\

CHAPTER – 01

PROJECT INSTIGATION

1.0 Introduction

Simulation - based approaches have been applied to various system optimization problems in industries. Although simulation approaches are popular and powerful, it should be emphasized that simulation is just one of possible experiments for an interested system without affecting the existing system. When we get the result of simulation and they are found apt and suitable then imposed on the system.

Talking about the road and highways, a single traffic flow distribution and one type of tollbooth at the toll plaza were assumed to build a queuing model. However, when there are various types of traffic flow patterns for different time periods and not identical tollbooths, the problems become more complex and time consuming to analyse. Due to these drawbacks from the operations research-based methods, simulation-based approach is the apt methodology that has been used for the analysis of toll plaza performance measures in the literature.

1.1 Objectives

The objectives of the system are-

- To create a model of the existing system.
- To identify the factors which control the system.
- To predict the behaviour of the system.
- To quantitatively represent the system using rules.
- To check the efficiency of the “Queue Theory” upon the system by applying it to the system.
- To clearly depict the study in a format that is self- explanatory.
- Handle the traffic flow without any problem.

The main problem of toll plaza is how we optimize the queue length of vehicles and the time of customers in the system. Next problem of the highway toll plaza is to reduce waiting time of customers. If the time taken by the vehicles in the toll plaza were as short as possible, the toll plaza would obtain a larger traffic capacity. To improve the traffic capacity, the relationships among the amount of the time taken by vehicles entering the system, service time, the number of tollbooths and the departing time of the vehicles must be considered simultaneously.

This project module is a simulation system of a real time toll plaza system and it implement the Queue Theory in which the real time focus is provided to the balancing the cost of waiting against the cost of idle time of the server. Moreover a special attention is provided to the load balancing such that load is equally distributed between the each server.

1.2 Scope

This project has a large scope as it has the following features which help in making it easy to use, understand and modify it:

- Graphical study of the simulation
- Traffic management at a toll plaza is shown
- Based upon arrival pattern, the behaviour of the servers, the service time, load factor, the decision for a vehicle is taken instructing which server's queue it should join
- A separate queue is managed for the server, which manages the vehicle waiting to get the response time from the server.
- Whenever a decision is made for the vehicle entered in to the system. On the basis of this decision the toll plaza instruct the vehicle to join a particular queue.
- The server's idle time and the server's cumulative departure time is manipulated accordingly.

This simulation system can be used by the people interested in study the Toll Plaza system and can be used by teachers to illustrate the concept of Queue Theory model and how it can be used in simulation and modelling of a system to the students.

Main Points are:-

- Graphical interface for depicting the behaviour of the system
- Traffic at Toll Plaza Management Strategy.
- Load balancing amongst various servers
- Real-time system properties are used.
- Vehicle arrives in a stochastic manner.

Here are some of the assumptions that we had in this system that are:
(1) The arrivals to the toll plaza are taking place in a random fashion.
(2) The service times are exponentially distributed.
(3) The customers are served on First Come First Serve basis.
(4) There is a limit of the customers at toll plaza because this system is just for demonstrating and analysing.

This stimulatory system is a 3 server system.

1.3 Platform Required

The system's front-end and back-end are both developed using the "C" Language. Using the "graphics.h" library of the "C" Language the graphical view is prepared. Below are some of the inductor lines about the "C" language.

C is a general-purpose, procedural, imperative computer programming language developed in 1972 by Dennis M. Ritchie at the Bell Telephone Laboratories to develop the UNIX operating system. C is the most widely used computer language. It keeps fluctuating at number one scale of popularity along with Java programming language, which is also equally popular and most widely used among modern software programmers.

It can be run using the Dev C++ (a gcc compiler for windows) enabled with the graphics.h library. It is a free full-featured integrated development environment (IDE) distributed under the GNU General Public License for programming in C and C++. It is written in Delphi. It is bundled with, and uses, the MinGW or TDM-GCC 64bit port of the GCC as its compiler. Dev-C++ can also be used in combination with Cygwin or any other GCC-based compiler Dev-C++ is generally considered a Windows-only program, but there are attempts to create a Linux version: header files and path delimiters are switchable between platforms.

1.4 Overview

The project **"TOLL - GATE"** is a simulation project used to analyse the optimal behaviour of a toll plaza and come out with an approach in order to make the system work efficiently and enhance the system performance by introducing some add-ons functionality like load balancing, server - queue management, management of customers on the basis of the arrival pattern, service pattern and maintaining the queue for each server distinctly in the order in which the customer comes to the server i.e., FCFS mode.

CHAPTER – 02

PROJECT ANALYSIS

A sensitivity analysis of toll plaza performance was used to evaluate possible alternatives of the toll plaza configuration on the basis of only the number of manual booths. In this study we first assume that the number of tollbooths is fixed.

2.1 Product Perspectives

The perspective of the product is quiet simple yet seems important one. We use the simulation study to make a system which depicts like a real system so that if we want to implement any new policy to the system can be first tested upon the simulated system and based upon the results obtained the changes can be done to the policy and then retested. In this way, when we get an apt policy to our system then it is applied to the real-world system. This whole procedure is implemented to ensure that any bad result of the policy while it is in testing phase would not disturb the system.

The above explained perspective is well supported by our system. First we implement the simulated system and made changes to the policies as per the requirement and then we can implement this on the real time system.

This same load balancing approach using the queue theory is well implemented by the Banking Systems now-a-days.

2.2 Product Function

Delays and queuing problems are most common in daily life situation, whether in a supermarket, a bank, a ticket office, or in traffic. The stochastic characteristics of queuing processes make it difficult to fully predict queuing behaviour. Waiting times in queuing systems are not linear. When the workload, or occupancy rate, of a server doubles, waiting times may quadruple. While if the workload of server is higher than 90 percent, waiting times can become 10 times as large as the actual service or handling times. After workload, a second main reason for waiting times is the notion of variability. If the handling times varied, this average wait time can change arbitrarily. Higher variability in arrival patterns or service times thus directly results in higher waiting times. To keep waiting times low as possible, it is thus important to keep the variability in service times as low as possible.

The functioning of the product is to optimize the queue for each server (toll booth) of the system so that there is a proper load balancing among the different server. These queues are optimized on the basis of the arrival pattern, the service time of the vehicle, the waiting time of the system.

Therefore, the system makes a decision for every new entering vehicle into the system keeping the arrival time and the time when the system will be free.

The decision making process of the system is explained below:

At(k) : arrival time for the k^{th} element

Decision Making Process

			Facility 1					Facility 2					Facility 3				
k	At(k)	CAT(k)	ST(k,1)	CDT(k,1)	IDT(k,1)	QL(1,k)	WT(1,k)	ST(k,2)	CDT(k,2)	IDT(k,2)	QL(2,k)	WT(2,k)	ST(k,3)	CDT(k,3)	IDT(k,3)	QL(3,k)	WT(3,k)
1	0	0	10	10	0	-	-	-	-	-	-	-	-	-	-	-	-
2	5	5	-	-	-	-	-	3	7	5	-	-	-	-	-	-	-
3	10	15	-	-	-	-	-	-	-	-	-	-	10	25	15	-	-
4	7	22	-	-	-	-	-	12	34	15	-	-	-	-	-	-	-
5	3	25	5	30	15	-	-	-	-	-	-	-	-	-	-	-	-
6	2	37	-	-	-	-	-	-	-	-	-	-	15	42	2	-	-
7	7	34	15	49	4	-	-	-	-	-	-	-	-	-	-	-	-
8	6	40	-	-	-	-	-	10	50	6	-	-	-	-	-	-	-
9	4	44	-	-	-	-	-	-	-	-	-	-	10	56	0	1	2
10	1	45	5	54	0	1	4	-	-	-	-	-	-	-	-	-	-

CAT(k) : cumulative arrival time for the kt^h element
ST(k,i) : Service time for the k^{th} element at the i^{th} server
CDT(k,i) : cumulative departure time for the k^{th} element at i^{th} server
IDT(k,i) : idle time for the server waiting for the k^{th} element
QL(i,k) : queue length of the i^{th} server adding k^{th} element
WT(i,k) : waiting time for the response by k^{th} element in i^{th} server queue

2.3 Constraints

Arrival Process

- How customers arrive e.g. singly or in groups (batch or bulk arrivals)
- How the arrivals are distributed in time (e.g. what is the probability distribution of time between successive arrivals (the ***inter arrival time distribution***))
- Whether there is a finite population of customers or (effectively) an infinite number

The simplest arrival process is one where we have completely regular arrivals (i.e. the same constant time interval between successive arrivals).

