**Data Structures and its Applications**

**Q)** **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 they were originally. WAP that process a group of inputs Each input line contains an A – arrival & D- Departure and a licence plate no. Cars are assumed to arrive and depart in the order specified by the input. The program should print a message 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.**

**A)**

**Code:**

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

typedef struct garage

{

char plat[15];

int count;

struct garage \*link;

}node;

typedef struct manager

{

int present;

node \*exit;

node \*link;

}head;

head \*start;

void arival();

void depart();

void display();

void initial()

{

head \*temp=(head \*)malloc(sizeof(head));

start=temp;

start->link=NULL;

start->present=0;

}

void main()

{

initial();

int choice;

do

{

printf("\nEnter Your choice: \n1.New Car Arival \n2.Car Departure\n3.Cars present in the Garage\n4.Exit\n");

scanf("%d",&choice);

switch (choice)

{

case 1 : arival(); break;

case 2 : depart();break;

case 3 : display();break;

case 4 : exit(0);break;

default: printf("Enter a valid response");break;

}

} while (1);

}

void arival()

{

if(start->present>=8)

printf("\nThe garage is already full Kindly wait or you can leave\n");

else

{

node \*temp =(node \*)malloc(sizeof(node));

if(temp==NULL)

printf("\n===Memory OverFlow===\n");

else

{

printf("Enter car Plate No. :- ");

scanf("%s",temp->plat);

temp->count=0;

temp->link=NULL;

if(start->present==0)

{

start->link=temp;

start->exit=temp;

start->present+=1;

}

else

{

temp->link=start->link;

start->link=temp;

start->present+=1;

}

}

}

}

void depart()

{

if(start->present==0)

printf("The garage is empty.");

else

{

char m[15];

printf("Enter the Car plate No.:- ");

scanf("%s",m);

node \*trav=start->link;

while(trav!=NULL && (strcmp(trav->plat,m)!=0))

{

trav=trav->link;

}

if(trav==NULL)

printf("No Such car is present in the garage please rech out to the security for furthur information.");

else

{

node \*adder=trav->link;

while(adder!=NULL)

{

adder->count+=1;

adder=adder->link;

}

node \*prev=start->link;

while(prev->link!=trav && prev!=trav)

{

prev=prev->link;

}

if(start->link==start->exit)

{

start->link=NULL;

start->exit=NULL;

}

else if(trav==start->exit)

start->exit=prev;

if(trav!=NULL && trav!=start->link)

prev->link=trav->link;

else if(trav==start->link)

start->link=trav->link;

else

prev->link=NULL;

trav->link=NULL;

start->present-=1;

printf("Your car was taken out of the garage %d times inorder to give way to the other cars.\n",trav->count);

free(trav);

}

}

}

void display()

{

if(start->present==0)

printf("The garuage is Empty.");

else

{

node \*trav=start->link;

while(trav!=NULL)

{

printf("\ncar %s taken out %d times",trav->plat,trav->count);

trav=trav->link;

}

printf("\n");

}

}

**Output:**









