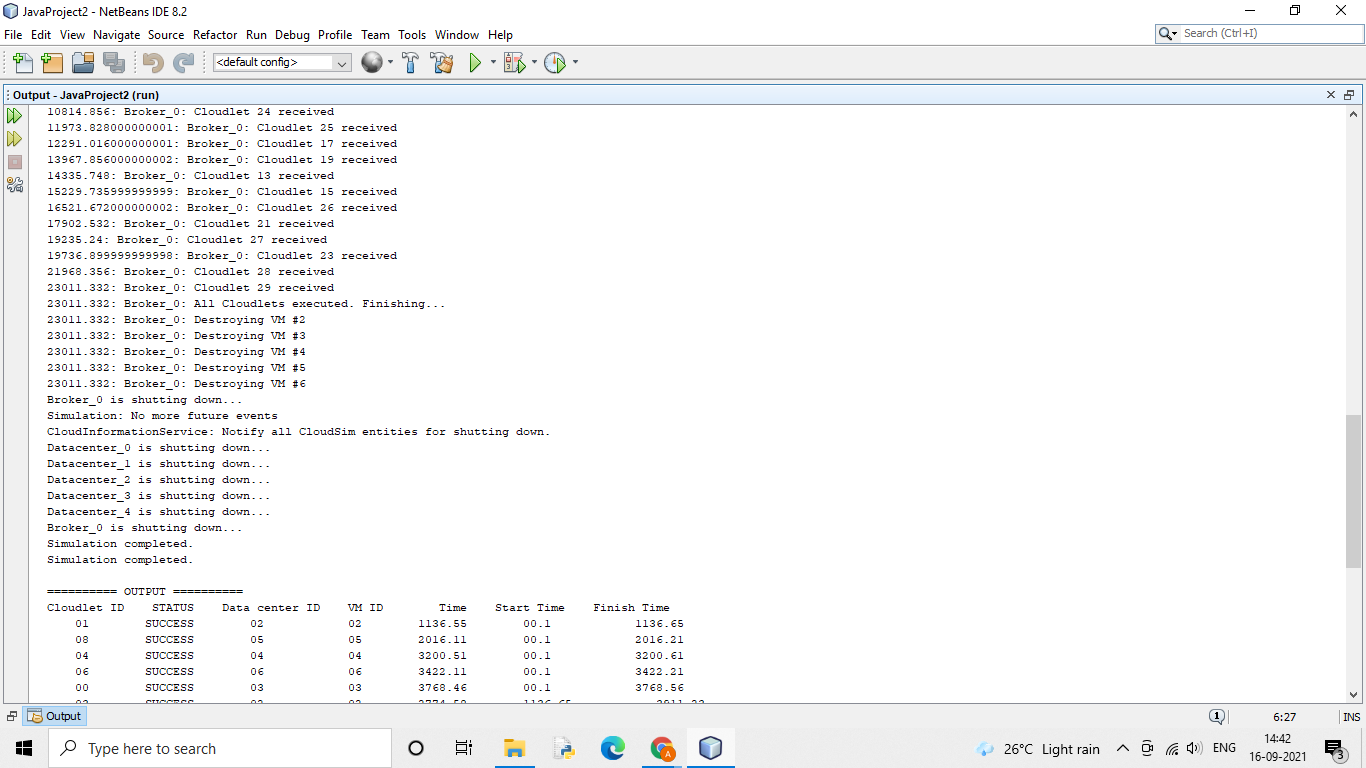