A Poisson stream of arrivals corresponds to arrivals at random. In a Poisson stream successive customers arrive after intervals which independently are exponentially distributed. The Poisson stream is important as it is a convenient mathematical model of many real life queuing systems and is described by a single parameter - the average arrival rate. Other important arrival processes are scheduled arrivals; batch arrivals; and time dependent arrival rates (i.e. the arrival rate varies according to the time of day).

In our system this arrival pattern of vehicle is stochastic rather than deterministic. There may be vehicle arriving continuously/simultaneously or they can arrive with a gap of time means at discrete points of the time

Service Mechanism

- Description of the resources needed for service to begin
- How long the service will take (the ***service time distribution***)
- The number of servers available
- Whether the servers are in series (each server has a separate queue) or in parallel (one queue for all servers)
- Whether pre-emption is allowed (a server can stop processing a customer to deal with another "emergency" customer)

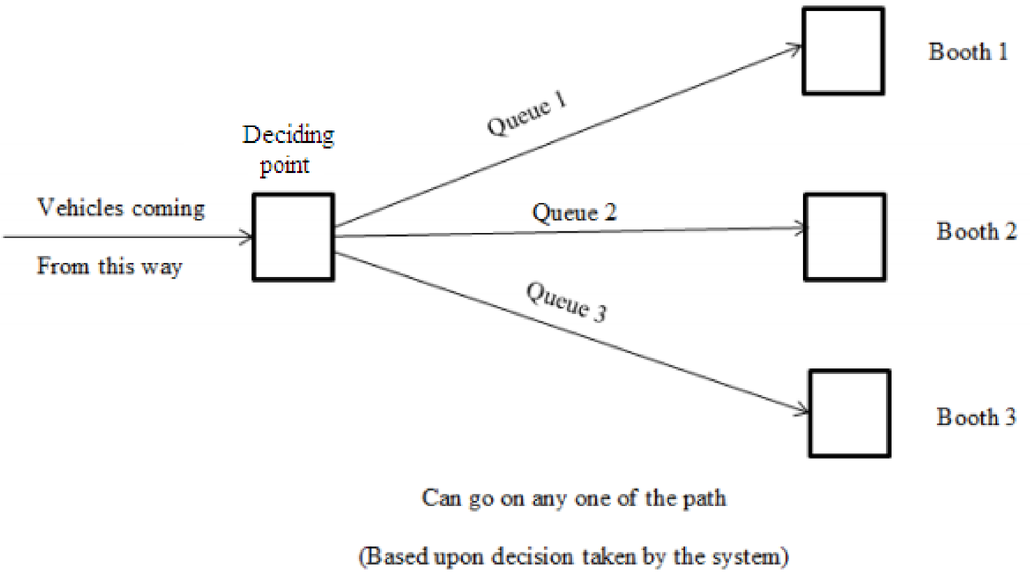
Assuming that the service times for customers are independent and do not depend upon the arrival process is common. Another common assumption about service times is that they are exponentially distributed.

In our system the number of server are limited i.e., three and will serve the customers on the FCFS (First Come First Serve) basis.

2.4 Architecture Design

The architectural design of the system can be expressed using the fallowing diagram

Traffic Flow Type	Probabilistic



Number of Booth	Three(1,2, and 3)
Type of Distribution	Poisson Distribution

The system mainly consist of two parts

1. Graphical Interface

> The graphical interface consist of the graphical end through which user can view the simulation process graphically and can see how the process is going on.

2 Logical Function:

> The logical function or this part primarily analyse the current system status and then make decision for a new entrant to join a server's queue.

Therefore there is a continuous and mutual interaction between these two parts of the system.

The relationship and dependency between these two components of the project can be best represented by the following block diagram.

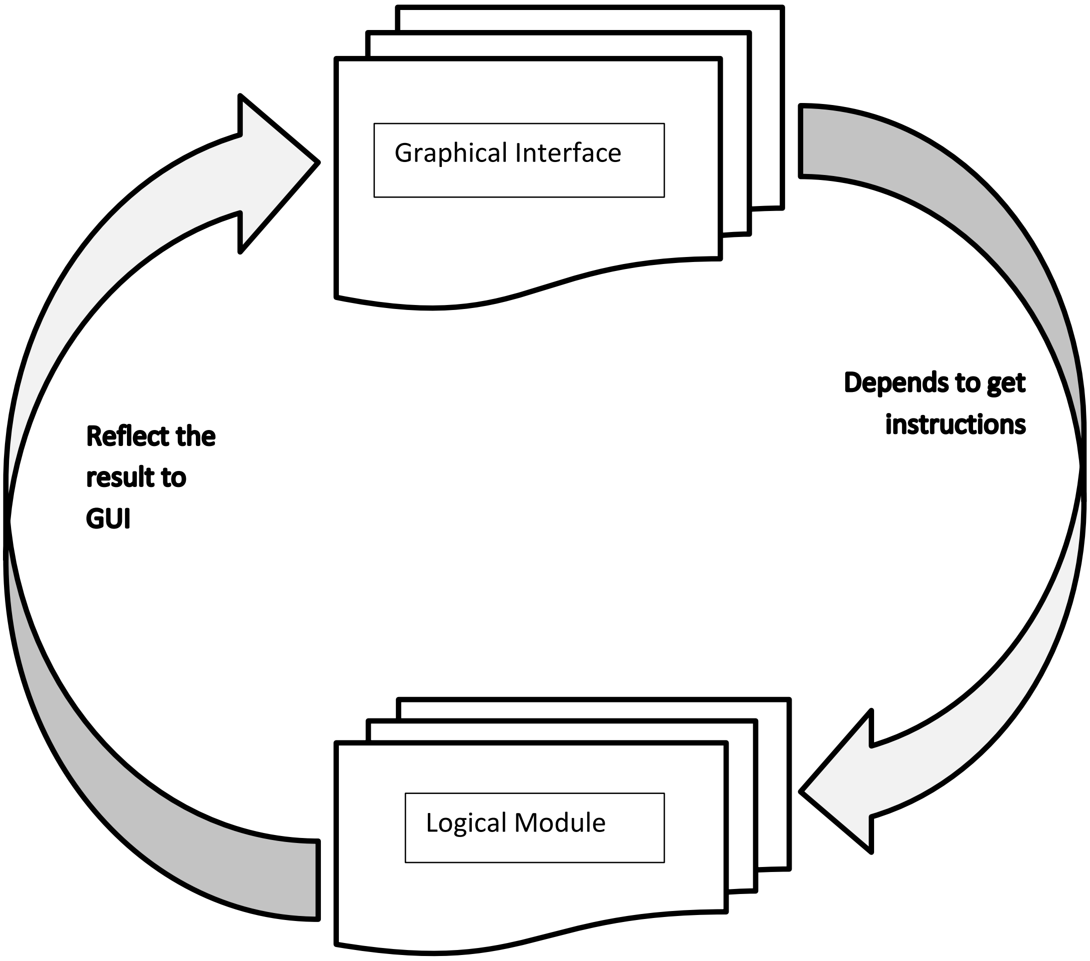
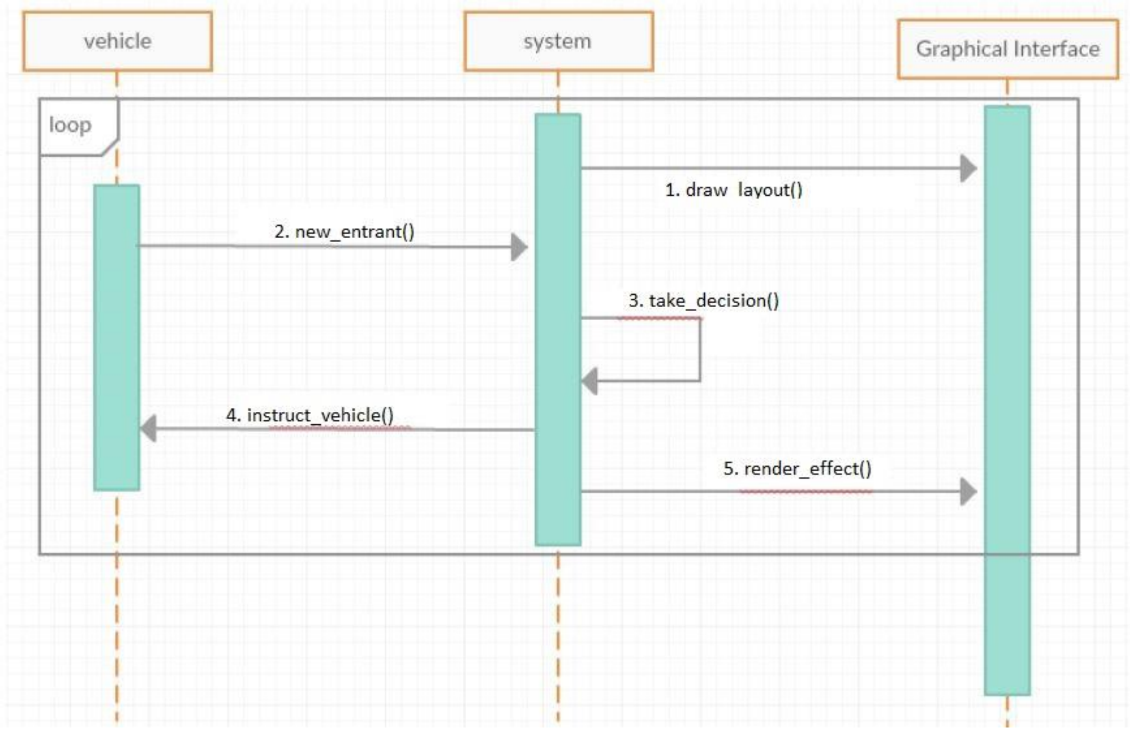


