**CE412 Project 3 Report**

**NAME: Mert Yağmur GRADE:**

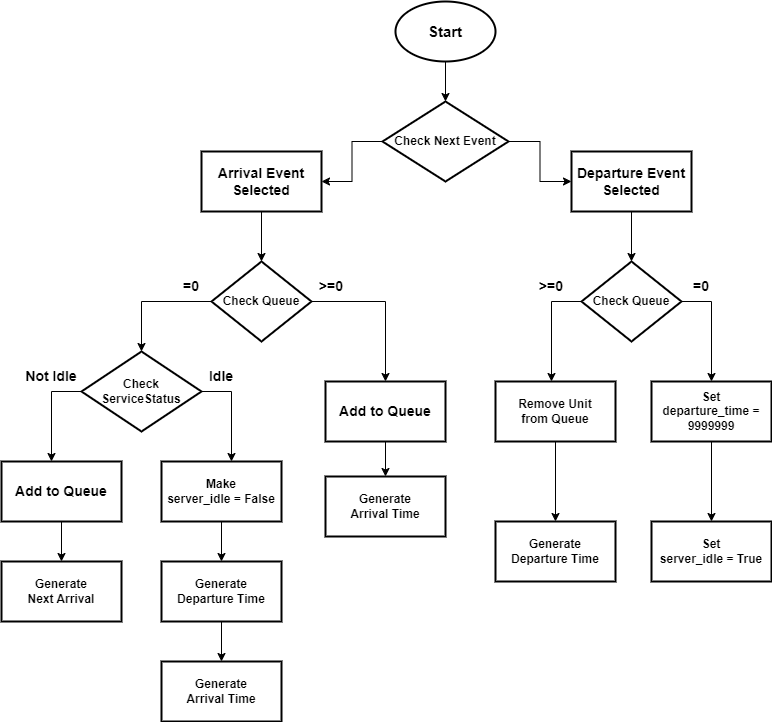
**NUMBER: 20191701012**

“By submitting this project report, I declare that I didn’t get any help from any kind of resources, I didn’t give help to anyone, and all the material submitted completely belongs to me. “

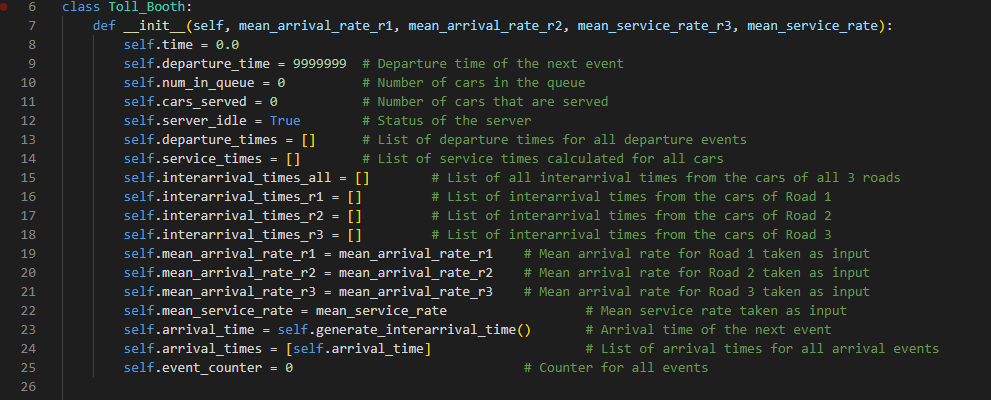
**Method:**

In this project, I made some research first and learned that there are 2 approaches that I could use for simulating such a system: event-based and process-based. There is a useful library called SimPy that can be used for process-based simulations on Python, which is my preferred programming language for this project. I chose to not use it as it abstracts away a lot of logic and preferred to go with the event-based approach.

