IE3081 – Modeling and Discrete Simulation Course

Homework-5

	Name Surname	School ID
1	Berkkan RENÇBER	150119011
2	Murat ÖZCAN	150119008
3	Samet CAN	150120528

CONTENT

1) SYSTEM COMPONENTS

a) AGENTS

- **Person:** It is the basic agent type in the system and represents the people who join the system.
- **Student:** It represents students in the system. It is inherited from the Person agent.
- **Personnel:** It represents the personnel in the system. It is inherited from the Person agent.

b) RESOURCES

- **StudentSource:** This resource enables the creation of a new student.
- **PersonnelSource:** This resource enables the creation of a new personnel.

c) QUEUES

- **OutsideQueue:** This queue represents the protruding portion of the queues inside the refectory.
- **InsideQueue:** It is the first queue inside the refectory. Represents the order in which students receive food. It takes precedence over the second query.
- **InsideQueue2:** It is the second queue inside the refectory. Represents the order in which students receive food.
- **PersonnelQueue:** This queue represents the personnel's food queue, and if the student queue gets too full, a few of the students will join this queue.

d) DELAYS

- **Staff1:** This delay represents staff members who fill dishes for students in the InsideQueue. It fills dishes at random times with the StaffSpeed table function.
- **Staff2:** This delay represents staff members who fill dishes for students in the InsideQueue2. It fills dishes at random times with the StaffSpeed table function.
- **Staff3:** This delay represents staff members who fill dishes for students or personnels in the PersonelQueue. It fills dishes at random times with the StaffSpeed table function.

e) SERVICES

- **Tables:** It represents the tables in the refectory and has a capacity of 160 people and gets it from the TablePool.

f) SINKS

- **OutOfDestroyer:** With this sink, students who abandon the system are destroyed when the OutsideQueue is too long.
- **Destroyer:** With this sink, students and personnel who eat and leave the system are destroyed.

g) SELECT OUTPUTS

- **OutDesicion:** With this select output, students decide to exit the system if the queue outside is long.
- **PersonnelQueueDesicion:** With this select output, students decide to switch to the PersonnelQueue if their reserved queue is full and the personnel queue is empty enough.
- **StudentQueueDesicion:** With this select output, students decide to switch to the least queue of the queues allocated to them.

h) TABLE FUNCTIONS

- **StaffSpeed:** This function generates random values to indicate the working speed of the staff.
- **EatsTime:** This function generates random values to indicate the eating speed of students and personnel.

i) RESTRICTED AREA

- **restrictedArea:** This allows food filling to stop as soon as all tables are occupied and prevents unexpected exits from the system.

j) SCHEDULES

- **StudentSchedule:** In addition, we ensure that our school's refectory gives the most realistic results in simulation. For this, we use a schedule that will represent the busiest time of students between 12:20 and 13:00, similar to the real one.
- **PersonnelSchedule:** For the same purpose, it determines the time of the personnel to reach the system.

k) RESOURCE POOL

- **TablePool:** It was created with a capacity of 160 for use in the Tables service.