Figure: Mutual relationship between modules

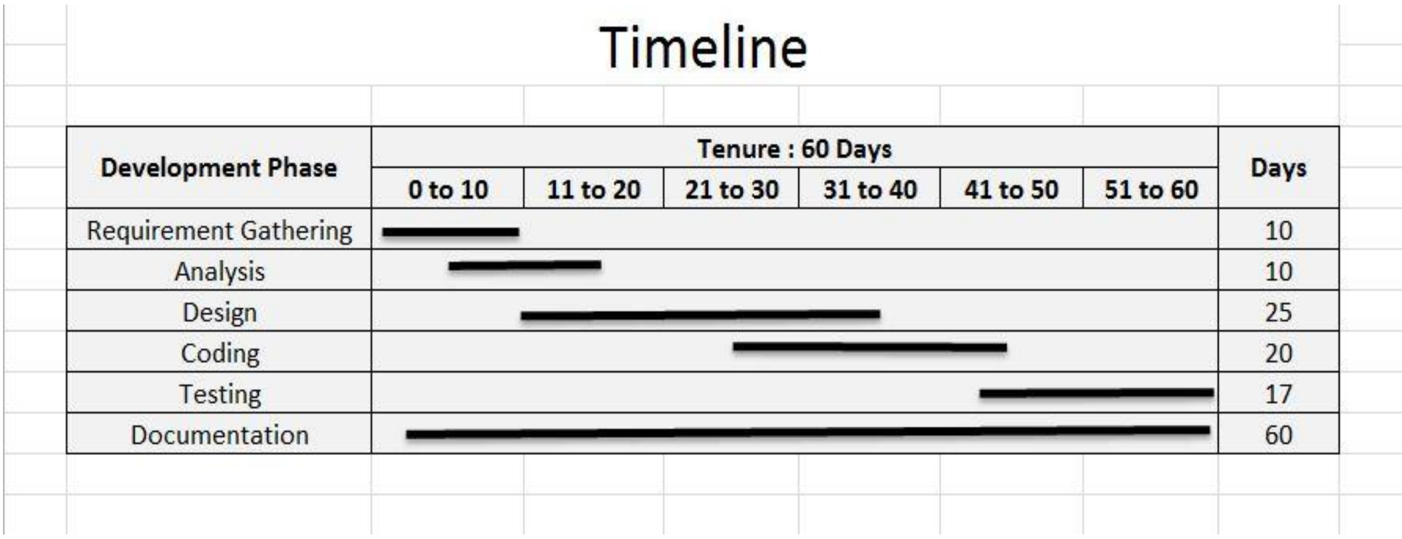
CHAPTER – 03

SYSTEM DESIGN DIAGRAMS

3.1 Sequence Diagram



3.2 Timeline Diagram or Gantt chart


Timeline

Development Phase	Tenure : 60 Days						Days
	0 to 10	11 to 20	21 to 30	31 to 40	41 to 50	51 to 60	
Requirement Gathering							10
Analysis							10
Design							25
Coding							20
Testing							17
Documentation							60

CHAPTER – 04

FORM DESIGN

4.1 List of all modules

Following is the list of all modules that are used along with the functions present inside them and the utility of the module and the functions

1. **Toll**
   This module is the instantiation module of the whole project. It deals with the adequate including of the header files and some of user defined cpp modules. Moreover It present the first look of the project and passes the control to other modules. In all total it consist of the main function of the project which is also the starting point to run the system.

2. **About**
   This module have no such value concerning with the main functioning of the system. This module presents the developer's details and then returns the control back to the main function. It have a delay times for which this module hold the page then returns control.

3. **Tollgraph**
   This is the module that actually deals with the graphical representation of the project. We can also call it as the front-end creating module of the project. The module consist of many functions inside it as

   makgraph(): It created the basic layout of the toll plaza structure.

   check_queue_length(ch): this function check for the present queue length for the all the toll booth and print the vehicle as per the queue length

   booth1(),booth2(),booth3(): these function helps in displaying the movement of the vehicle towards the particular booth on the basis of the decision made at the decision point.

   maketoll(): this function is the principle function that control the above listed functions in this module as well as it also provide a bridge between the graphic controlling function and the decision making functions.

4. **Auxil**

   This module deals with the decision making process of the toll-plaza and analysing the parameter of the toll booth the module will return the booth number that the vehicle should join.

   This module also contains of some functions as well as structures that are used.

   Following are the structures that are defined in this module:

   structure car: this is a structure named as car. It contains fields such as the arrival time, service time, waiting time, departure time for every instant or we can say every instant of this structure represents one vehicle.

   structure tollserver: there are three instant of this structure each for one booth. Each booth has individual property of ideal time, queue length, cumulative departure time.

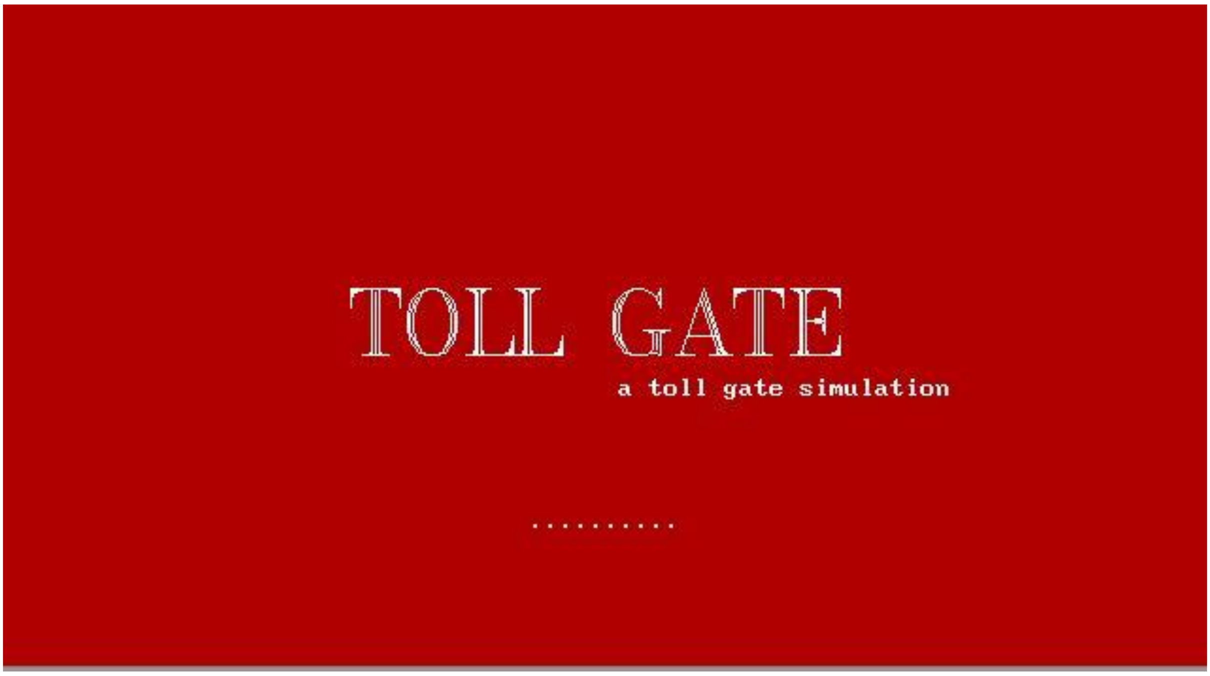
   Following are the functions defined inside this module:

   Init(): this function initialises the arrival time and service time for all the instances of the structure car. Moreover the arrival time is chosen randomly between the range of 1 to 5 and then cumulative arrival time is calculated and is assigned as the arrival time of the vehicle. Similarly the service time is chosen randomly between the ranges 1 to 15 and is assigned to the service time of the instant of the respective car.

