TITLE: PARKING MANAGEMENT

A MINI PROJECT REPORT

Submitted by

DHANUSH KARTHIK K S . (Register No: 20TUCS034)

HAMSANAND K V. (Register No:20TUCS043)

HARISH B. (Register No: 20TUCS050)

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

(Accredited by NBA)

SRI KRISHNA COLLEGE OF TECHNOLOGY

An Autonomous Institution | Accredited by NAAC with 'A' Grade

Affiliated to Anna University | Approved by AICTE

KOVAIPUDUR, COIMBATORE 641042

NOVEMBER 2021

TITLE

1. **DESCRIPTION OF THE PROJECT**

2. **LIST OF MODULES**

3. **CODING**

4. **SCREEN SHOTS**

PARKING MANAGEMENT

DESCRIPTION OF THE PROJECT:

This project is developed to make the parking easier in metropolitan cities with the help of computer application. This project is developed using linked list. In this project it will get details from the user to park the car in the empty slot and generates a receipt with token number. In order to take your car from the parking area you have to just enter your token number, it will de-allocate your booked slot.

MODULES:

DOUBLY LINKED LIST:

It is convenient to traverse lists backwards. Add an extra field to the data structure, containing a pointer to the previous cell.

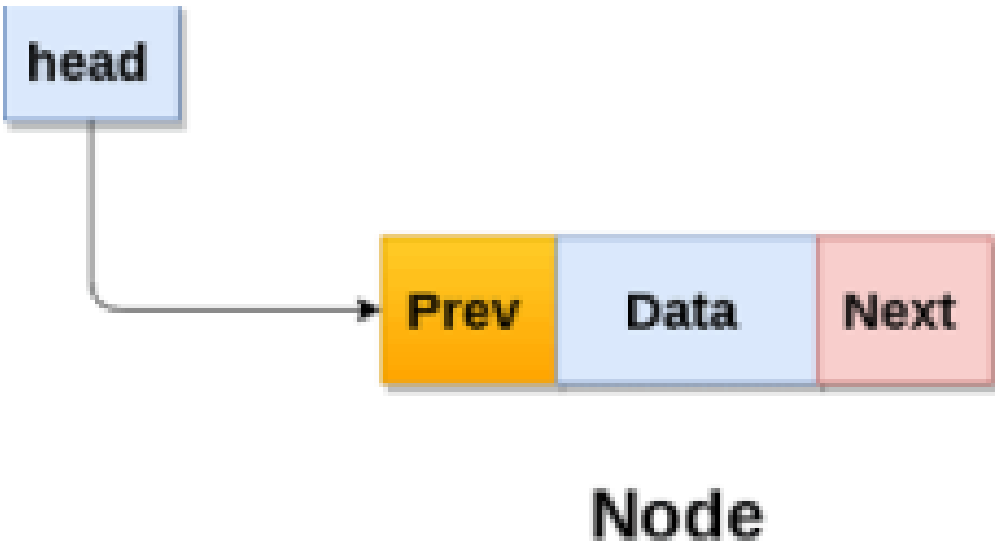
The cost of this is an extra link, adds to the space requirement and also doubles the cost of insertions and deletions because there are more pointers to fix.

A doubly linked list is a concrete data structure consisting of a sequence of nodes

1) Each node stores
2) Data (Element)

Pointers (Prev and Next) - Link to the previous and next node

The Pointers store the address of the previous and next node.



Node

STRUCTURE:

Structure is a user-defined data type in C language which allows us to combine data of different types together. Structure helps to construct a complex data type which is more meaningful. It is somewhat similar to an Array, but an array holds data of similar type only. In structure, data is stored in form of records.

In our project we have used structure to store data from the user and to create doubly linked list.

```
struct parking{
   char name[15],brand[12],type[10],no_plate[15];
   int no;
}pk[50];
```

