Green University of Bangladesh
Department of Computer Science and Engineering (CSE)
Faculty of Sciences and Engineering
Semester: (Spring, Year:2021), B.Sc. in CSE (Day)

Course Title: CSE 304 Microprocessors and Microcontrollers Lab
Course Code: Section: PC 193 - D

Lab Project Name: CAR Parking System.

Student Details

Name		ID
1.	Mohammad Nazmul Hossain	193902031
2.	Azmary Akter	193902019
3.	Fahim al Huq	193902022
4.	Jahid Hasan	193902001

Submission Date: 14/09/21
Course Teacher's Name: Md. Sultanul Islam Ovi

[For Teachers use only: Don't Write Anything inside this box]

Lab Project Status

Marks: …… Signature: ..... Comments:

..... Date: .....

Table of Contents

Chapter 1

Introduction

1.1 Introduction

The smart parking system implemented mainly in Europe, the United States, and Japan (Shaheen et al., 2005) are developed with the incorporation of advanced technologies and researches from various academic disciplines. With its deployment in the car park, it is hoped that it would solve the aforementioned problems faced by the patrons within the car park.

1.2 Design Goals/Objective

Parking management system, we aim to create a smart parking system that will not only calculate the total capacity for cars in the parking area but will also give services at a low cost where they can park their cars in an orderly fashion.

Chapter 2

Design/Development/Implementation of the Project

2.1 Section (Features of our project)

In our project, we are basically designing a parking system where one has to just enter the details of his vehicle (like a car, rickshaw, or bus) at the entry gate and then leave his/her vehicle there and the vehicle will automatically get parked at the available parking slot. If the parking area is full, a warning message of “Parking is full!” will be displayed. In this way, that person will be able to save time and energy.

2.2 Subsection (Code)

We have divided our project into two parts: the planning part, software implementation. First of all, we planned the layout of our project. Then we implemented our project on software. For the software implementation, we used Emu8086.

```
.model small
.stack 100h
.data
menu db '******************MENU**********************$'
menu1 db 'Press 1 for rikshw 200 BDT$'
menu2 db 'Press 2 for cars 300 BDT$'
menu3 db 'Press 3 for bus 400 BDT$'
menu4 db 'Press 4 to show the record$'
menu5 db 'Press 5 to delete the record$'
menu6 db 'Press 6 to exit$'
msg1 db 'Parking Is Full$'
msg2 db 'Wrong input$'
msg3 db 'car$'
msg4 db 'bus$'
msg5 db 'record$'
msg6 db 'there is more space$'
msg7 db 'the total amount is=$'
msg8 db 'the total numbers of vehicles parked=$'
msg9 db 'the total number of rikshws parked=$'
msg10 db 'the total number of cars parked=$'
msg11 db 'the total number of buses parked=$'
msg12 db '***Record deleted successfully***$'
amount dw 0
count dw  '0'
am1 dw ?
am2 dw ?
am3 dw ?
```


```
   ;userinput

mov ah,1
int 21h
mov bl,al

mov dx,10
mov ah,2
int 21h
mov dx,13
mov ah,2
int 21h

   ;now compare
mov al,bl,
cmp al,'1'
je rikshw
cmp al,'2'
je car
cmp al,'3'
je bus
cmp al,'4'
je rec
cmp al,'5'
je del
cmp al,'6'
je end_
```

```
rikshaw proc
cmp count,'8'
jle rikshw1
mov dx,offset msg1
mov ah,9
int 21h
jmp while_
jmp end_

rikshw1:
mov ax,200
add amount, ax
mov dx,0 ; remainder is 0
mov bx,10
mov cx,0
l2:
        div bx
        push dx
        mov dx,0
        mov ah,0
        inc cx
        cmp ax,0
        jne l2

l3:
        pop dx
        add dx,48
        mov ah,2
        int 21h
loop l3
;mov am1,dx
inc count
;mov dx,count
inc r

jmp while_
jmp end_
```

