# PROJECT REPORT

**Course Name:** **Computer Organization and Assembly Language**

**Project Name:** **Parking System**

**Group Members: Asharib Ahmed (CSC-18F-096)**
**Mahnoor Anwar (CSC-18F-067)**

## Introduction to Assembly Language:

This is a brief introduction to assembly language. Assembly language is the most basic programming language available for any processor. With assembly language, a programmer works only with operations implemented directly on the physical CPU. Assembly language lacks high-level conveniences such as variables and functions, and it is not portable between various families of processors. Nevertheless, assembly

language is the most powerful computer programming language available, and it gives programmers the insight required to write effective code in high-level languages. Learning assembly language is well worth the time and effort of every serious programmer.

## **Project Discription:**

Parking system is a concept to park the vehicles. In this system, we have set 6 menus. 1 is for "rikshaw", 2 is for "car", 3 is for "Bus", 3 is to show the record , 5 is to delete the record and 6 is to exit.  In record some strings will be shown.

- Total vehicles parked
- Count of rikshaw
- Count of cars
- Count of bus
- Total amount of the vehicles

In this project We have basically used

JMP

JE

JNE

INC

DEC

```
1  |
2  .model small
3  .stack 100h
4  .data
5  msg0 db 0ah,0dh,'  $'
6  msg02 db 0ah,0dh, '      $'
7  menu db '*****************MENU*********************$'
8  menu1 db 'Press 1 for rikshw$'
9  menu2 db 'Press 2 for cars$'
10 menu3 db 'Press 3 for bus$'
11 menu4 db 'Press 4 to show the record$'
12 menu5 db 'Press 5 to delete the record$'
13 menu6 db 'Press 6 to exit$'
14 msg1 db 'Parking Is Full$'
15 msg2 db 'Wrong input$'
16 msg3 db 'car$'
17 msg4 db 'bus$'
18 msg5 db 'record$'
19 msg6 db 'there is more space$'
20 msg7 db 'the total amount is=$'
21 msg8 db 'the total numbers of vehicles parked=$'
22 msg9 db 'the total number of rikshws parked=$'
23 msg10 db 'the total number of cars parked=$'
24 msg11 db 'the total number of buses parked=$'
25 msg12 db '***Record deleted successfully***$'
26 amount dw 0
27 count dw  '0'
28 am1 dw ?
29 am2 dw ?
30 am3 dw ?
31
32
33 r dw '0'
34 c db '0'
35 b db '0'
```

we have set 20 variables from menu to menu 6, msg1 to msg12, amount and count for printing our desired strings and also we have used r,c,w to make their initially to 0

```
.code
main proc

mov ah,0 ; screen mode
mov al,14 ; number of vga
int 10h

mov ah,0bh
mov bh,00h ;background color
mov bl,12 ; color pixel
int 10h

mov bh,1 ; palette
mov bl,1 ; no palette
int 10h
```

Here we have used the little graphics in which we have changed the background color of the screen.

```asm
mov dx,offset menu
mov ah,9
int 21h

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


mov dx,offset menu1
mov ah,9
int 21h

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


mov dx,offset menu2
mov ah,9
int 21h

mov dx,10
mov ah,2
int 21h
mov dx,13
mov ah,2
int 21h
```
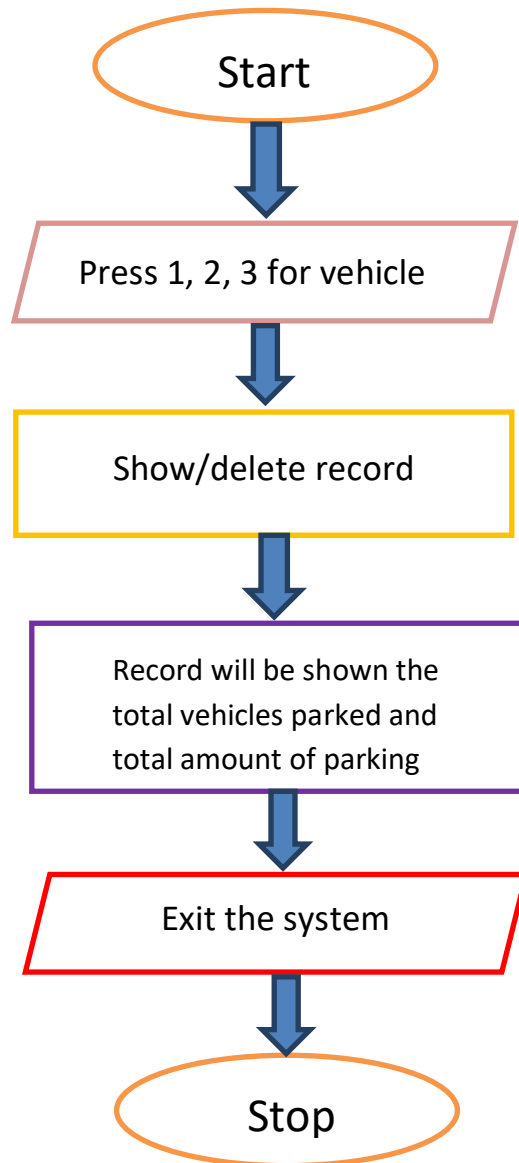
Here 6 menus will be print to press number for vehicles and to show the record and to delete the record.

```asm
  ;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_
```

Now user will input the desired number and it will compare either the number is correct or not.

# FLOW CHART :

Start

Press 1, 2, 3 for vehicle

Show/delete record

Record will be shown the
total vehicles parked and
total amount of parking

Exit the system

Stop

**CONCLUSION :**

This is really an interesting parking system. Anyone can park desired vehicle and multiple vehicles. We also gather a huge knowledge about Assembly Language by completing this project. We also hope that in future we can make more project in Assembly Language.

# -: <u>THANK YOU</u> :-