char[15]	char[12]	char[10]	char[15]	int
name	brand	type	no_plate	no

CODING:

```
#include <stdio.h>
#include<stdlib.h>
#include<string.h>
int count=0,row=0,column=0,user = 0,to;
int token[5][10];
struct parking{
    char name[15],brand[12],type[10],no_plate[15];
    int no;
}pk[50];
struct Node{
    int data;
    struct Node* next;
    struct Node* down;
}*head = NULL,*newNode,*temp,*prevNode;
void newnode( int data){
    if(count >=10){
        row++;
    }
    newNode = (struct Node*)malloc(sizeof(struct Node));
    newNode->data = data;
```

```
    newNode->next = NULL;
    newNode->down = NULL;
    if(head == NULL){
        head = temp = prevNode = newNode ;
    }else{
        temp->next = newNode;
        temp = temp->next;
        if(row >= 1){
            temp->down = prevNode;
            prevNode = prevNode->next;
        }
    }
    count++;
}
void park(struct parking pk[]){
    temp = head;
    row = column = count = 0;
    while(temp!=NULL){
        if(count%10 == 0 && count != 0){
            column  = 0;
            row++;
```

```
        }
        if(temp->data == 0){
            int cho;
            printf("\n\nTHERE IS AN EMPTY SLOT IN ROW: %d
COLUMN: %d.\nDO YOU WANT TO
PARK?(YES(1)/NO(0)):",row+1,column+1);
            scanf("%d",&cho);
            if(cho == 1){
                system("clear");
                printf("YOUR VEHICLE HAS BEEN PARKED IN ROW:
%d COLUMN: %d.\n",row+1,column+1);
                pk[user].no = token[row][column];
                user++;
                temp->data = 1;
                break;
            }else{


            }
        }
        column++;
        count++;
        temp = temp->next;
```

```
    }
}
void takeaway(){
    temp = head;
    for(int i = 0;i<5;i++){
        for(int j = 0;j<10;j++){
            if(token[i][j]==to){
                temp->data = 0;
                printf("\nYOUR CAR IS IN ROW: %d COLUMN:
%d.",row+1,column+1);
                break;
            }
            temp=temp->next;
        }
    }
}
void display(){
    printf("PARKING VIEW :  \n\n|0| - REPRESENTS EMPTY SLOT
\n|1| - REPRESENTS OCCUPIED\n");
    temp = head;
    int n =0;
```

```
    printf("\n-------------------------------------------------------------------
---\n");
    while(temp!=NULL){
        if(n%10 == 0 && n!=0){
            printf("|end|");
            printf(" \n |      |      |      |      |      |      |      |      |\n");
            printf(" V      V      V      V      V      V      V      V      V
V\n");
        }
        printf("|%d| -> ",temp->data);
        temp = temp->next;
        n++;
    }
    printf("|end|");
    printf("\n-------------------------------------------------------------------
---\n");
}
void getdata(struct parking pk[]){
    printf("Enter the details: \n");
    printf("NAME: ");
    scanf("%s",pk[user].name);
    printf("BRAND: ");
```

```
    scanf("%s",pk[user].brand);
    printf("TYPE(CAR □ / BIKE □ ): ");
    scanf("%s",pk[user].type);
    printf("NUMBER PLATE: ");
    scanf("%s",pk[user].no_plate);
}
void displaydetails(struct parking pk[],int to){
    for(int i = 0;i<50;i++){
        if(to == pk[i].no){
            printf("\nDetails of the car: \n");
            printf("NAME: %s",pk[i].name);
            printf("\nBRAND: %s",pk[i].brand);
            printf("\nTYPE: %s",pk[i].type);
            printf("\nNUMBER PLATE: %s\n",pk[i].no_plate);
        }
    }
}
void receipt(struct parking pk[]){
    int a=user-1;
    printf("--------------------------------\n");
    printf("        RECEIPT OF PARKING        ");
```

```
    printf("\nDetails of the car: \n");
    printf("NAME: %s",pk[a].name);
    printf("\nBRAND: %s",pk[a].brand);
    printf("\nTYPE: %s",pk[a].type);
    printf("\nNUMBER PLATE: %s\n",pk[a].no_plate);
    printf ("\nTOKEN NUMBER : %d",token[row][column]);
    if(strcmp(pk[a].type,"CAR")==0 || strcmp(pk[a].type,"car")==0 ||
strcmp(pk[a].type,"Car")==0){
        printf("\nPARKING CHARGE : RS.50");
    }else if(strcmp(pk[a].type,"BIKE")==0 ||
strcmp(pk[a].type,"bike")==0 || strcmp(pk[a].type,"Bike")==0){
        printf("\nPARKING CHARGE : RS.25");
    }else{
        printf("\nPARKING CHARGE : RS.100");
    }
    printf("\n-------------------------------\n");
}
int main()
{
    int ch=1,temp,t=2;
    for(int i = 0;i<50;i++){
        if(i==0)
```

```
            newnode(1);
        else if(i%3 == 0 || i%4 == 0 && i%2 == 0)
            newnode(0);
        else
            newnode(1);
    }
    for(int i = 0; i<5;i++){
        for(int j=0;j<10;j++){
            token[i][j]=951+((i*10)+j);
        }
    }
    printf("----WELCOME TO PARKING AREA----\n");
    while(ch!=0){
        printf("DO  YOU WANT TO PARK OR DRIVE
AWAY?\nPARK(1)\nDRIVE AWAY(2)\n");
        scanf("%d",&ch);
        system("clear");
        if(ch==1){
            char d;
            getdata(pk);
            system("clear");
            display();
```

```
        park(pk);
        display();
        printf("Enter any value to generate your recipt: ");
        scanf("%s",&d);
        system("clear");
        receipt(pk);
        printf("Enter 1 to exit.");
        scanf("%d",&temp);
        system("clear");
        ch = 1;
    }if(ch == 2){
        t=2;
        while(t==2){
        system("clear");
        printf("ENTER YOUR TOKEN NUMBER: ");
        scanf("%d",&to);
        displaydetails(pk,to);
        printf("IS THIS YOUR DETAILS?(YES(1)/NO(0)): ");
        scanf("%d",&t);
        system("clear");
        if(t==1){
```

```
            takeaway();
            display();
            printf("\nTHANKS FOR CHOOSING OUR PARKING \n
VISIT AGAIN ");
            printf("\nEnter 1 to exit.");
            scanf("%d",&temp);
            system("clear");
        }else{
            t=2;
        }
        }
    }
    return 0;
}
```

