TMA1301 Computational Methods

Assignment

Title: Queue Simulation

Prepared by:

No.	Name and ID	E-mail
1	Muhammad Uzair Bin Abdul Razak (1191303163)	1191303163@student.mmu.edu.my
2	Kishen Kumar A/L Sivalingam (1191101423)	1191101423@student.mmu.edu.my
3	Avinash Imanuel A/L Gana Raj Imanuel (1181103370)	1181103370@student.mmu.edu.my
4	Satishnair S/O Krishnan(1181103485)	1181103485@student.mmu.edu.my

Details of the simulation

- ➢ Firstly, the simulation will display choices of Random Number Generator for users to choose.

- ➢ The user needs to enter the number of customers for the whole simulation and maximum number of customers allowed in the centre.

- ➢ Customers need to Enter any key to record their temperature, arrival time, reaching the centre and departure time.

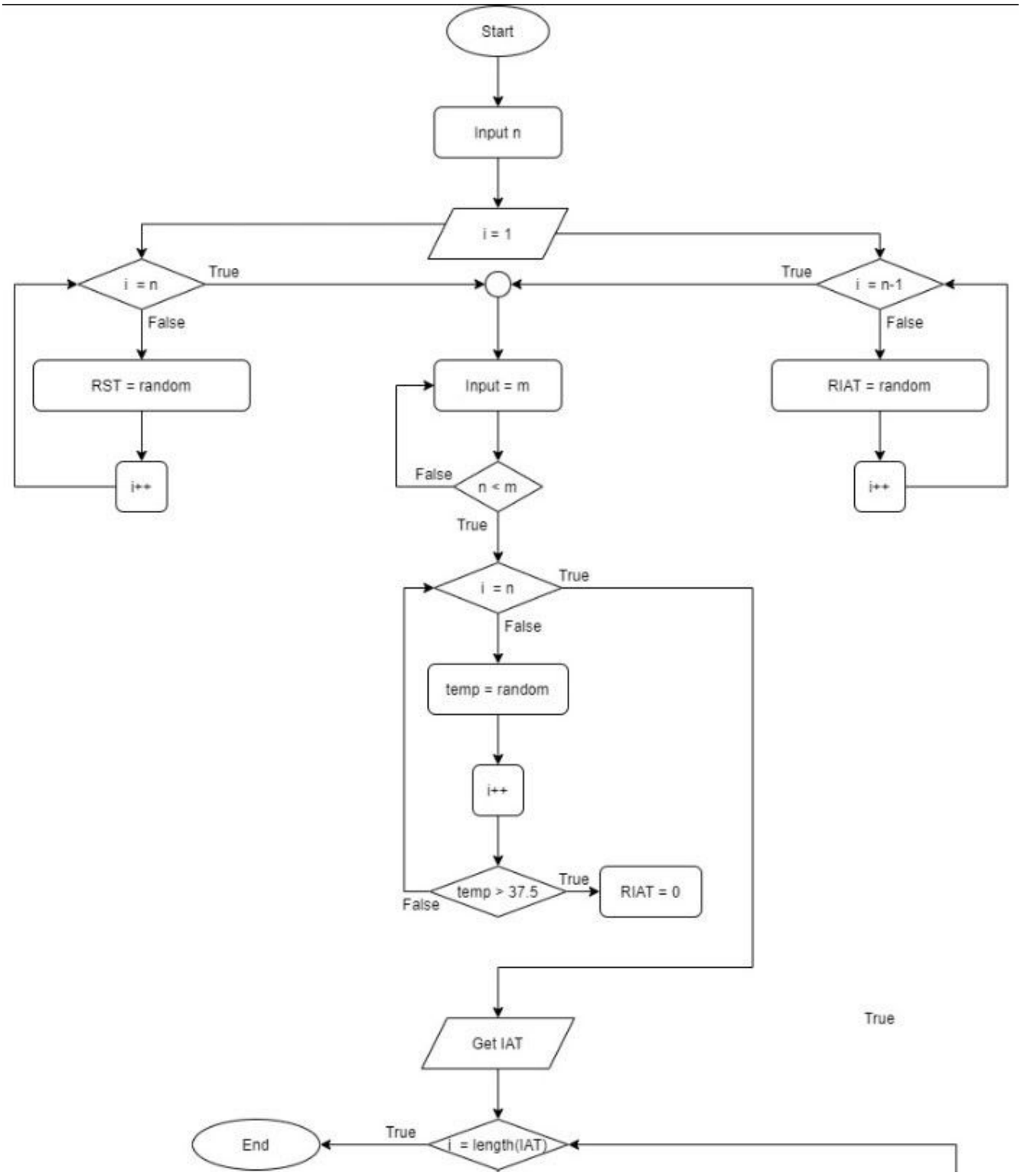
- ➢ Then the simulation will exhibit the message of arrival, entering the centre and departure time of the customers from time to time.