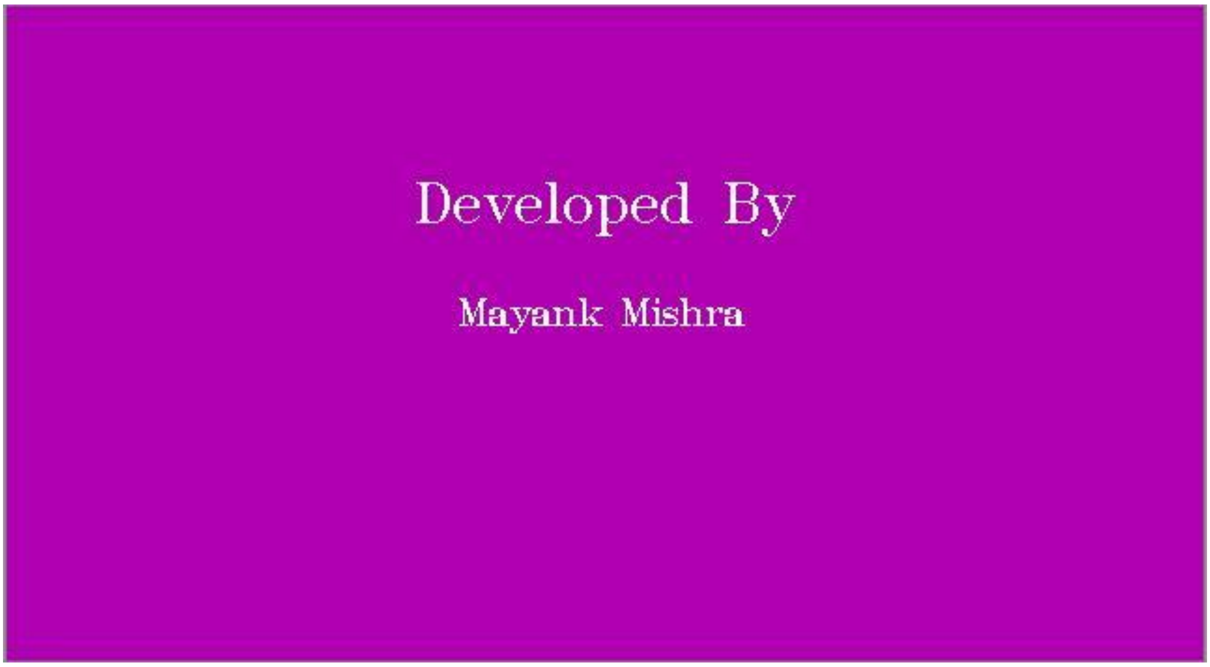
   Decision(): this function takes decision for the vehicle. The decision is taken primarily on the cumulative departure time property of the tollserver.

   Updatequeuelen(): this function will update queue length when a new vehicle joins the particular queue for the car or the vehicle depart from the booth on completion of the service time.