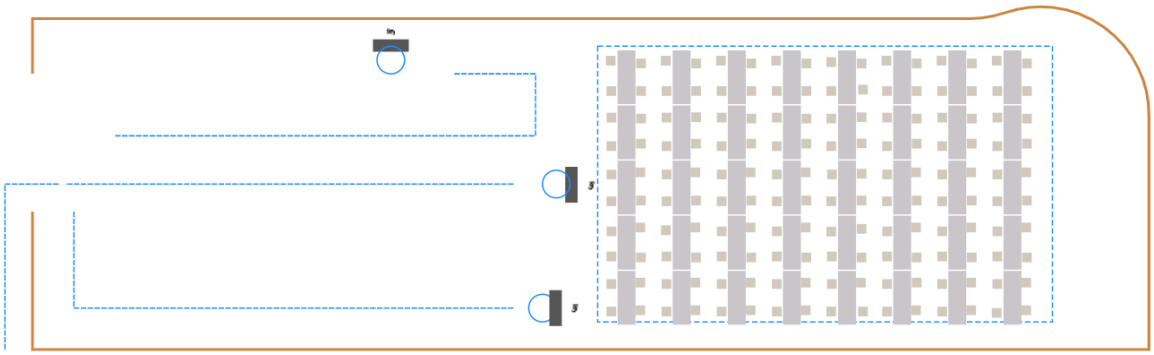
l) TIME MEASUREMENT

- **OutsideQueueStart:** It keeps the time for students to enter the queue outside.
- **OutsideQueueEnd:** It keeps the time for students to exit the queue outside.
- **InsideQueueStart:** Keeps the entry time of students entering InsideQueue.
- **InsideQueueEnd:** Keeps the exit time of students leaving InsideQueue.
- **InsideQueue2Start:** Keeps the entry time of students entering InsideQueue2.
- **InsideQueue2End:** Keeps the exit time of students leaving InsideQueue2.
- **PersonnelQueueStart:** Keeps the entry time of personnels and students entering the PersonnelQueue.
- **PersonnelQueueEnd:** Keeps the exiting time of personnels and students leaving the PersonnelQueue.

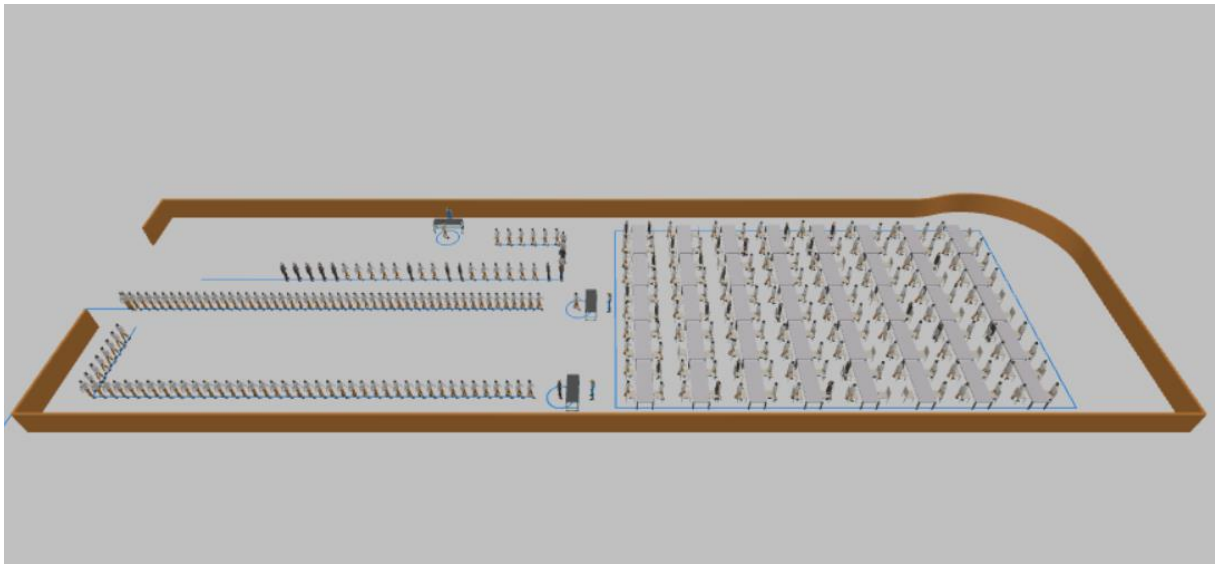
2) RELATIONS

- There are 3 agents. These are person, student and personnel. Student and Personnel are inherited from Person.
- We have 4 tails. 2 of them are put in order for students to get food and there is a prioritization among each other according to the length of the queue. Another queue, Personnel queue, has been created for personnel, but in order to reduce the density in student queues, it is accommodated by the student depending on the conditions. Our last queue represents the queue before splitting into 3 special queues and if this queue is too long some of the students will leave the system.
- Students and personnel in the system use the system as we integrate it from real life. Most of the students use the system between 12:20 and 13:00. The intensity between the other hours is less. The staff are like that too. We have scheduled these times.
- The university cafeteria is only open between 11:00 and 13:00 on weekdays. Therefore, it is closed outside of these hours.
- We have 2 exits. The first is for students who leave the system and the other is for students and personnel who have finished their work in the system.
- With Time measurement, the total time spent by the agents waiting in the queues is calculated.
- In case the tables in the system are full, it is ensured that the personnel wait until at least one of the chairs on the tables is empty so that there is no disruption in the system.