- ➢ The simulation then provides the table of the service time for different counters and the table of the inter- arrival time.

- ➢ The system we created will display the results of the simulation into 2 tables

- ➢ Lastly, the system will evaluate the results of the simulation, for example average waiting time, average time spent, probability that a customer has to wait and average service time for each counter.

Flow-Chart


Start
Input n
i = 1
i = n
True
False
RST = random
i++
Input = m
False
n < m
True
i = n
True
False
temp = random
i++
temp > 37.5
True
False
RIAT = 0
i = n-1
True
False
RIAT = random
i++
Get IAT
True
End
True
i = length(IAT)

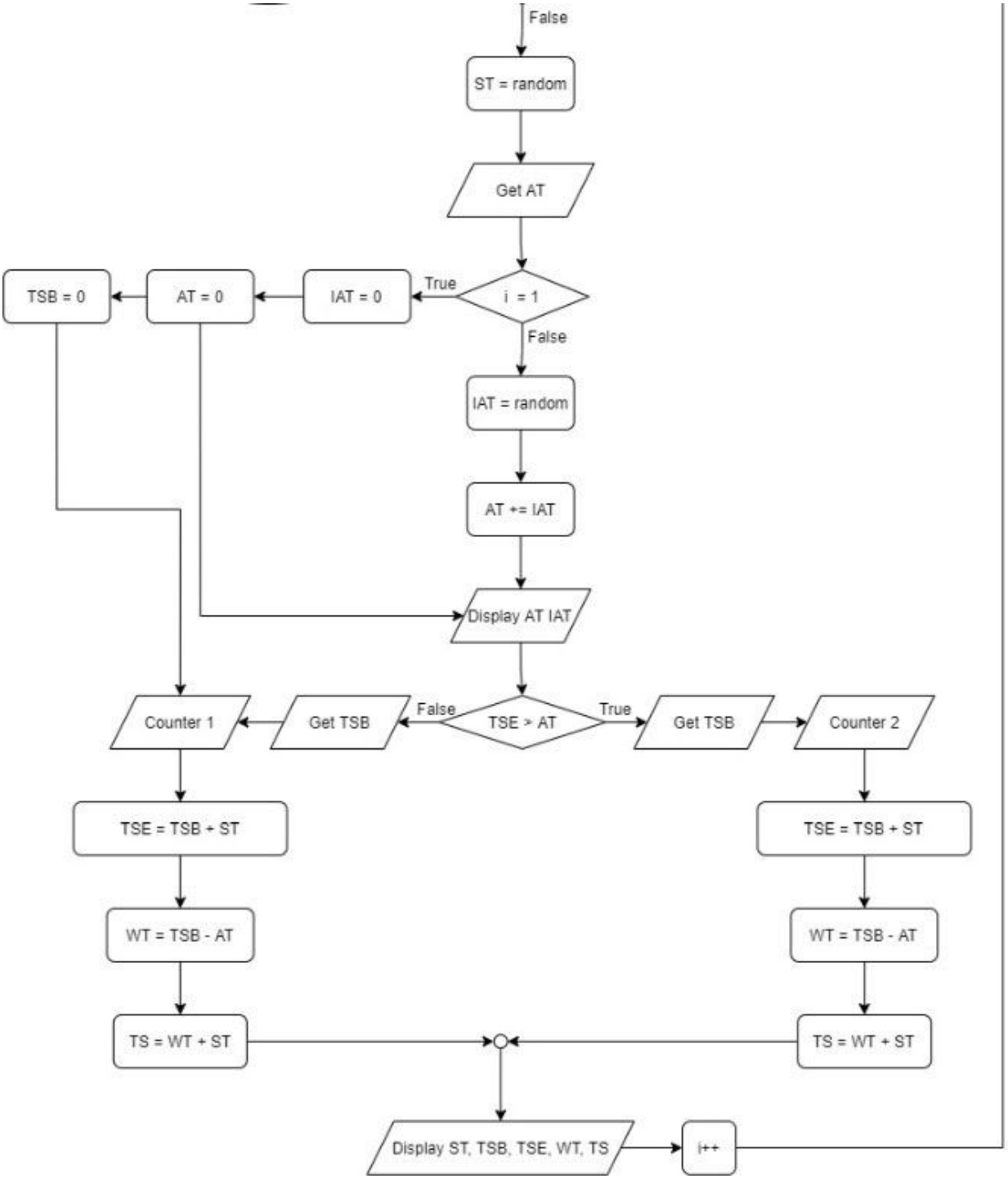


Figure 1.0: Flowchart representing the general algorithm of the queue system throughout the simulation

Explanation for the important source codes

1. Random Number

    1.1 Temperature

    Output = (40.0-30.0)*rand(1,5)+30.0

    Generates a random number between the given range

    1.2 Inter Arrival Time

    Output = [round(rand(1, x-1)*100)]

    Generates a random number between the given range

    1.3 Service Time

    Output = [round(rand(1, x)*100)]

    Generates a random number between the given range

2. Linear Congruential Generator

    2.1 Multiplicative

```
intArr(1) = randi(0, 100);
    for i=2:x-1
       intArr(i) = mod((13*intArr(i-1)), 55);
    end
    interArrivalTime = [intArr]

    serArr(1) = randi(0, 100);
    for i=2:x
       serArr(i) = mod((13*intArr(i-1)), 55);
    end
    serviceTime = [serArr]
```

Generates a random number while implementing the Multiplicative LCG for Inter arrival time and Service time

2.1 Additive

```
    intArr(1) = randi(0, 100);
for i=2:x-1
    intArr(i) = mod((intArr(i-1)+78), 34);
end
interArrivalTime = [intArr]

serArr(1) = randi(0, 100);
for i=2:x
    serArr(i) = mod((serArr(i-1)+65), 75);
end
serviceTime = [serArr]
```

Generates a random number while implementing the Additive LCG for Inter arrival time and Service time