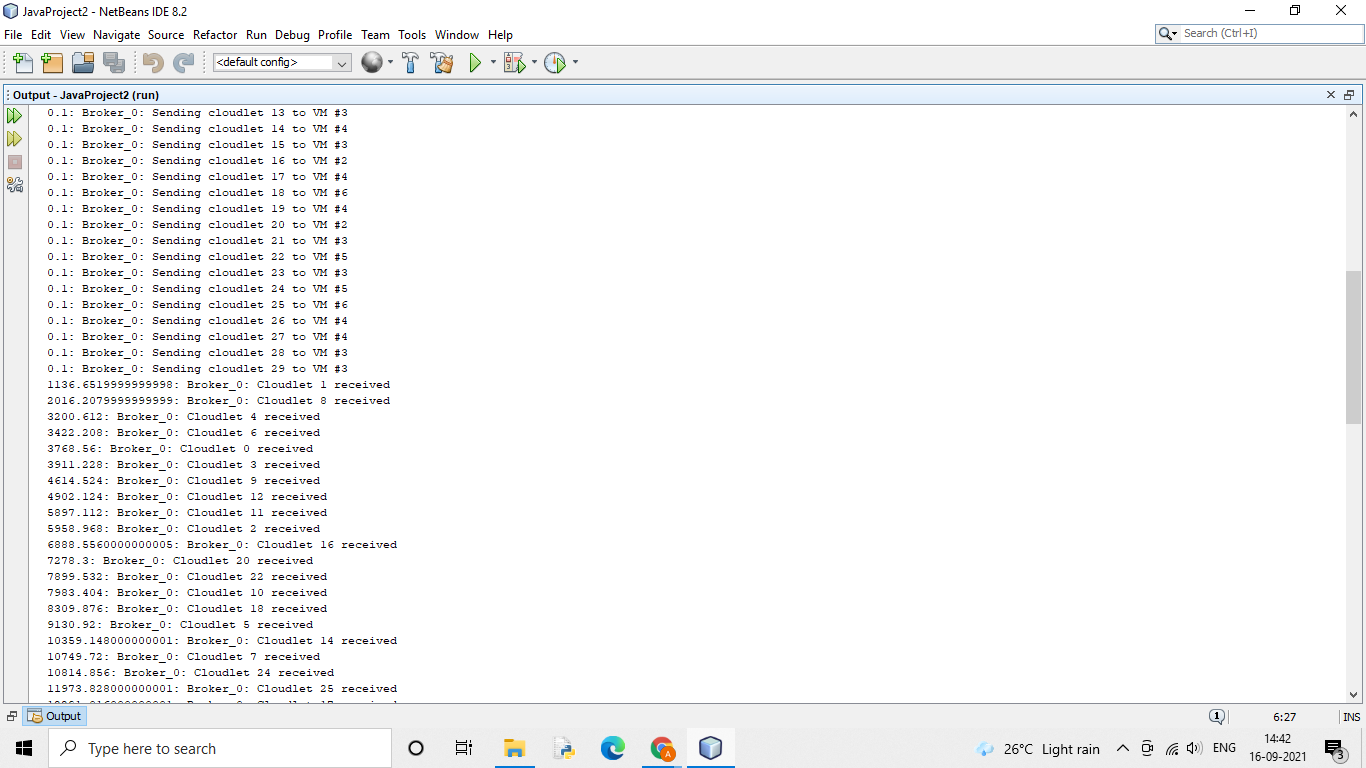
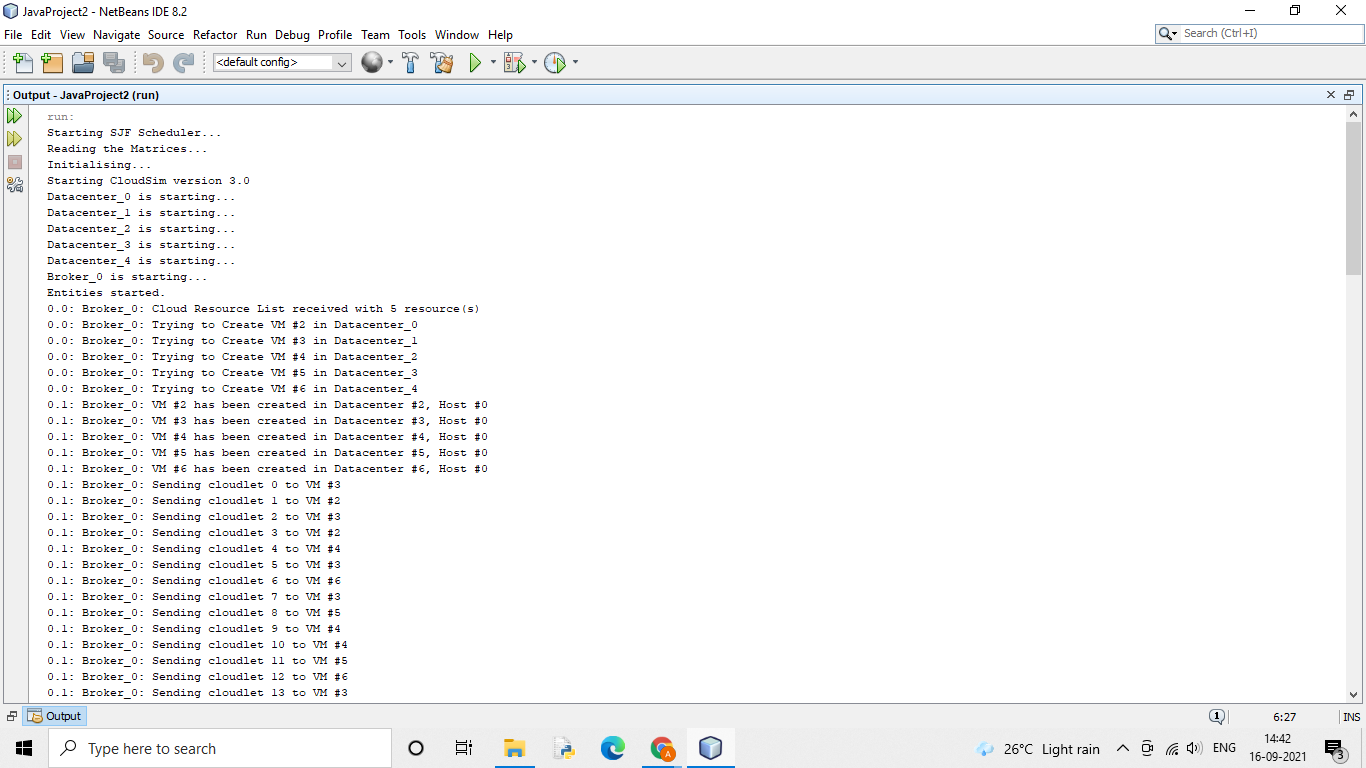