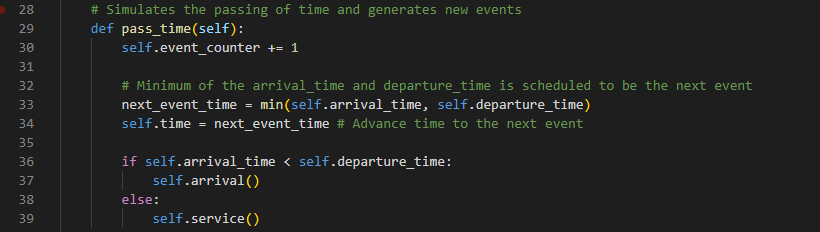
**Flow Chart:**

****

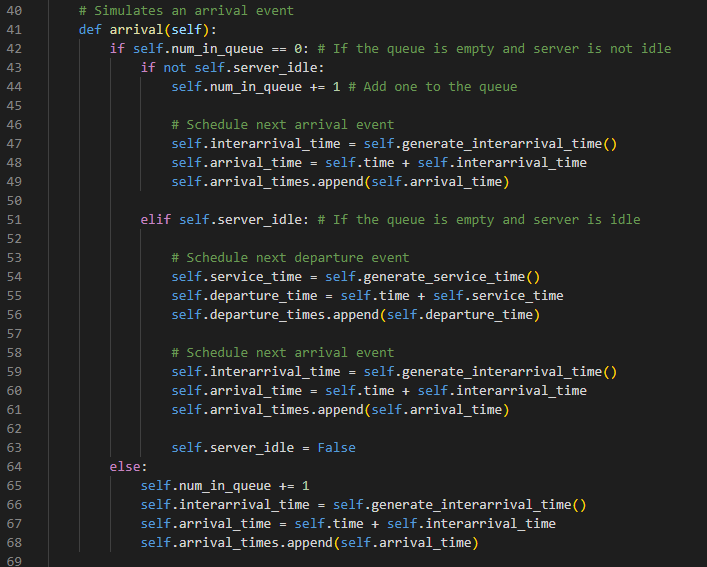
**Code:**

****

In order to make it easier to track different variables, the simulation I designed works with a simulation class called “Toll\_Booth”. The \_\_init\_\_ method takes arrival and service rates as parameters and creates a simulation object.



The pass\_time function belongs to the simulation class and simulates the passing of time and generates new events. Event counter is updated each time this function is called. The next event is calculated from the minimum of arrival and departure times and the time is advanced to the next event.



Arrival function simulates the arrival events. There are possibilities when an arrival event occurs:

**1-)** **Queue is empty and server is not idle:**

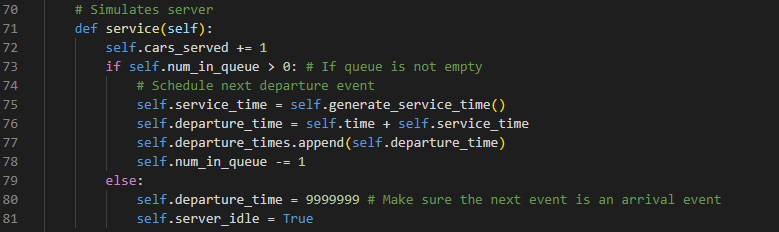
The current unit is added to the queue and the next arrival event is scheduled

**2-)** **Queue is empty and server is idle:**

Departure for the current unit and arrival for the next unit is scheduled, server is made not idle.

**3-)** **Queue is not empty:**

The current unit is added to the queue and arrival time for the next unit is scheduled



