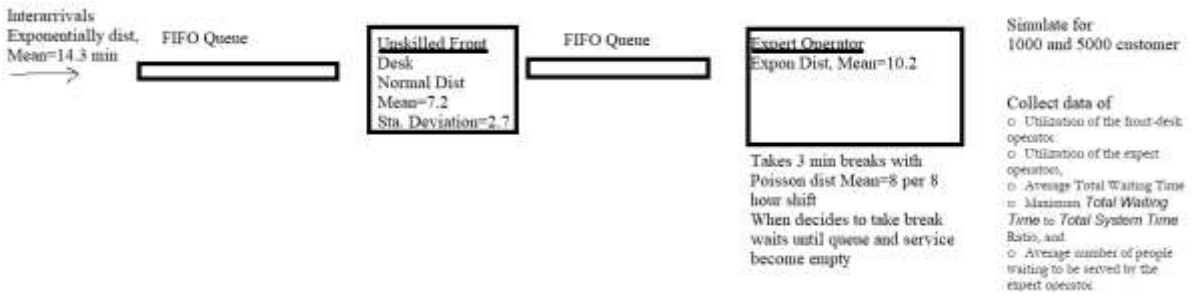
IE306.02 2018-2019/2

ASSINGMENT1

İsmet Dağlı 2014400153

Serkan Özel 2015400123

Ömer Faruk Özdemir 2016400048



-We implement a simulation project using python, simpy.

-In our project there are two operators, and a queue before each operator.

-We implement the system using simply, we created 1 environment and 2 processes.

1 process is the operator.

-For creating waits in the system, we use yield keyword from simpy, it enables us to stop the process for some amount of time.

-Customers arrive with exponentially distributed interarrivals.

-We create customers with this method, and they enter to the first service one by one, when they can.

-Customer's interarrival times are exponentially distributed. We create random numbers with this script.

```
Interarrival=random.expovariate((1/14.3))
```

-We create lognormal random numbers for time spent in first server, with the script below.

```
m1=7.2

v1=2.7

mu1=np.log((m1**2/((v1+m1**2)**0.5)))

variance1=(np.log((v1**2+m1**2)/m1**2))
```

```
duration= random.lognormvariate(mu1,variance1**0.5)
```

-After leaving the first server, we try to put customers to second server.

After customer leaving the first server, we now can think the rest as a new system.

But here we have an exception, customers renege if they are not served within an exponential distributed random time. Our Renege_Rate is 1/60, since mean is 60.

We created random renege time as follows.

```
renege_duration = random.expovariate(RENEGE_RATE)
```

If renege duration is less then time to enter to the second server, our customer leaves the queue and the system after renege duration.

Else enter to the second operator.

And we implemented it like this.

```
results = yield req | env.timeout(renege_duration)
        global total_time
        if req in results:
            print('%s is assigned to the expert operator at %g' % (self.name, self.env.now))
```

```
            total_waiting_time += self.env.now - expert_arrival
            queue2_waiting_times.append(self.env.now - expert_arrival)
            total_waiting_times.append(total_waiting_time)
            yield self.env.process(self.ask_question2())
            print('%s is done with the expert operator at %g' % (self.name, self.env.now))
            total_time = self.env.now
        else:
            print("----------------Customer "+str(self.name)+" reneged after "+ str(self.env.now - expert_arrival) + " of waiting.")
            total_time = self.env.now
```

-Break process works as follows: When the expert operator decides to take a break, the queue in front of the second operator is being saved in a list. And then give_break process constantly checks if all the customers that was in the recorded queue has gone. Then the operator takes the break without waiting the other incoming customers to be served.

- If break decision is already taken taking another decision does not affect the workflow, break is taken when the queue finishes that was recorded when the first decision is taken.

We hold statistics in every event, and they accumulate to total in the end:

Definition of Events:

- Utilization of operator 1 is calculated through adding the all time of services made by operator 1 and it is divided by the total time(ending time of environment).

- Utilization of operator 2:(expert operator) is calculated through adding the all time of services made by operator 2(expert operator) and it is divided by the total time(ending time of environment).

- Average total waiting time is: Waiting time for each customer is calculated while they exit the queue. Then each waiting time is added the list. At the end, all of waiting time are summed up and divided by the total customer number.

- Maximum total waiting time to total system time ratio: For each customer, it is calculated by total waiting time of customer divided by the system time of customer and all of them is added to list. At the end, each the maximum one is selected.

-Average number of people waiting to be served by the expert operator: For each customer, the time at which customer is entered service for expert operator is subtracted by the time at which customer is entered queue 2. So, they're waiting time is calculated on queue 2. Then, each of them is summed up and calculated by total time.

First part of our code is for 1000 sample, and second part of our code for 5000 sample.

-Outputs

1000 sample

Utilization of operator 1 is:0.49637058282789154

Utilization of operator 2 is:0.6320799744064244

Average total waiting time is:12.138848436248429

Maximum total waiting time to total system time ratio 0.9339563412820246

Average number of people waiting to be served by the expert operator 0.4592841721662935

5000 sample

Utilization of operator 1 is:0.4960601105509688

Utilization of operator 2 is:0.6072939701964999

Average total waiting time is:12.144721767115254

Maximum total waiting time to total system time ratio 0.9636788076011426

Average number of people waiting to be served by the expert operator 0.5342478531869789

Interpretation of Results:

We have two different sets of results. At first, 1000 sample is used. At the second, 5000 sample is used. We have slightly different results. We can easily say that “the more sample the better and realistic results” We have compare for each results:
1-utilization of operator 1 : the 5000 sample is slightly less than the 1000 sample. The difference is 0.003~. Therefore it is so slight that no absolute result can’t be inferred.

2-utilization of operator 2 : the 5000 sample is less than the 1000 sample. The difference is 0.3~. It may not be ignored just like in the first case. The result may not different from each other because the random numbers in the first 1000 may slightly different from the real probability. When we incraese the number of sample, we decrease the difference between real probability and our result.

3- Average total waiting time: it should be again inferred from the fact that more data we have better result we get.

4-Maximum total waiting time to total system time ratio: the 5000 sample should be higher than the 1000 sample. in the 5000 sample, we increase the probability of maximum ratio because we have more probability. But it should be forgotten that 1000 sample may rarely have the more ratio.

5-Average number of people waiting to be served by the expert operator: Again, the more data the more real case example is calculated.