**SJF**

****

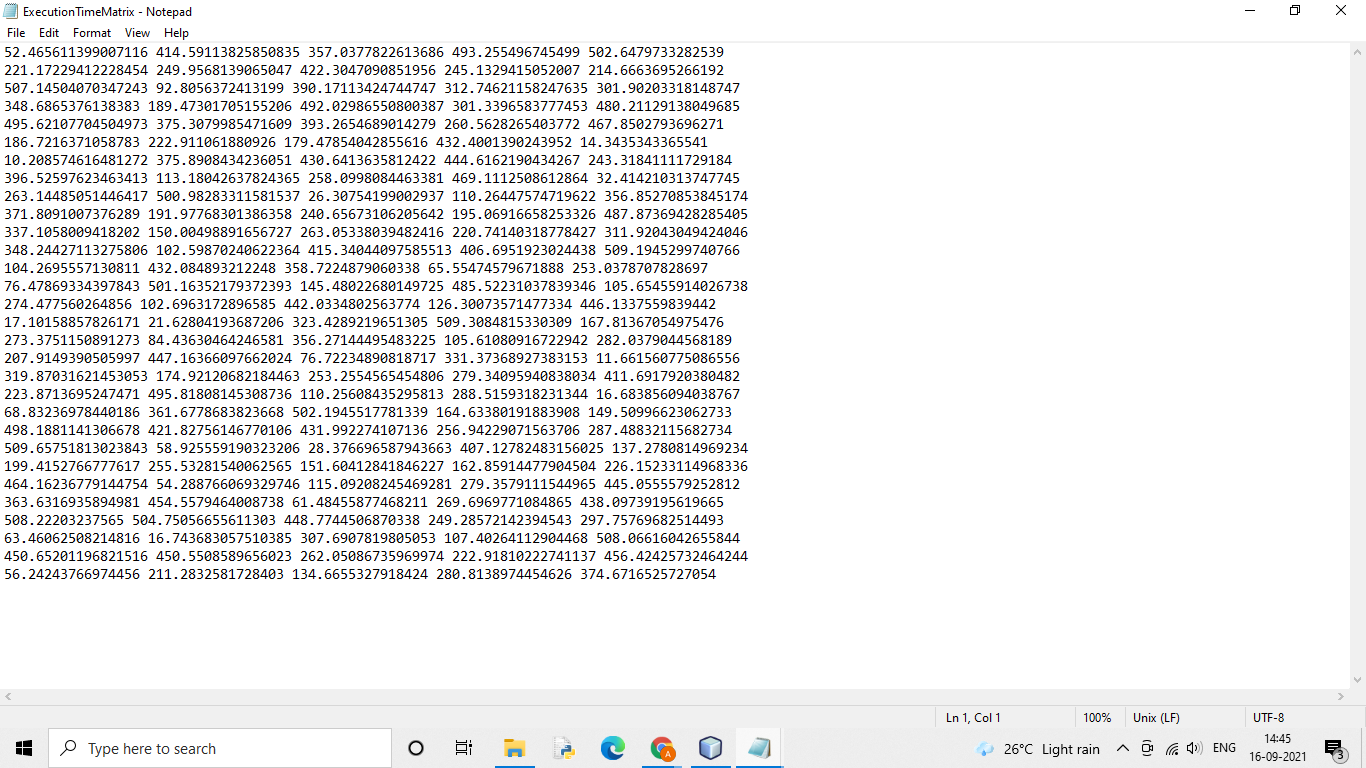