3. Custom Random Number Generator

1.2 Inter Arrival Time
Output = [mod(round(rand(1, x-1)*80 + 35),101)]
Generates a random number between the given range

1.3 Service Time
Output = [mod(round(rand(1, x)*75 + 37),71)]
Generates a random number between the given range

4. Inter Arrival Time Table

4.1 Inter Arrival Time range

```
//Determines the Inter Arrival time value according to the range

        for i = 1:length(interArrivalTime)
            if((interArrivalTime(i) >= firstrand(1)) &&
        (interArrivalTime(i) <= cdfrands(1)))
                IaT(i+1) = 1 ;
            end
            if((interArrivalTime(i) >= firstrand(2)) &&
        (interArrivalTime(i) <= cdfrands(2)))
                IaT(i+1) = 2 ;
            end
            if((interArrivalTime(i) >= firstrand(3)) &&
        (interArrivalTime(i) <= cdfrands(3)))
                IaT(i+1) = 3 ;
            end
            if((interArrivalTime(i) >= firstrand(4)) &&
        (interArrivalTime(i) <= cdfrands(4)))
                IaT(i+1) = 4 ;
            end
            if((interArrivalTime(i) >= firstrand(5)) &&
        (interArrivalTime(i) <= cdfrands(5)))
                IaT(i+1) = 5 ;
            end
        end
```

5. Service Time table

5.1 Service time range Counter 1

```
  //Determines the Service time value  for Counter 1 according to the
range


      for i = 1:x
         if((serviceTime(i) >= firstrand(1)) &&  (serviceTime(i) <=
cdfrands(1)))
            st(i) = 2 ;
         end
         if((serviceTime(i) >= firstrand(2)) &&  (serviceTime(i) <=
cdfrands(2)))
            st(i) = 3 ;
         end
         if((serviceTime(i) >= firstrand(3)) &&  (serviceTime(i) <=
cdfrands(3)))
            st(i) = 4 ;
         end
         if((serviceTime(i) >= firstrand(4)) &&  (serviceTime(i) <=
cdfrands(4)))
            st(i) = 5 ;
         end
         if((serviceTime(i) >= firstrand(5)) &&  (serviceTime(i) <=
cdfrands(5)))
            st(i) = 6 ;
         end

      end
```