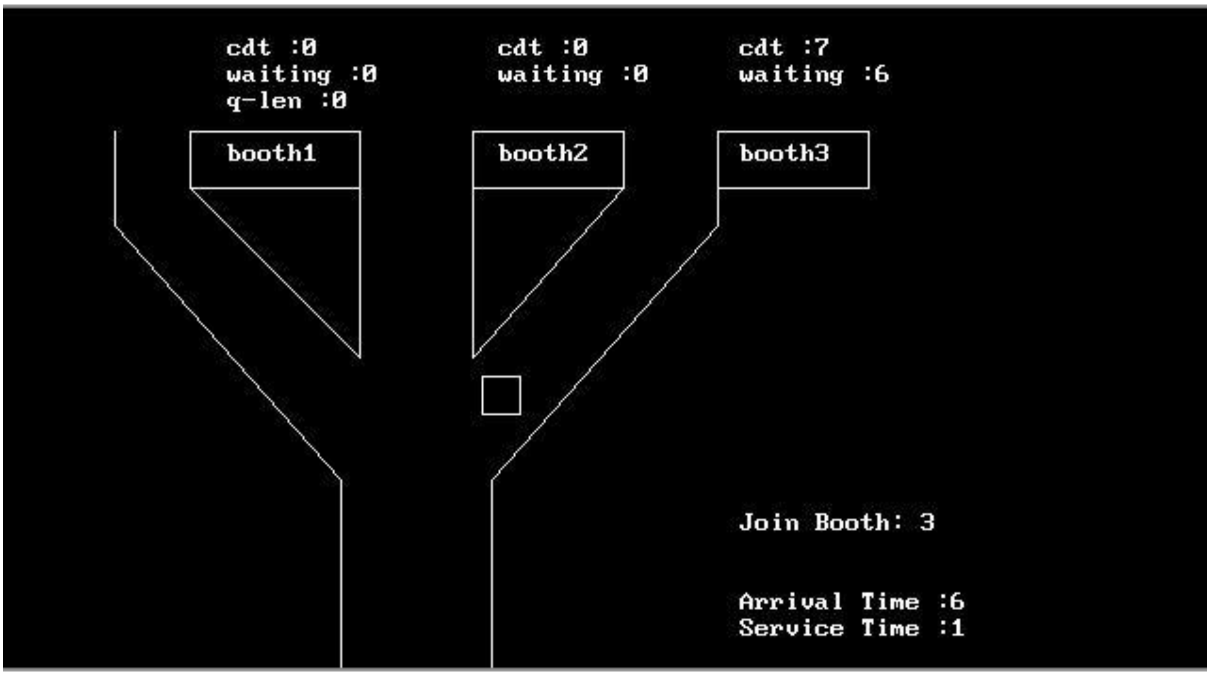
4.2 Project Screen Shot



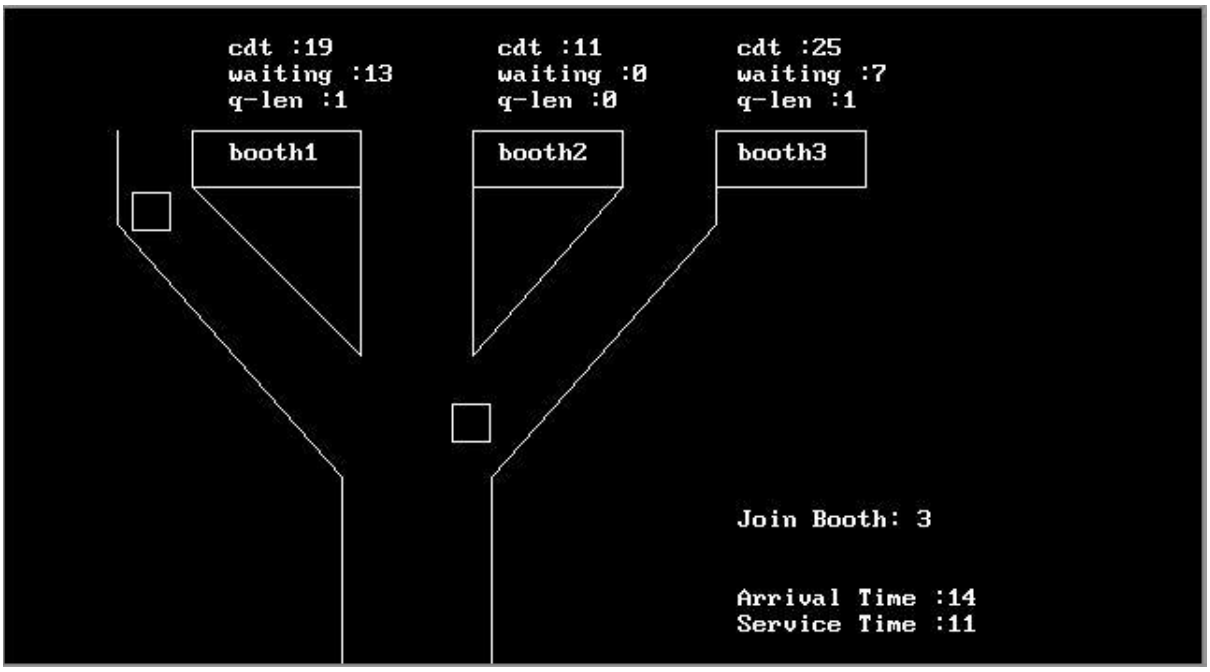
Opening Screen



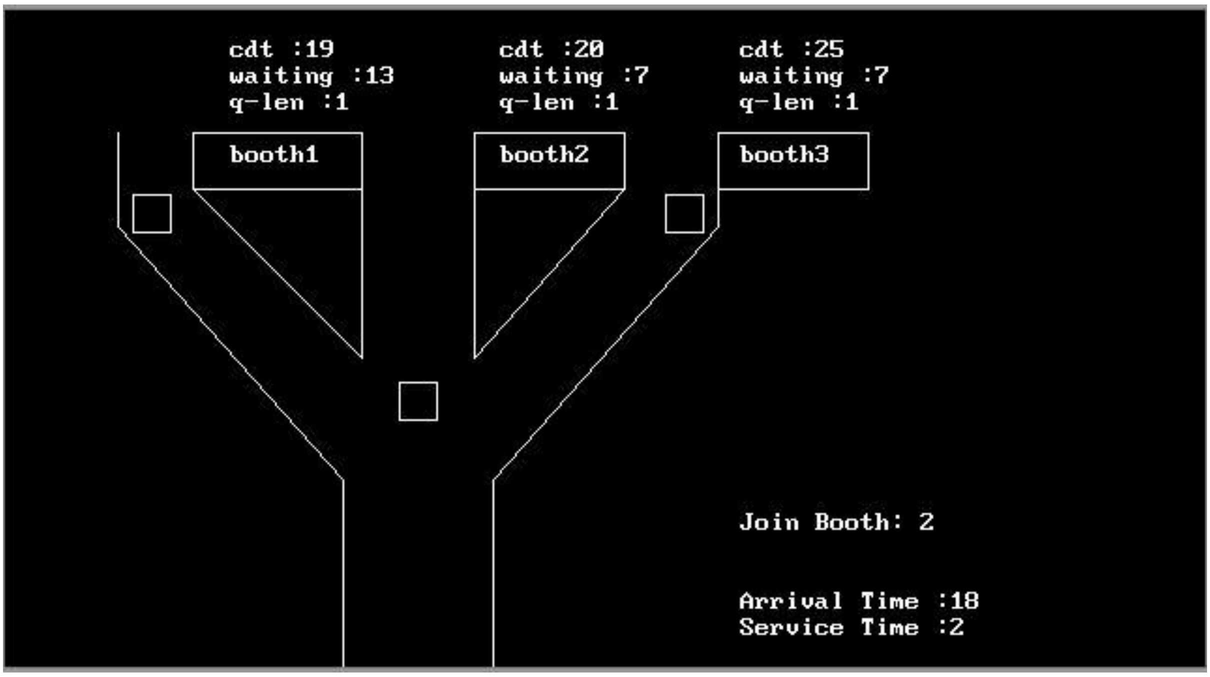
Developer info Screen



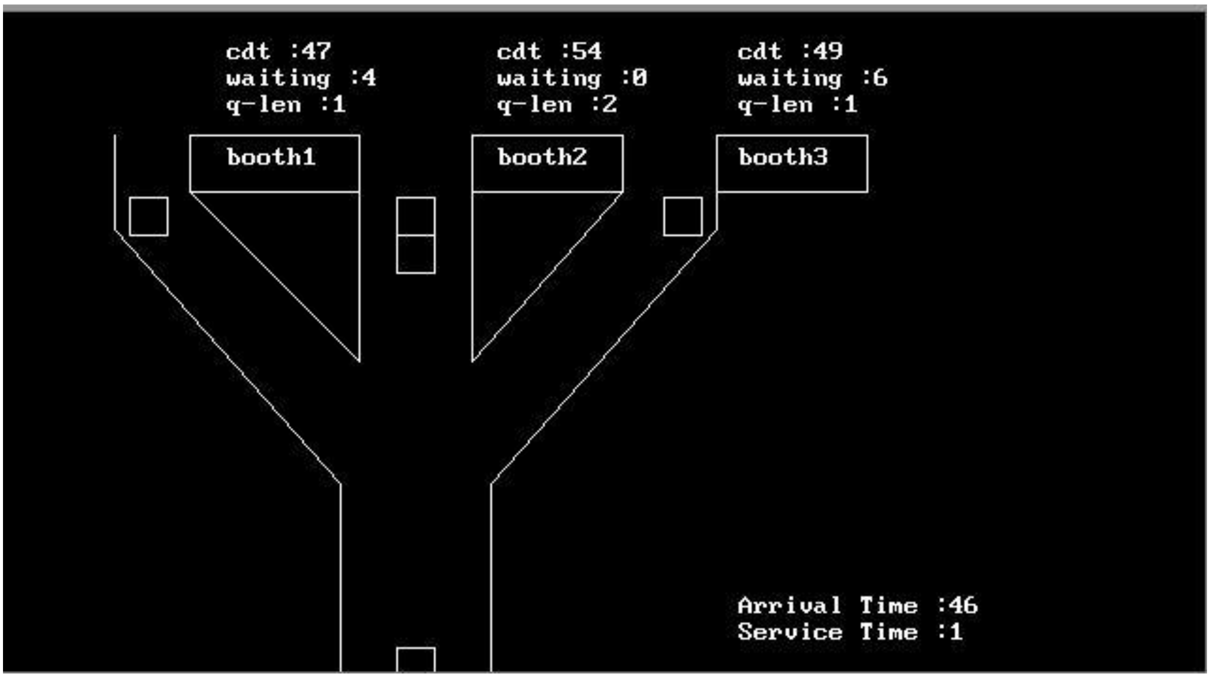
Simulation Screen shot -1



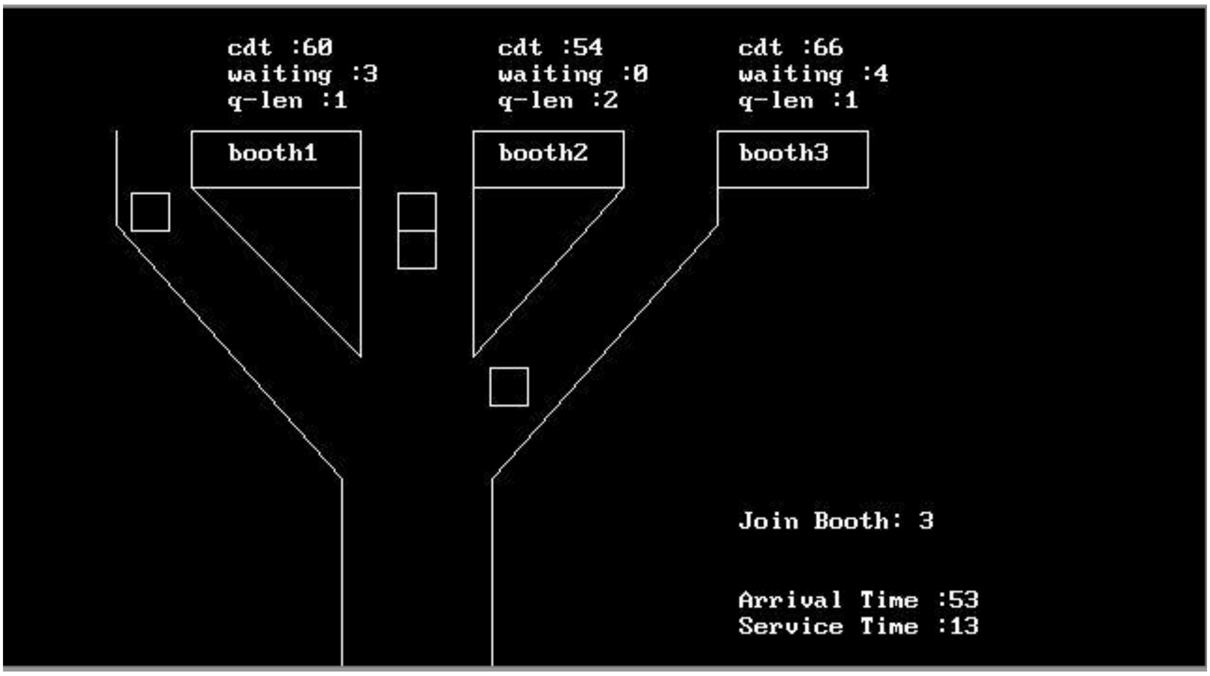
Simulation Screen Shot - 2



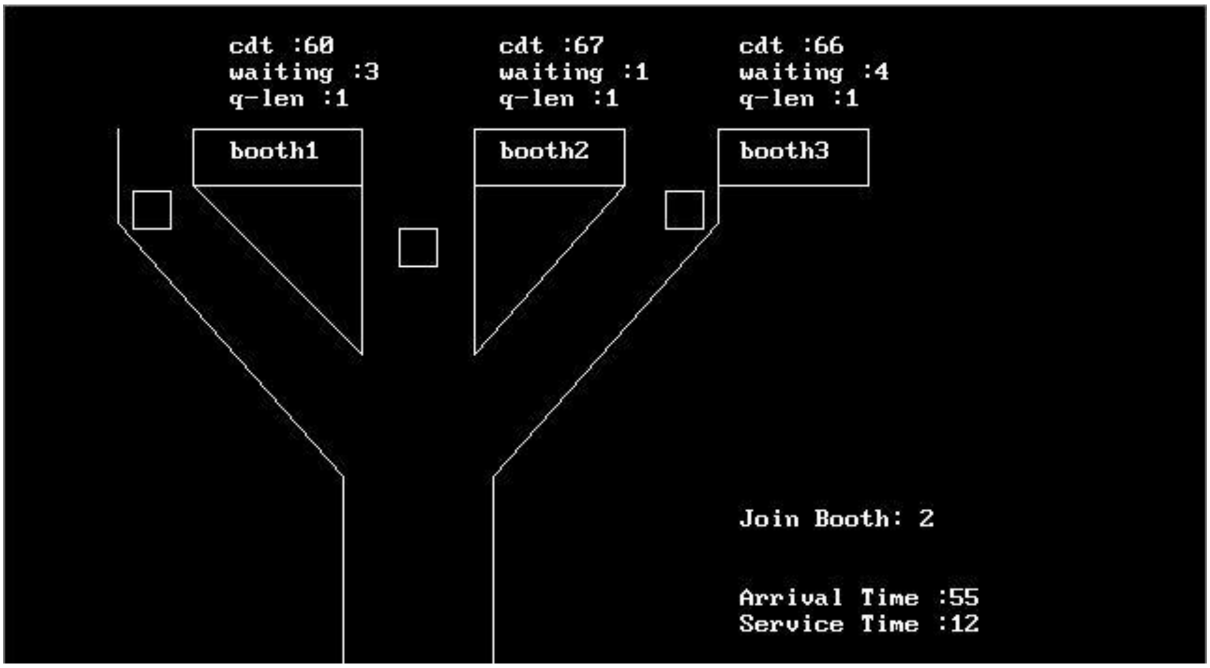
Simulation Screen Shot – 3



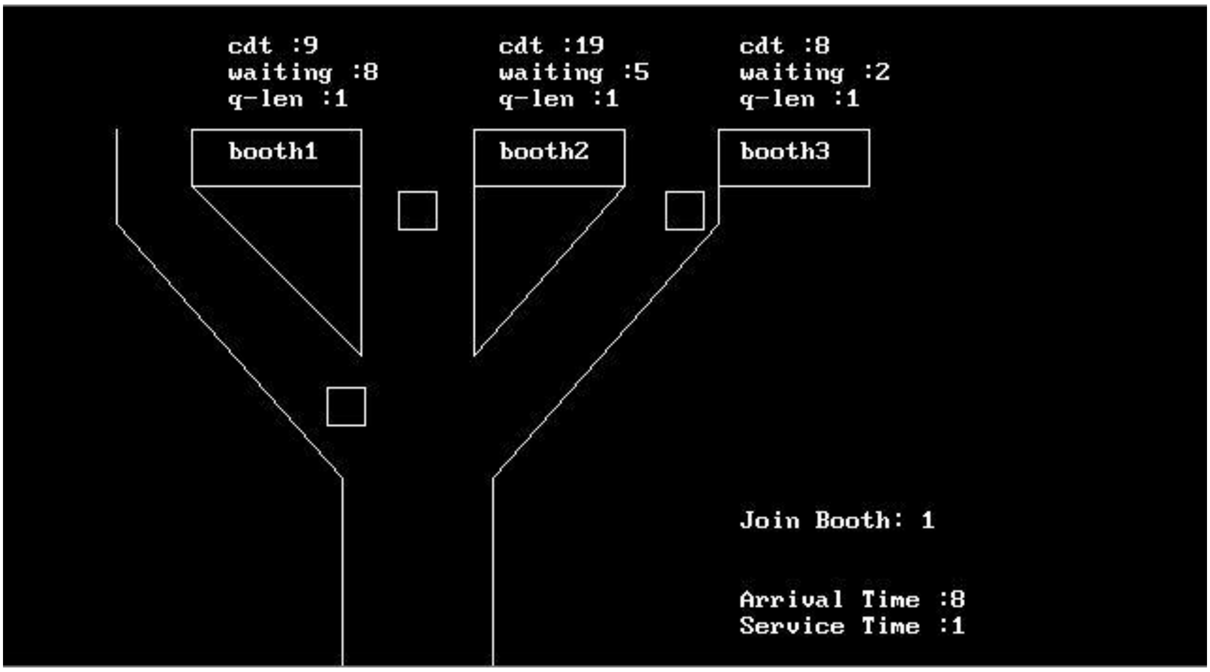
Simulation Screen Shot- 4



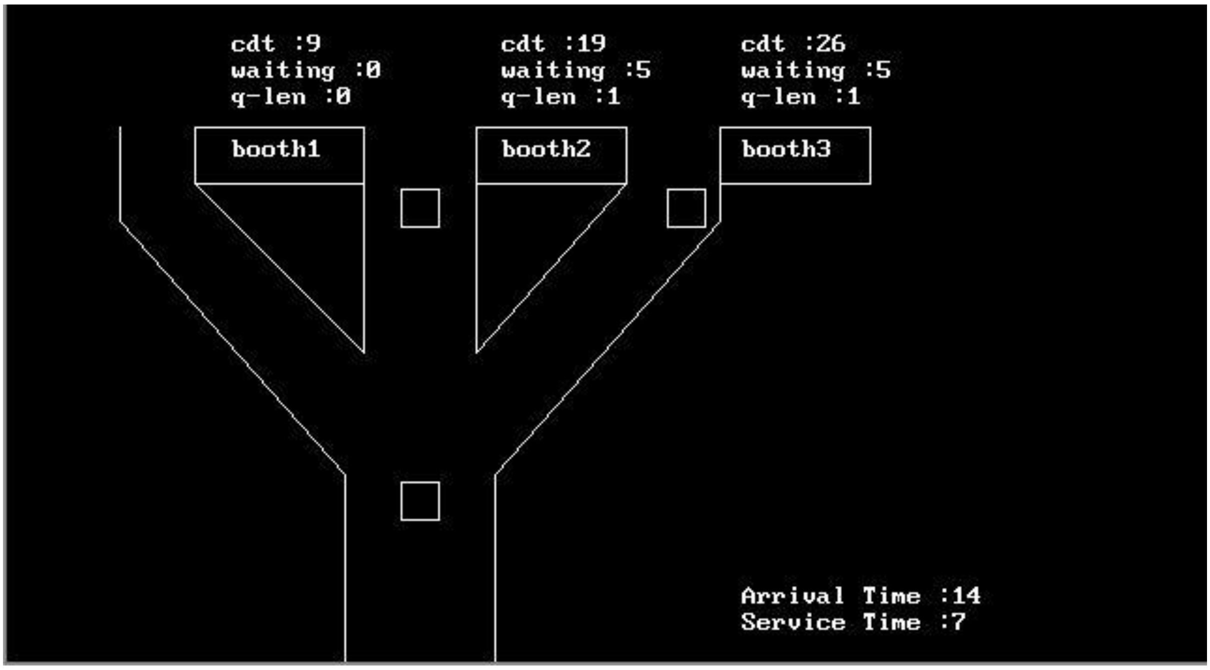
Simulation Screen Shot - 5



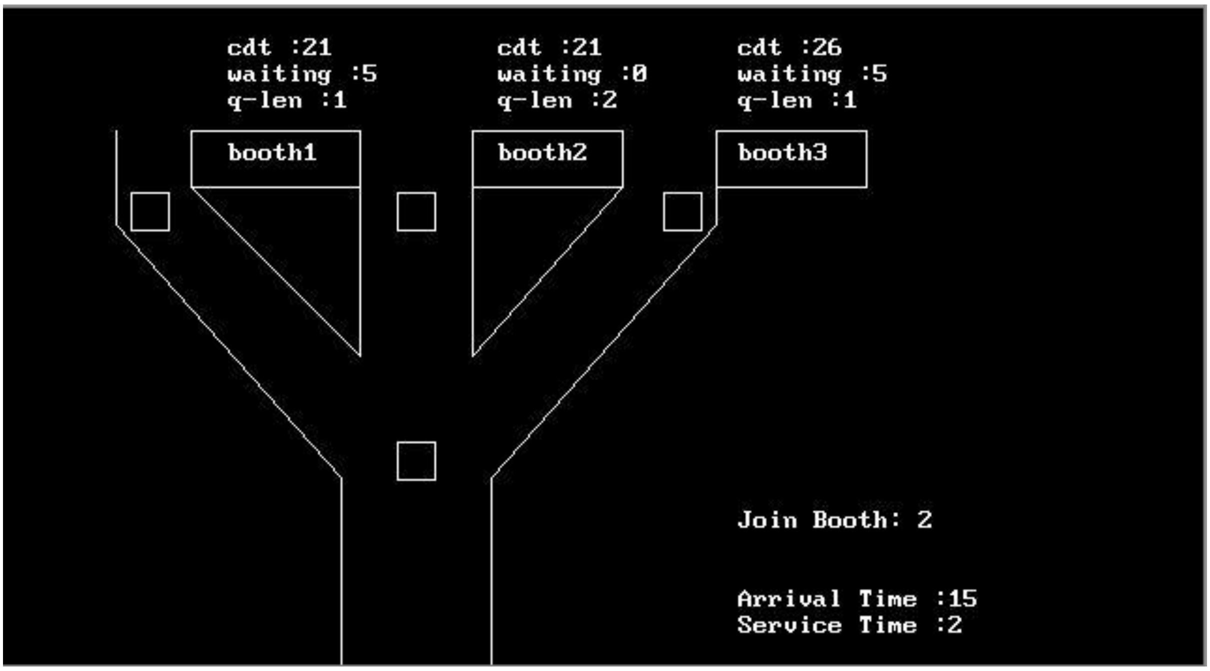
Simulation Screen Shot - 6



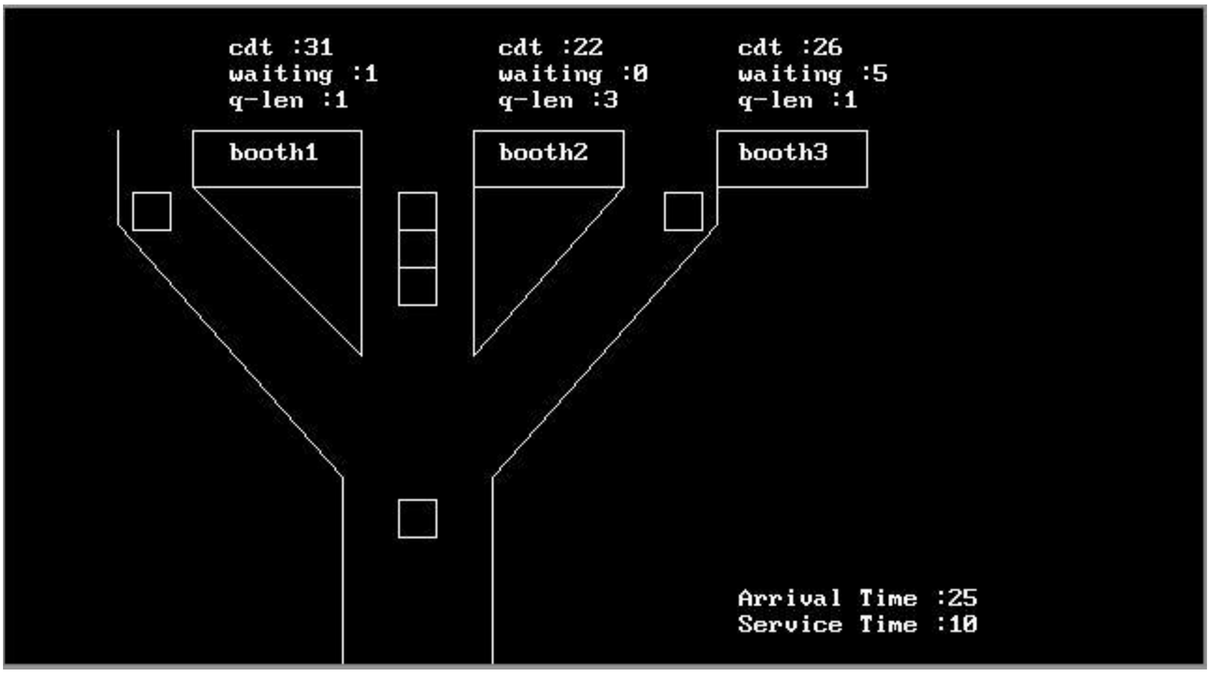
Simulation Screen Shot - 7



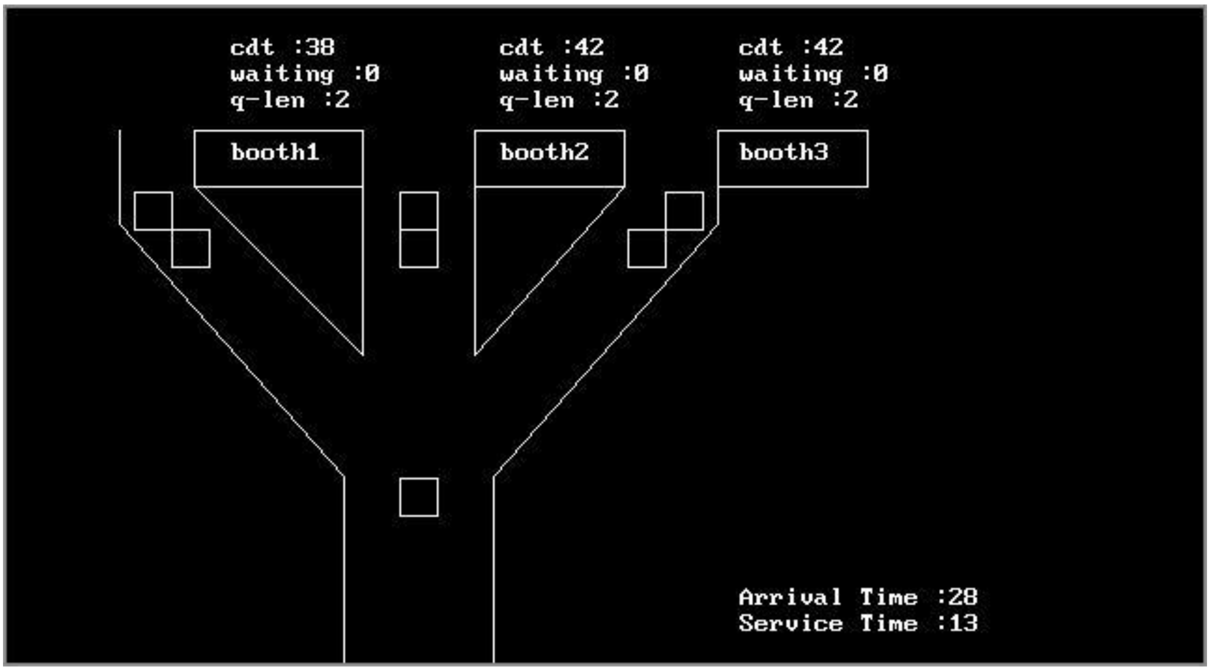
Simulation Screen Shot - 8



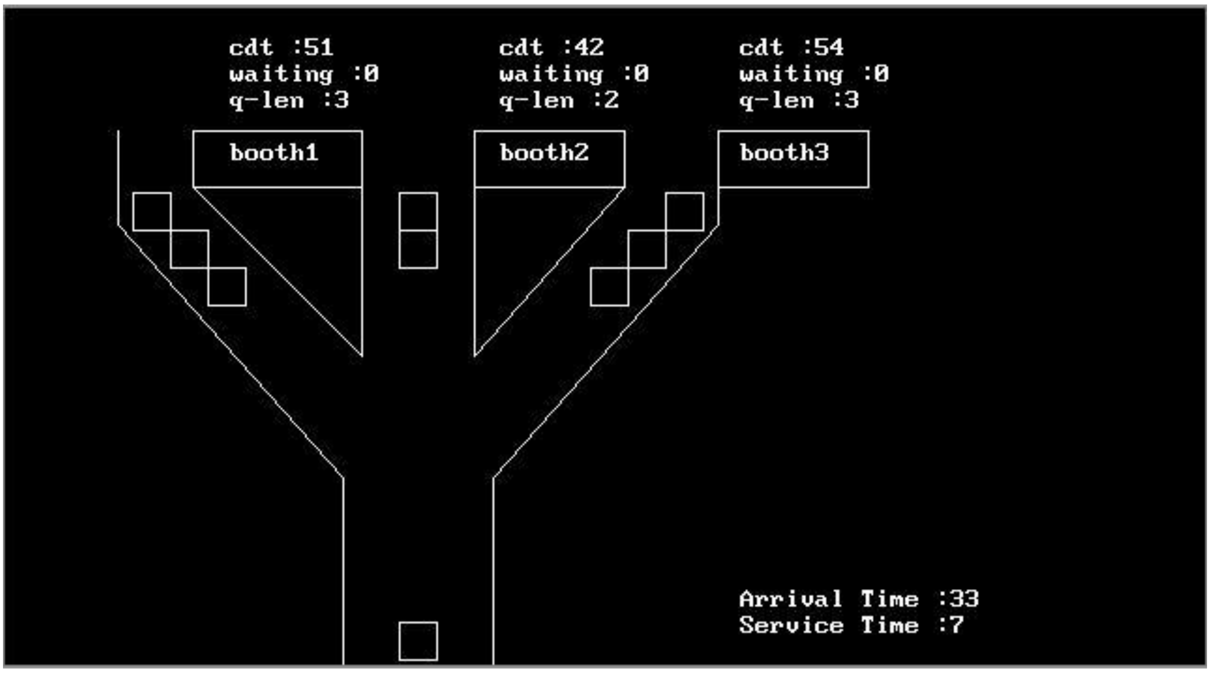
Simulation Screen Shot – 9