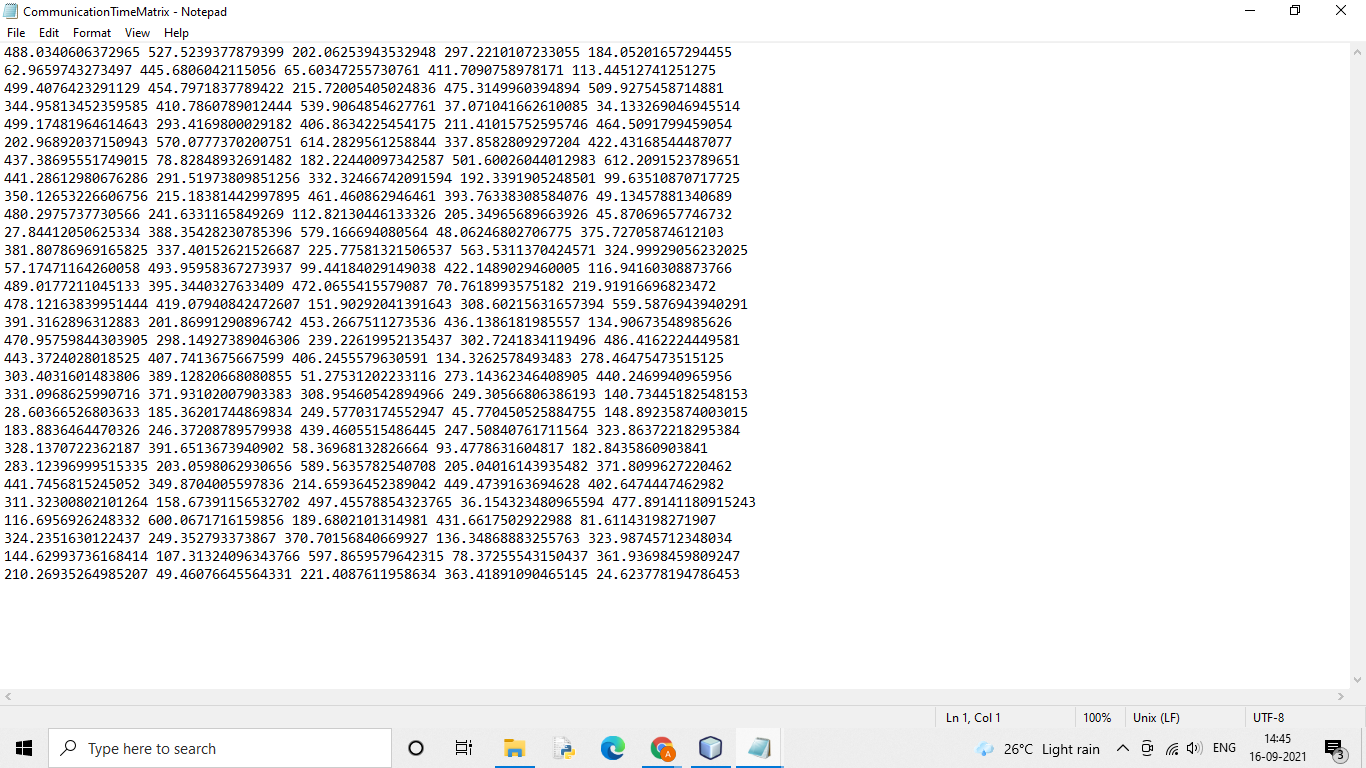
****

