****

**SRN: PES1UG20CS415**

**Name: SHRUJAN**

**Section: G**

**Date: 28/9/2021**

**Q1:** **A parking garage contains a single lane and hold 8 cars. Garage has only 1 one entry and one exit. If a person arrives and takes his/her car that is not nearest to exit, all cars blocking its path are moved out then the person’s car is driven out and other cars are restored in same order as thery were originally.WAP that process a group of input Each input line contains an A – arrival & D- Departure and a lisence plate no. Cars are assumed to arrive and depart in the order specified by the input. The program should print a mesage when car whenever car arrives / departs.When**

**car arrives, the message should specify whether / not there is a room for the car in the garage. If there is no room the car leaves without entering garage. When a car departs, the message should include the no of times that the car was**

**moved out of the garage to allow other cars to depart.**

**Code:**

#*include*<stdio.h>

#*include*<stdlib.h>

typedef struct car{

    int numbPlate;

    int moved;

}Car;

void *depart*(Car \*garage,int \*top,int checkPlate);

void *arrival*(Car \*garage,int \*top,int size,int lPlate);

void *display*(Car \*garage,int top);

int *main*(){

    Car \*garage;

    int top=-1;

    int lPlate,checkPlate;

    int size=8;

    char ch;

    garage=(Car \*)*malloc*(size\*sizeof(Car));

*while*(1)

    {

*display*(garage,top);

*fflush*(*stdin*);

*printf*("\nA.Arrive\nD.Departure\nQ.Exit\nEnter your Choice :");

*scanf*("%c",&ch);

*switch*(ch){

*case* 'A':*if*(top ==size-1)

                    {

*printf*("\nGarage is Full \n");

*break*;

                    }

*else*{

*printf*("\nEnter The license plate number ");

*scanf*("%d",&lPlate);

*fflush*(*stdin*);

*arrival*(garage,&top,size,lPlate);

                    }

*break*;

*case* 'D':*printf*("\nEnter the Licence Plate Number");

*scanf*("%d",&checkPlate);

*depart*(garage,&top,checkPlate);

*break*;

*case* 'Q':*exit*(0);

        }

    }

}

void *arrival*(Car \*garage,int \*top,int size,int lPlate)

{

    (\*top)++;

    garage[\*top].numbPlate=lPlate;

    garage[\*top].moved=0;

}

void *depart*(Car \*garage,int \*top,int checkPlate)

{

    int i=\*top;

    int t=-1;

    Car temp[8];

*while*(i>=0 && (checkPlate != garage[i].numbPlate))

    {

        i--;

    }

*while*(\*top>i)

            {

                t++;

                temp[t].numbPlate=garage[\*top].numbPlate;

                (garage[\*top].moved)++;

                temp[t].moved=garage[\*top].moved;

                (\*top)--;

            }

*printf*("\nCar %d Removed It was Moved %d times\n",garage[\*top].numbPlate,garage[\*top].moved);

*while*(t >= 0)

            {

                garage[\*top].numbPlate=temp[t].numbPlate;

                garage[\*top].moved=temp[t].moved;

                (\*top)++;

                t--;

            }

            (\*top)--;

}

void *display*(Car \*garage,int top)

{

    int t=top;

*while*(t>=0)