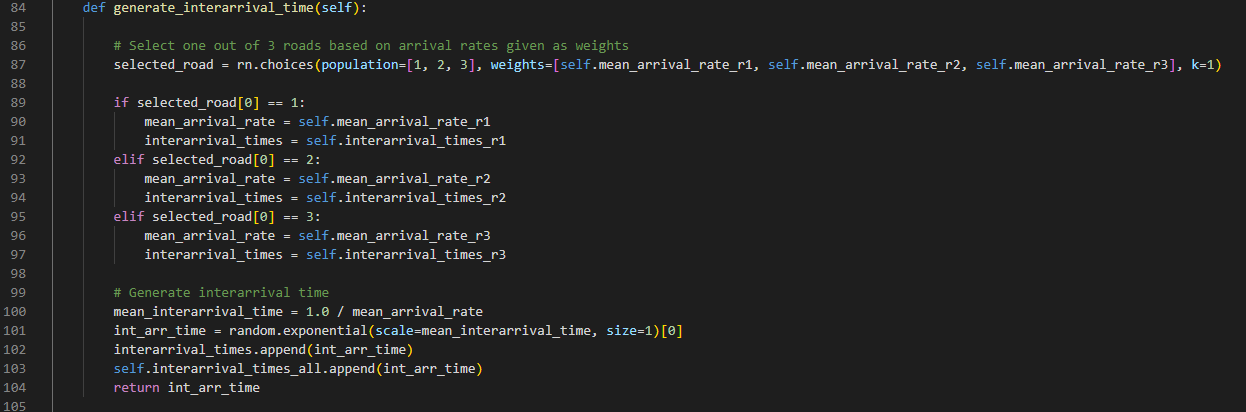
Service function simulates the server of the system. It updates the cars\_served variable each time it’s called. Service function is invoked when the time for a departure event comes. In that case there are 2 possibilities:

**1-) Queue is empty:**

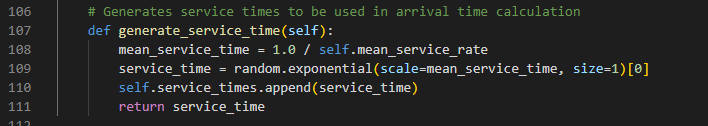
Departure time is set to be a very large number to ensure the next event scheduled is an arrival event. Server is made idle

**2-) Queue is not empty:**

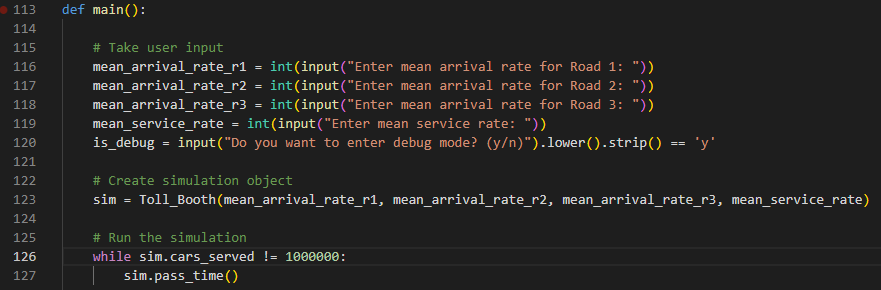
One unit is removed from the queue to be serviced. Departure event for the next unit is scheduled.

****

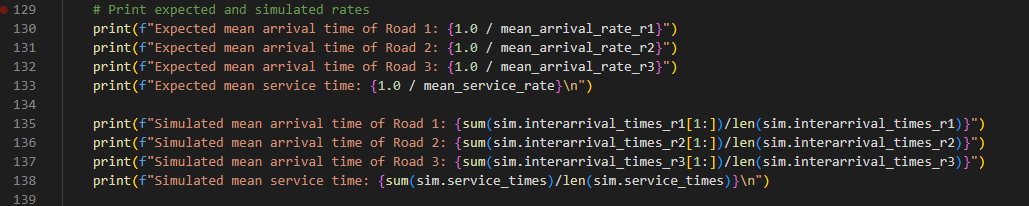
This function generates interarrival times to be used in arrival time calculation. Road is selected randomly with weights of arrival rates and interarrival time for that road is calculated and saved.

****

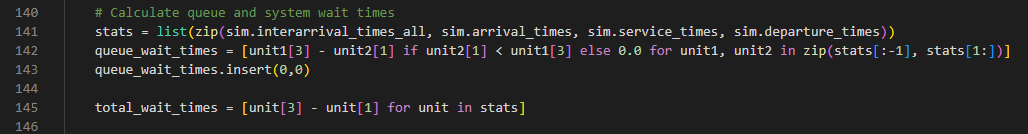
This function generates service times to be used in arrival time calculation.