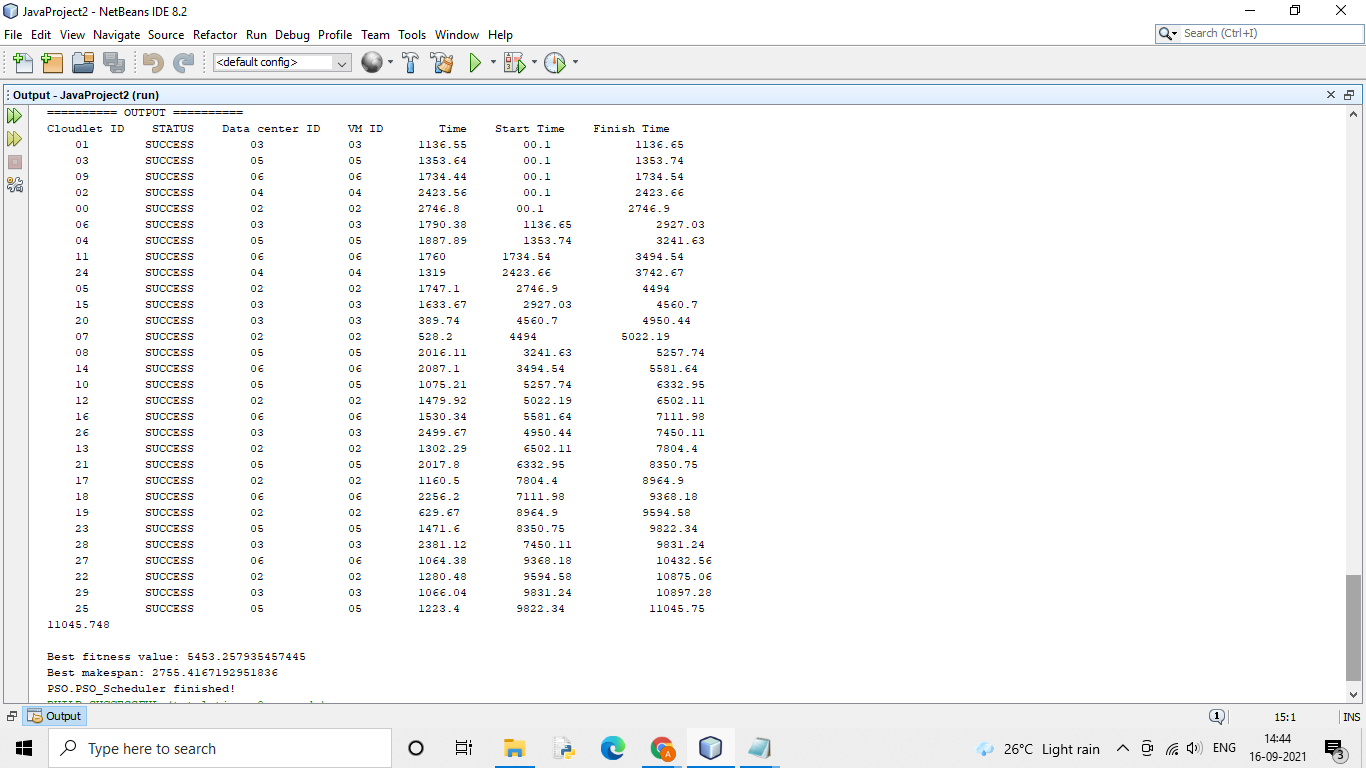
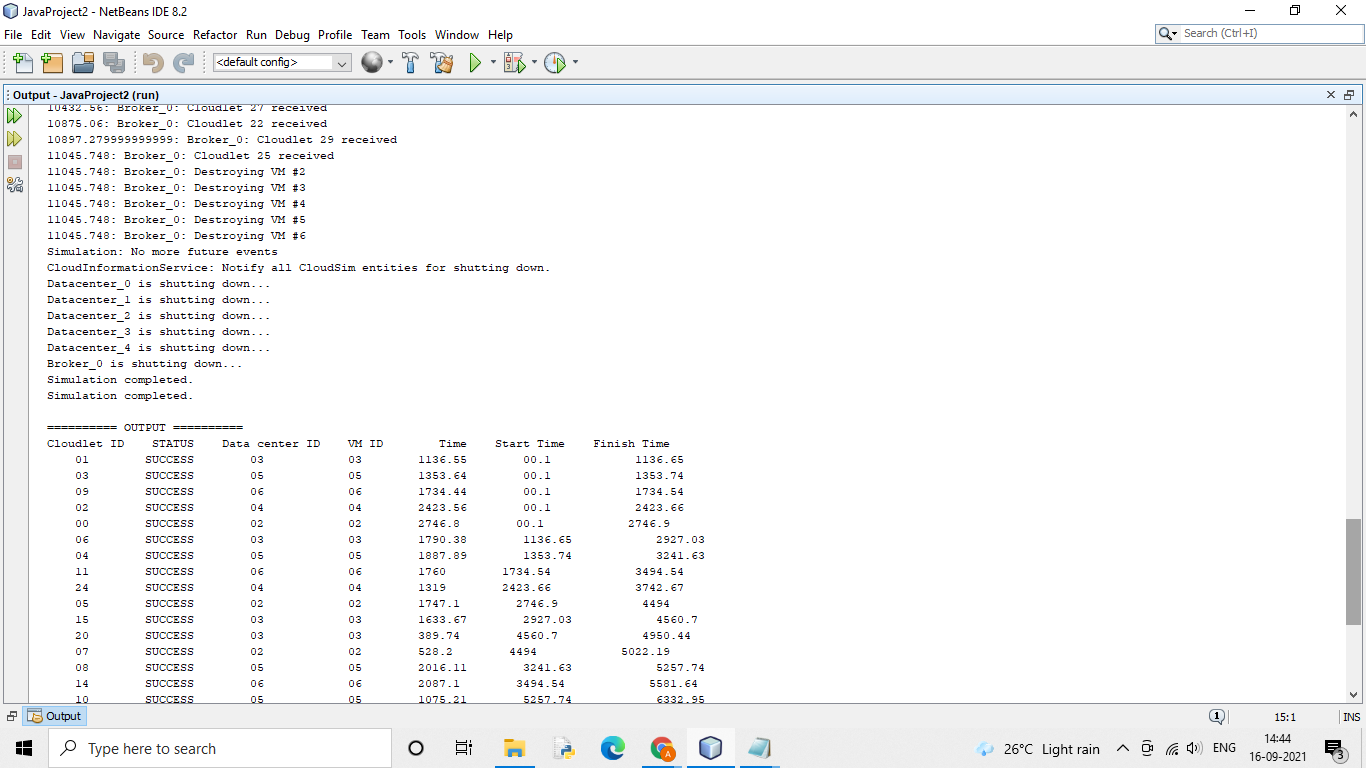