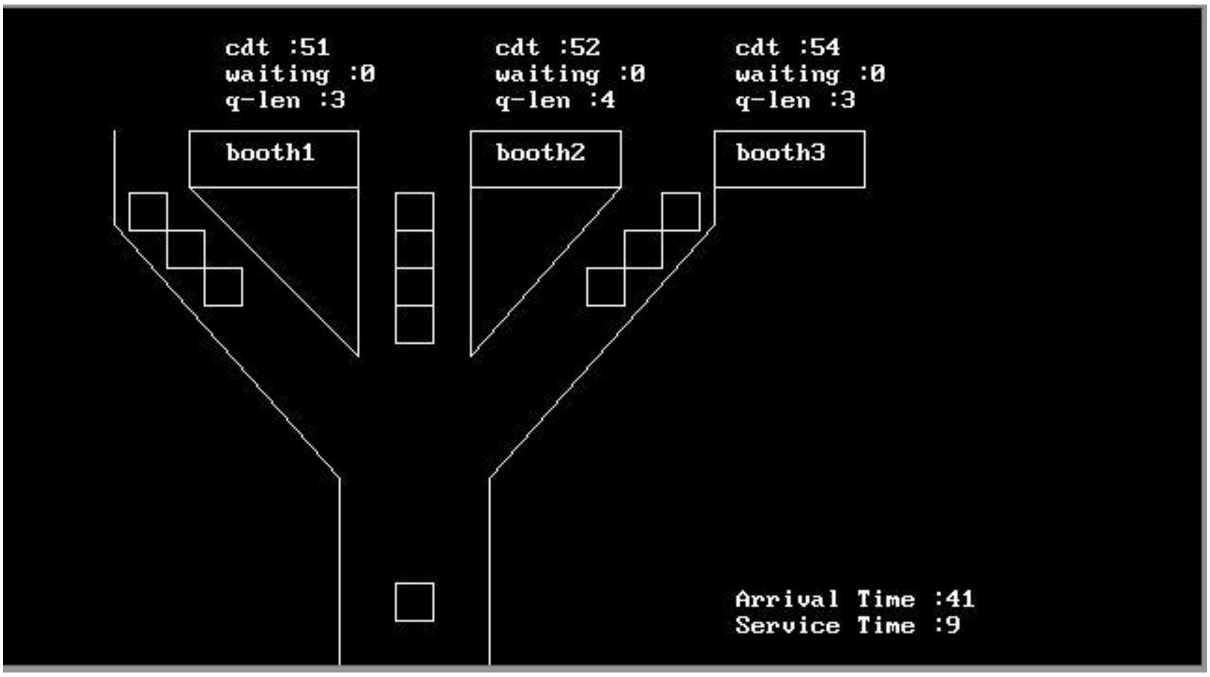
Simulation Screen Shot - 10



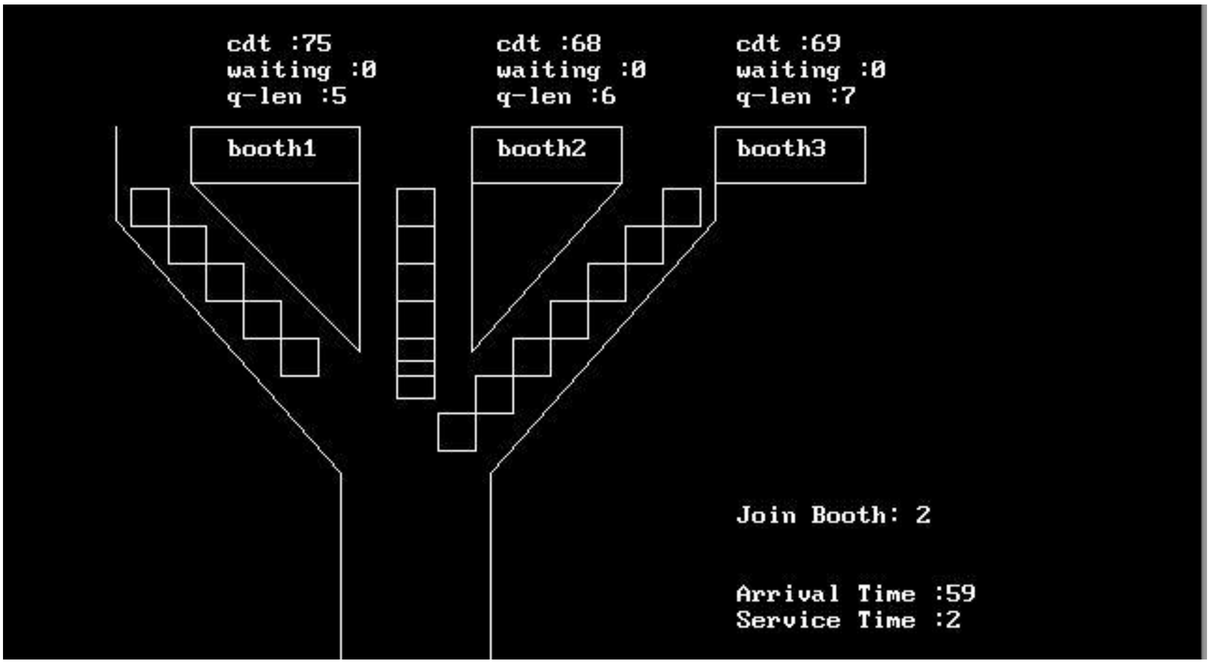
Simulation Screen Shot – 11



Simulation Screen Shot – 12



Simulation Screen Shot – 13



Simulation Screen Shot - 14

CHAPTER – 05

FUTURE SCOPE

The project have a future scope that if this simulation study shows some effectible and impressive result in decreasing the traffic at the toll plaza and will address a fair share of load amongst various toll booth at a toll plaza system. Then after this we can work by giving it some more similarity to the real world toll plaza system introducing different vehicle and lanes that are to be allocated to these different vehicles and categories based decision can also be implemented.