3) VIEW OF 2D MODEL



4) VIEW OF 3D MODEL



5) GENERATION OF RANDOM VARIATES

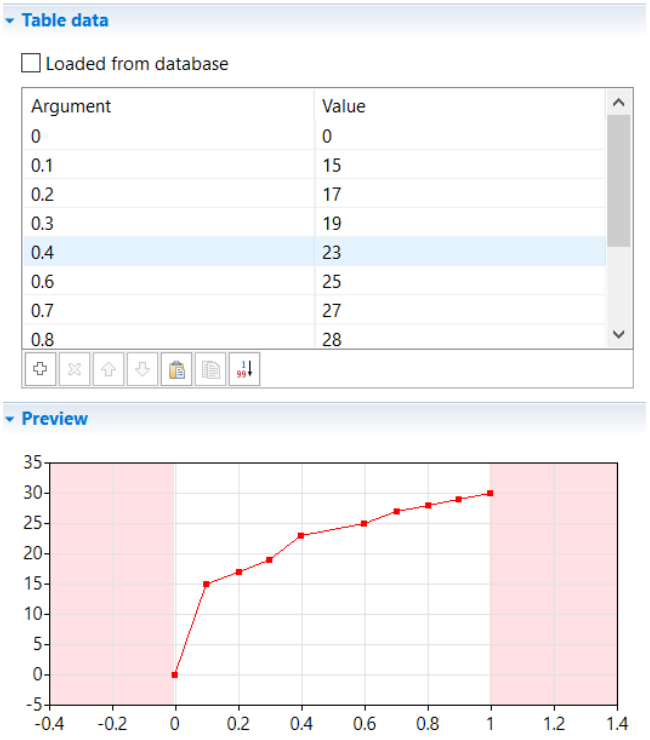
- The random seed is used to generate a random number.
- We use Table Functions for Staffs and Tables in our system for meal times. When using them, random values are generated in them.
- StudentSchecule and PersonnelSchecule are used to represent the random formation of students and personnel interarrivals.

6) INPUT VARIABLES

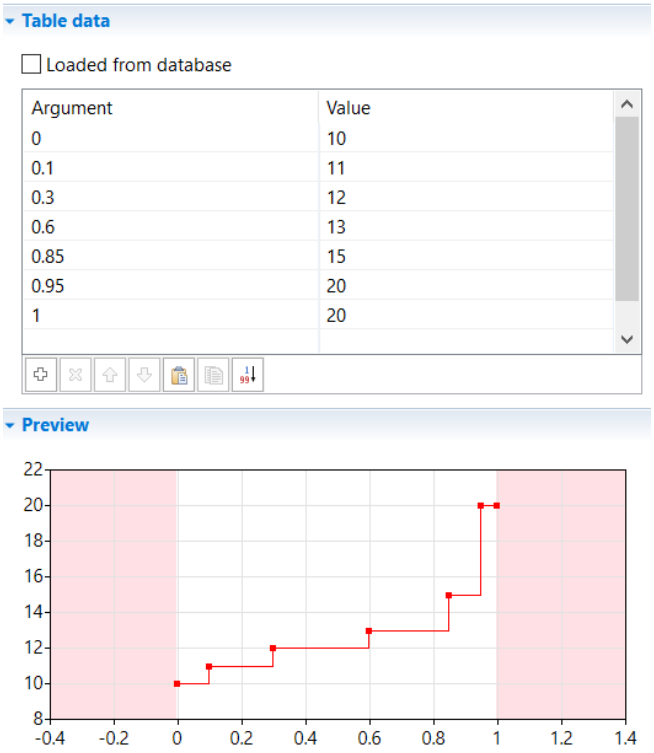
- Input variables in the system are StudentSchedule, PersonnelSchedule, StaffSpeed, EatsTime, OutDesicion, PersonnelQueueDesicion and StudentQueueDesicion, respectively. Our input parameters are discrete value integers.