5.1 Service time range Counter 2

```
  //Determines the Service time value  for Counter 2 according to the
range

      for i = 1:x
         if((serviceTime2(i) >= firstrand(1)) &&  (serviceTime2(i) <=
```

```
cdfrands(1)))
            st2(i) = 3 ;
        end
        if((serviceTime2(i) >= firstrand(2)) &&  (serviceTime2(i) <=
cdfrands(2)))
            st2(i) = 4 ;
        end
        if((serviceTime2(i) >= firstrand(3)) &&  (serviceTime2(i) <=
cdfrands(3)))
            st2(i) = 5 ;
        end
        if((serviceTime2(i) >= firstrand(4)) &&  (serviceTime2(i) <=
cdfrands(4)))
            st2(i) = 6 ;
        end
        if((serviceTime2(i) >= firstrand(5)) &&  (serviceTime2(i) <=
cdfrands(5)))
            st2(i) = 7 ;
        end

    end
```

6. Calculations

6.1 Average waiting time

```
TotalWaitingTime = 0;

for i = 1:length(interArrivalTime)
   TotalWaitingTime = TotalWaitingTime + WT(i);

end

AvgWaitingTime = TotalWaitingTime/length(serviceTime);
```

6.1 Average Time Spent

```
TotalTimeSpent = 0;

for s = 1:length(interArrivalTime)
   TotalTimeSpent = TotalTimeSpent + timeSpent(s);
end

AvgTimeSpent = TotalTimeSpent/length(serviceTime);
```

6.1 The probability that a customer has to wait

```
count = 0;

for m = 1:length(interArrivalTime)
   if (WT(m) > 0)
      count = count + 1;
   end
end

probability = count/length(serviceTime);
```

6.1 Average service time for each counter

```
% counter 1

AvgServiceTime = sums/length(serviceTime);

printf('\n\nThe Average Service Time For Counter 1: %0.1f \n',
AvgServiceTime)


% counter 2

AvgServiceTime2 = sums2/length(serviceTime);
```

```
    printf('\n\nThe Average Service Time For Counter 2: %0.1f ',
AvgServiceTime2)

    disp(' ')
```

Screenshots of the tables and the evaluation results

Service Time Table

Service Time Table

Counter 1

Service Time	Probability	CDF	Random number range
2	0.10	0.10	1 - 10
3	0.50	0.60	11 - 60
4	0.10	0.70	61 - 70
5	0.05	0.75	71 - 75
6	0.25	1.00	76 - 100

Counter 2

Service Time	Probability	CDF	Random number range
3	0.32	0.32	1 - 32
4	0.30	0.62	33 - 62
5	0.20	0.82	63 - 82
6	0.04	0.86	83 - 86
7	0.14	1.00	87 - 100

Inter-Arrival Time Table

Inter-Arrival Time Table

Inter-Arrival Time	Probability	CDF	Random number range
1	0.30	0.30	1 - 30
2	0.10	0.40	31 - 40
3	0.20	0.60	41 - 60
4	0.25	0.85	61 - 85
5	0.15	1.00	86 - 100

Simulation Table

Customer no.	Temperature (Celsius)	Random Number for Inter Arrival-Time	Inter-Arrival Time	Arrival Time	No. of Customers In the Centre	Time entering The centre
1	33.03	-	-	0	0	0
2	37.88	0	0	0	0	0
3	36.97	12	1	1	3	2
4	36.70	22	1	2	3	3
5	32.72	32	2	4	4	4

		Counter 1			Counter 2				
Customer No.	RN ServiceTime	ServiceTime	Time Service Begins	Time Service Ends	ServiceTime	Time Service Begins	Time Service Ends	Waiting Time	Time spent
1	67	4	0	4				0	4
2	57				4	2	6	0	4
3	47				4	3	7	0	4
4	37	3	4	7				0	3
5	27				3	6	9	0	3

Evaluation Results

```
The Average Waiting Time: 0

The Average Time Spent: 3

The probability that a customer has to wait: 0


The Average Service Time For Counter 1: 1.4


The Average Service Time For Counter 2: 2.2
```