    {

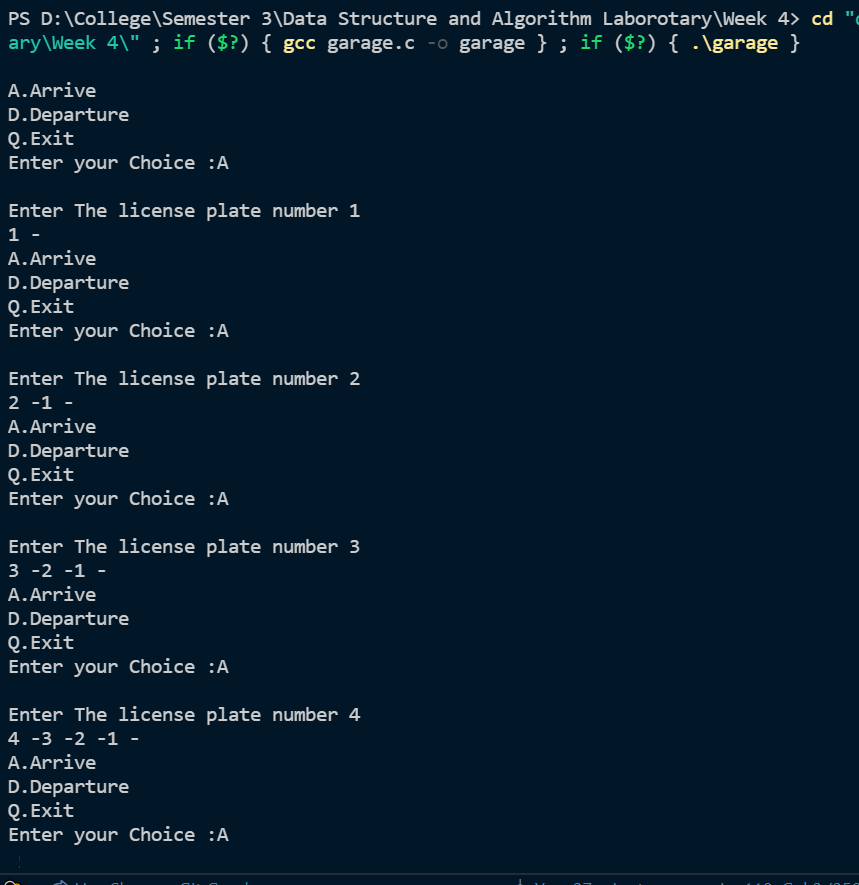
*printf*("%d -",garage[t].numbPlate);

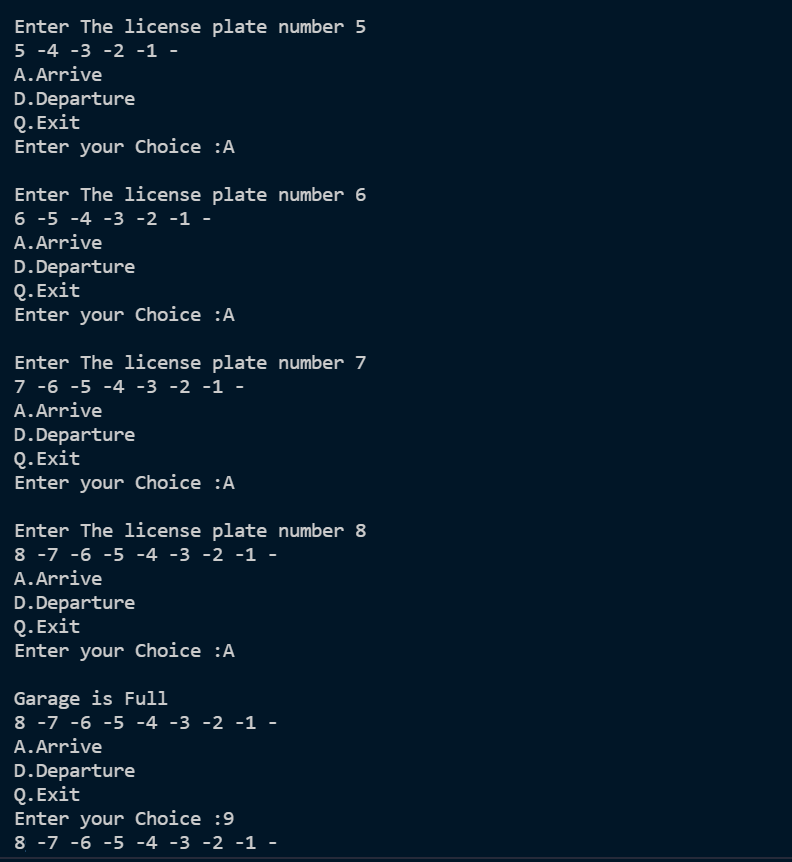
        t--;