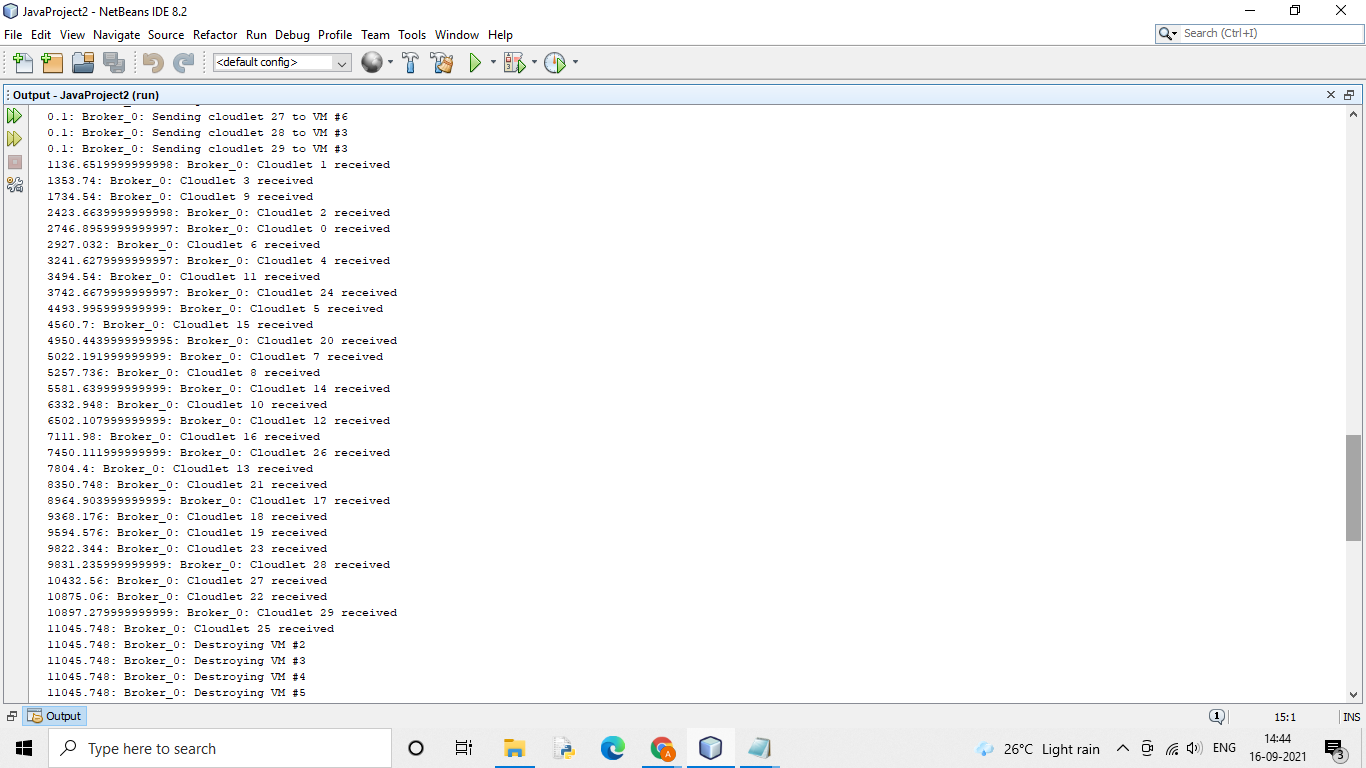