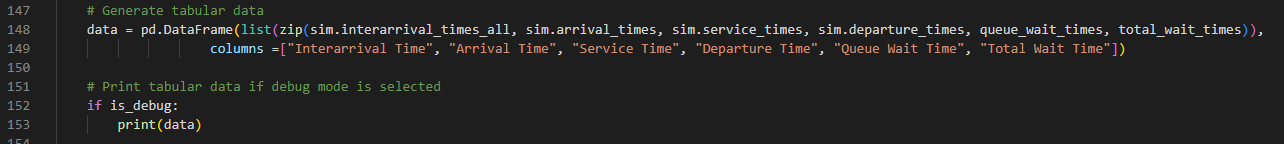
The main function first takes the necessary data from the user and then uses it to create the simulation object. Then the simulation is run until the served cars counter reaches the given limit.



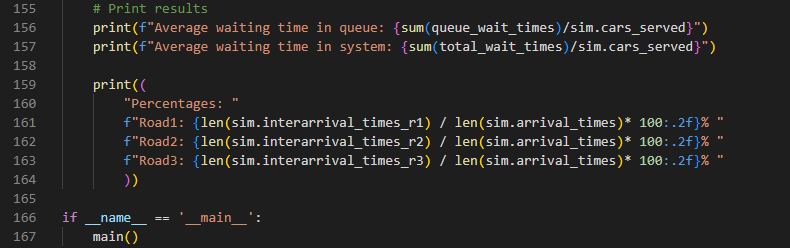
Expected times which are generated from the rates taken from the input is compared to the real results after the simulation is run to ensure there was no error in calculation.



Queue wait times and total wait times are calculated from the available data after the simulation is run.



All the individual stats calculated are shaped into tabular form for better readability. This table is only shown if debug mode is selected.



Other stats are displayed to the user.

**Sample Outputs:**

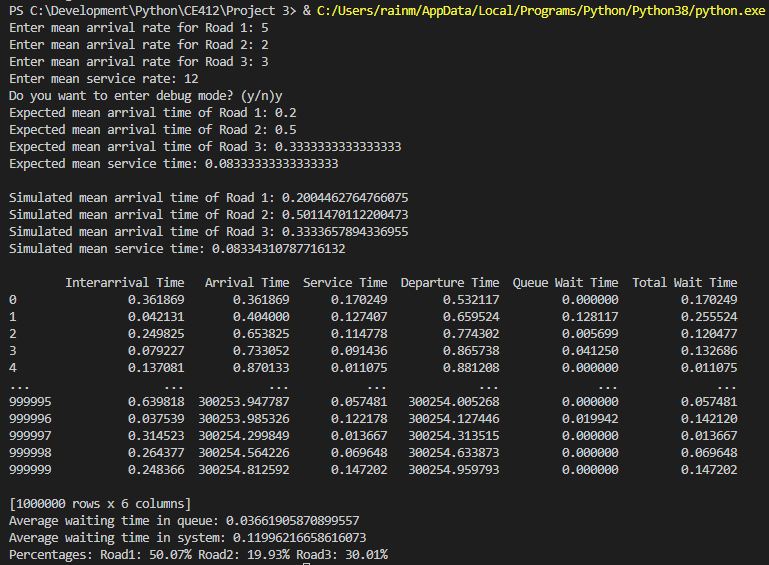
Arrival Rates (r1, r1, r1): 5, 2, 3

Service Rate: 12

Debug Mode: True

Size: 1.000.000 served cars

Time it took to run the simulation: ~30 seconds



**Comments:**

I observed that the simulated times converge to the expected times which tells me that the calculations in the simulations are correct. Furthermore, the percentages of the cars in different roads also satisfy the given arrival rates. The tabular data shown when debug mode is selected also shows the interarrival times, arrival times, service times, departure times, queue wait times and total wait times do not contradict each other. All in all, the simulation seems to working fine with correct results.