Moreover we can also bring some advance level technology use that can enhance our system in all together a new manner bringing the service time of the vehicle much lower for this we can implement the simulation of the payment system that which type of payment methodology (e-payment, card swiping, manual, RFID cards) can give a hike in the performance of the booth and will make the system fit.

We can study on some more traffic patterns to make the system more fruitful.

The estimated future traffic counts for the toll bridge are used to study the difference between the two traffic patterns.

CHAPTER – 06

CONCLUSION

In this system we study about the queue theory’s application by implementing it in the toll plaza system to optimize the queue length at the toll plaza system so that there can be fair share of load balancing at the toll plaza among the various toll booth.

This is done by keeping the track of the arrival pattern of the new entrance to the system and based upon the present system state it will move to particular server’s queue for which it is allotted.

Toll plaza performance measures such as average queue length, average waiting time, maximum queue length, and maximum waiting time at the tolls.

However there are still some of the features that can be included to making system much better and yes the system needs some more detailed study to be implemented as a real one.

CHAPTER – 07

REFERENCES

1. www.itk.ilstu.edu/faculty/chungli/IT179/doc%5C15S_IT179_Final_TeamProject.pdf
2. ntl.bts.gov/lib/10000/10600/10623/Toll_Plaza_Design.pdf
3. journals.sfu.ca/ijietap/index.php/ijie/article/viewFile/194/191
4. shodhganga.inflibnet.ac.in/bitstream/10603/7835/7/07_chapter%203.pdf
5. www.tutorialpoints.com
6. Simulation and Modeling by N.Deo