7) TYPE AND VALUES OF INPUT VARIABLES

- **EatsTime**



- **StaffSpeed**



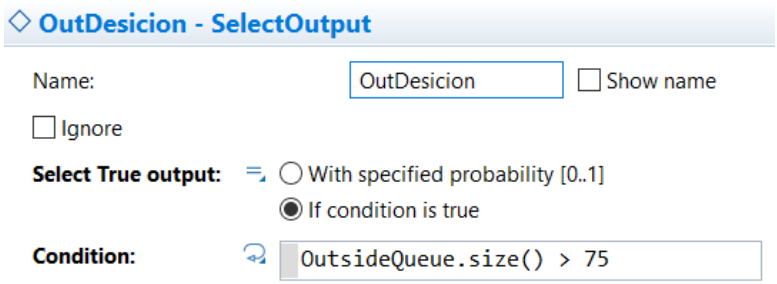
- **StudentSchedule**

Start	End	Value
11:00 AM	12:20 PM	300.0
12:20 PM	1:00 PM	750.0
1:00 PM	2:00 PM	200.0
2:00 PM	11:00 AM	0.0

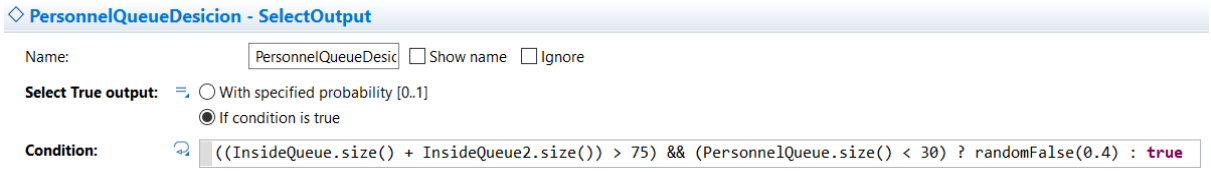
- **PersonnelSchedule**

Start	End	Value
11:00 AM	12:20 PM	30.0
12:20 PM	1:00 PM	100.0
1:00 PM	2:00 PM	20.0
2:00 PM	11:00 AM	0.0

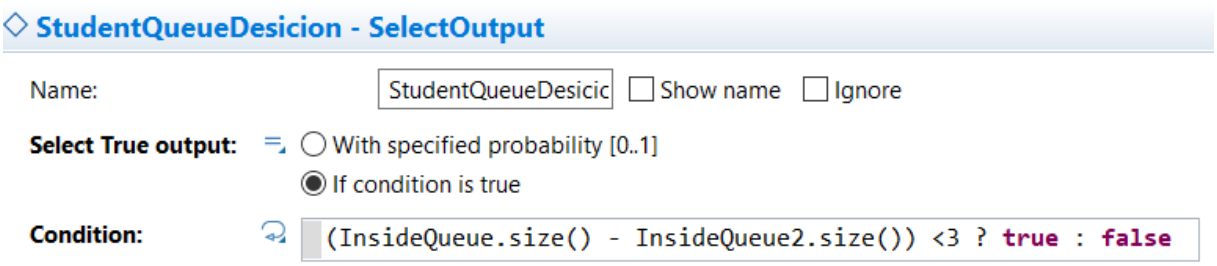
- **OutDesicion**



- **PersonnelQueueDesicion**



- **StudentQueueDesicion**



8) OUTPUT VALUES-VARIABLES AND RESPONSES

Output Variables	Results
Average Number Of Person Waiting In OutsideQueue	1.87
Average Number Of Person Waiting In InsideQueue	4.2
Average Number Of Person Waiting In InsideQueue2	3.83
Average Number Of Person Waiting In PersonnelQueue	1.85
Average Waiting Time In OutsideQueue	1.49 minutes
Average Waiting Time In InsideQueue	5.6 minutes
Average Waiting Time In InsideQueue2	10.18 minutes
Average Waiting Time In PersonnelQueue	6.84 minutes
Busy Time Ratio Of Staff1	0.21
Busy Time Ratio Of Staff2	0.13
Busy Time Ratio Of Staff3	0.1
Busy Time Ratio Of Tables	0.17

- **Average Time Spent In The System Per Person(Without TimeOuts) → Total Time/Number of Agents for every state = 1696.12/274(OutsideQueue) + 8642.72/1010(InsideQueue) + 238.3/1144(Staff) + 25583.94/1144(Tables) = 6.2 + 8.56 + 0.21 + 22.36 = 37.33 min.**

9) REFERENCES

- https://anylogic.help/
- Lecture Slides