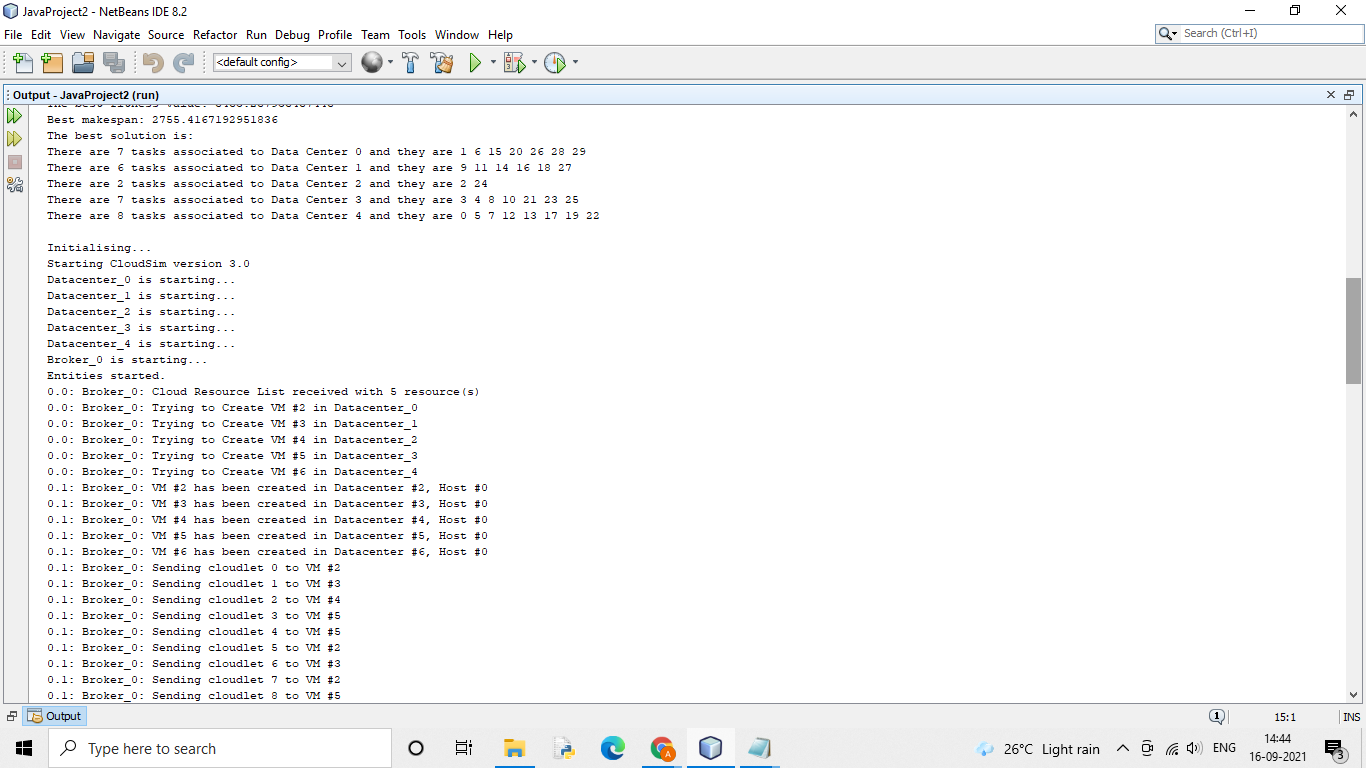
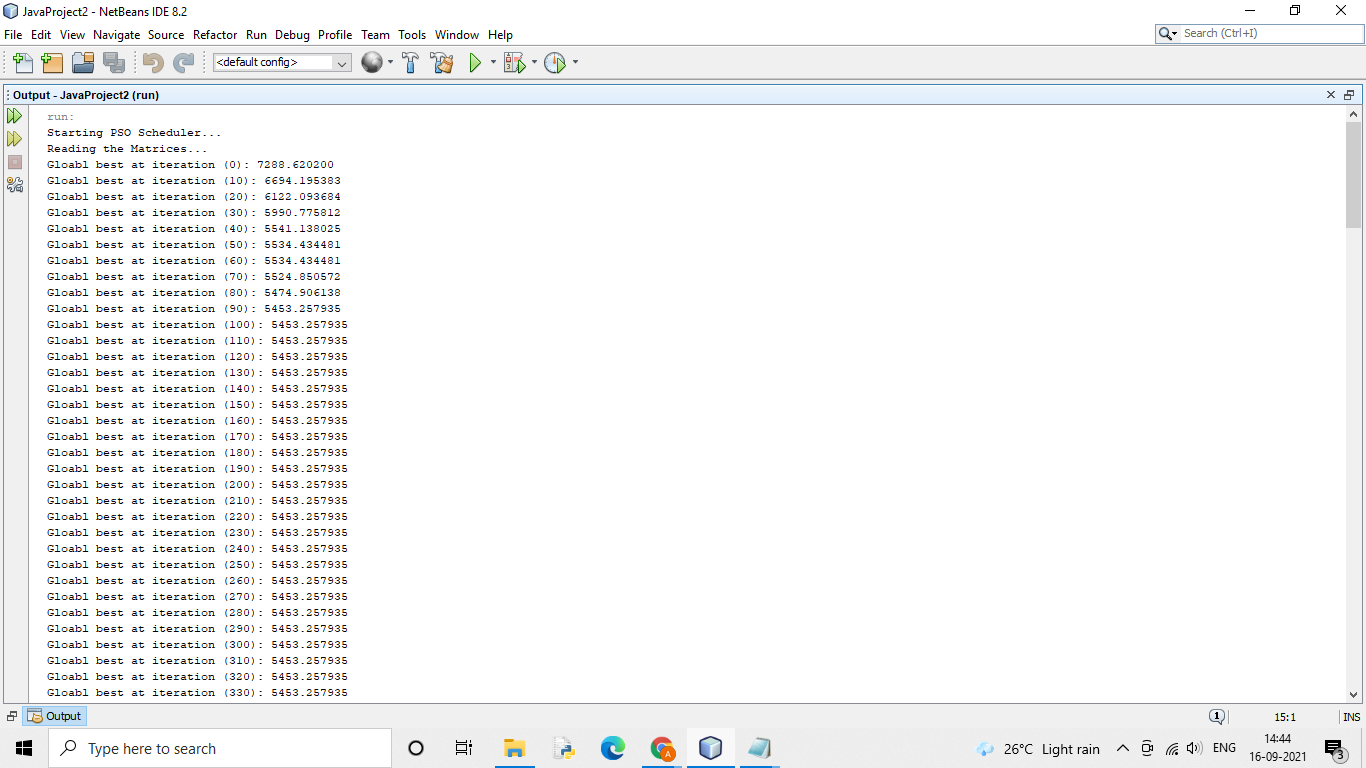
**Round Robin**

****

****

**(Extra)PSO**

****

****