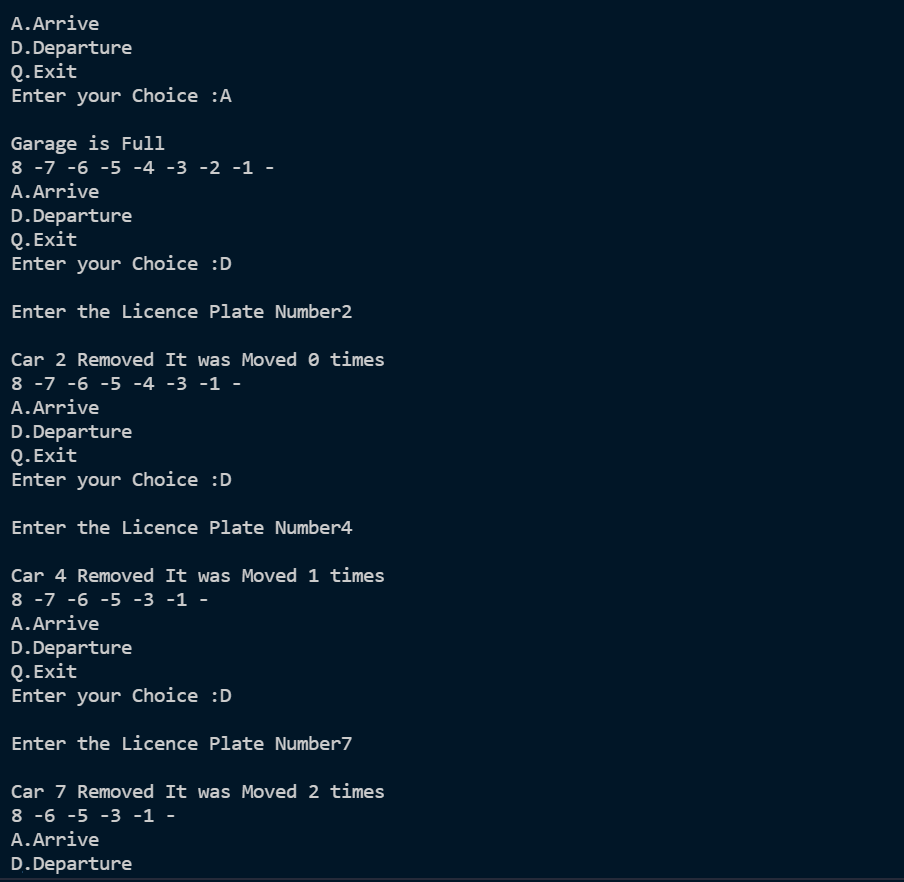
    }

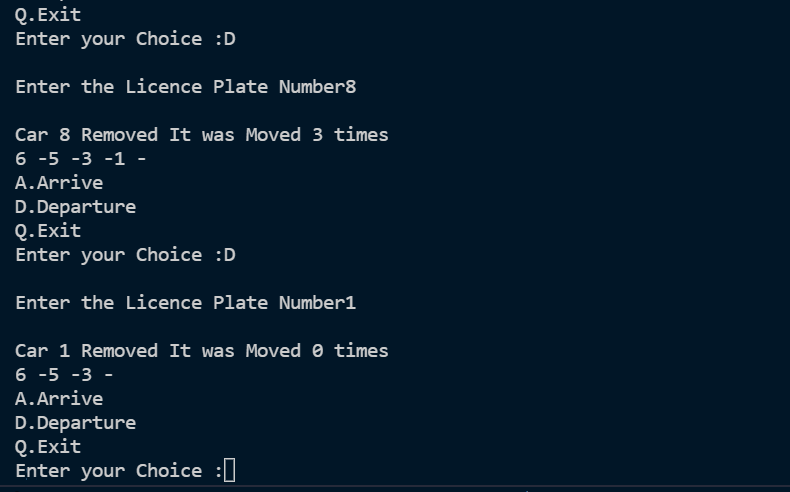
}

**Output:**

****

****

****

****