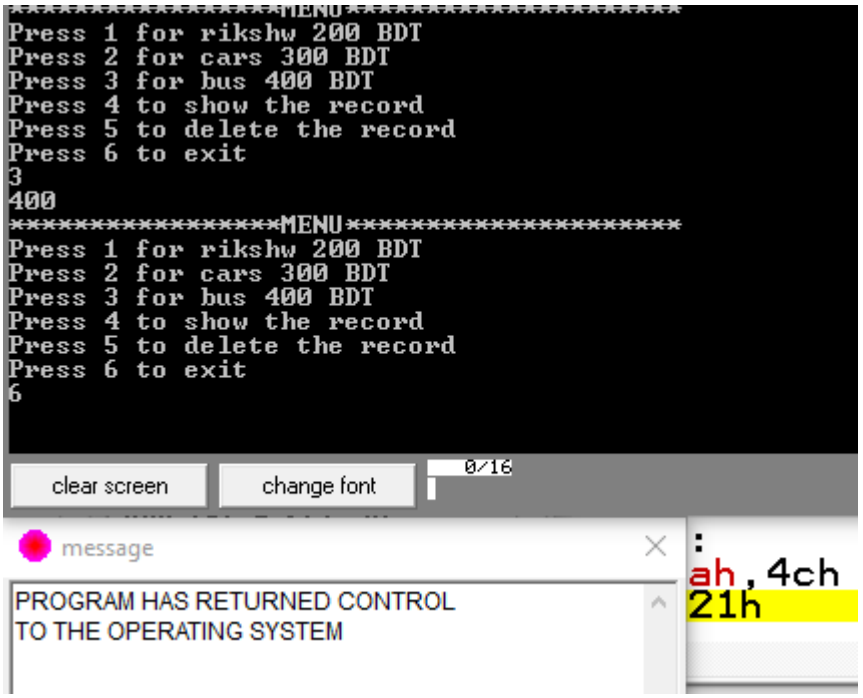
```
******************MENU**********************
Press 1 for rikshw 200 BDT
Press 2 for cars 300 BDT
Press 3 for bus 400 BDT
Press 4 to show the record
Press 5 to delete the record
Press 6 to exit
1
200
******************MENU**********************
Press 1 for rikshw 200 BDT
Press 2 for cars 300 BDT
Press 3 for bus 400 BDT
Press 4 to show the record
Press 5 to delete the record
Press 6 to exit
```

```
delt proc
mov r,'0'
mov c,'0'
mov b,'0'
mov amount,0
;sub amount,48
mov count,'0'
mov dx,offset msg12
mov ah,9
int 21h

mov dx,10
mov ah,2
int 21h
mov dx,13
mov ah,2
int 21h
```

```
Press 3 for bus 400 BDT
Press 4 to show the record
Press 5 to delete the record
Press 6 to exit
1
200
******************MENU**********************
Press 1 for rikshw 200 BDT
Press 2 for cars 300 BDT
Press 3 for bus 400 BDT
Press 4 to show the record
Press 5 to delete the record
Press 6 to exit
5
***Record deleted successfully***

******************MENU**********************
Press 1 for rikshw 200 BDT
Press 2 for cars 300 BDT
Press 3 for bus 400 BDT
Press 4 to show the record
Press 5 to delete the record
Press 6 to exit
```

```
; print here the whole amount
mov ax, amount

mov dx,0
mov bx,10
mov cx,0
totalpush:
          div bx
          push dx
          mov dx,0
        ;  mov ah,0
          inc cx
          cmp ax,0
         jne totalpush

totalprint:
          pop dx
          add dx,48
          mov ah,2
          int 21h
loop totalprint
```

```
Press 6 to exit
2
300
*****************MENU*********************
Press 1 for rikshw 200 BDT
Press 2 for cars 300 BDT
Press 3 for bus 400 BDT
Press 4 to show the record
Press 5 to delete the record
Press 6 to exit
4
the total amount is=1200
the total numbers of vehicles parked=4
the total number of rikshws parked=1
the total number of cars parked=2
the total number of buses parked=1
*****************MENU*********************
Press 1 for rikshw 200 BDT
Press 2 for cars 300 BDT
Press 3 for bus 400 BDT
Press 4 to show the record
Press 5 to delete the record
Press 6 to exit
```


**********MENU**********
Press 1 for rikshw 200 BDT
Press 2 for cars 300 BDT
Press 3 for bus 400 BDT
Press 4 to show the record
Press 5 to delete the record
Press 6 to exit
3
400
**********MENU**********
Press 1 for rikshw 200 BDT
Press 2 for cars 300 BDT
Press 3 for bus 400 BDT
Press 4 to show the record
Press 5 to delete the record
Press 6 to exit
6
clear screen
change font
0/16
message
PROGRAM HAS RETURNED CONTROL
TO THE OPERATING SYSTEM
:
ah,4ch
21h

Chapter 3

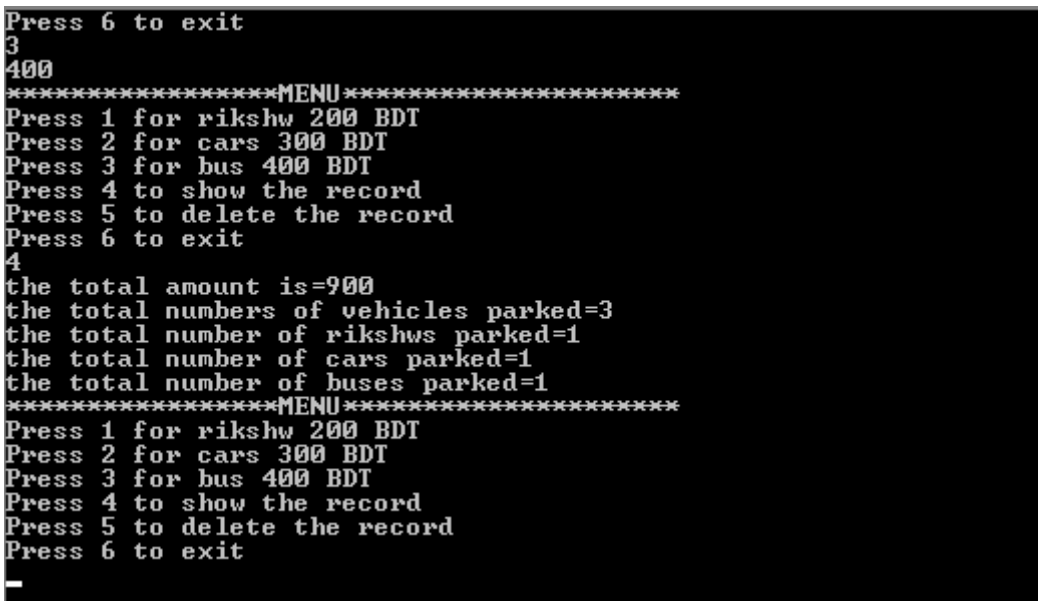
Performance Evaluation

3.1 Simulation Environment/ Simulation Procedure

1. The program of this project has been written in emu8086 software.
2. The project was written in assembly language x86
3. Our project was a small model and used a stack.
4. The operating system was Windows and we watched the output in the console of emu8086 Emulation software.

3.2 Results and Discussions

3.2.1 Results



3.2.2 Analysis and Outcome

- **We made this project considering the factors mentioned below :**
  The first step was the main screen. The main screen shows the welcoming Statement and then starts validating any information regarding the parking status based on the service list.

1. This is made for the end-user to interact with the vehicle parking system.
2. The user will be shown the menu and the user can select the menu he wants, this is convenient for the user, because, after selecting the menu the whole meu will be popped again in front of the end-user

Chapter 4

Conclusion

4.1 Introduction

The report clearly demonstrates the way the 3 level parking controller system was implemented. It was also highly beneficial as it gave us an insightful view of how advanced smart parking systems are programmed in order to be used in embedded systems later on. It is also quite important to mention that our code was optimized so it can hold fewer contexts , and avoid complex instruction.

4.2 Practical Implications

Consider these parking practices while you park:

1. While parking, make sure you park well inside the white lines.
2. Be mindful of other people's convenience.
3. Don't be a driveway blocker.
4. Avoid car parks and spaces reserved for certain groups – the elderly, physically challenged, or families.

4.3 Scope of Future Work

The future scope of the project:

1. This project can be used for the parking system in any shopping mall, multiplex.

2. It can be used for industries, commercial offices, and educational institutes.

3. This can be expanded in the sense of security.

4. We can add the pick and place facility to park the cars automatically. Automatic parking systems help reduce the land use for parking and maximize the efficiency of space usage.