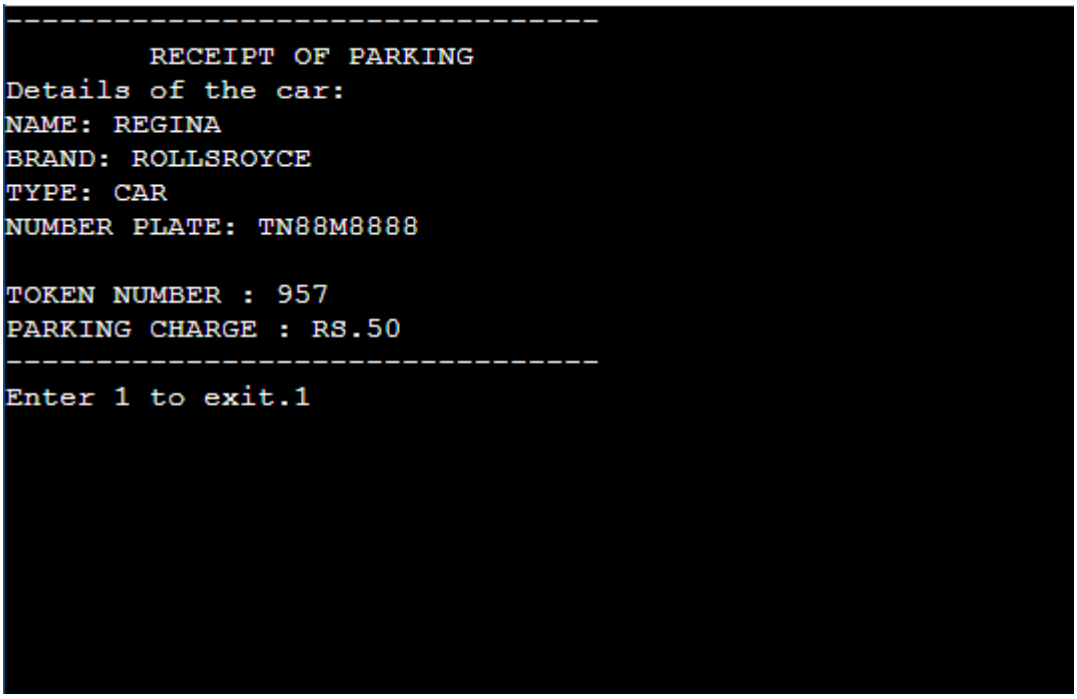
OUTPUT:

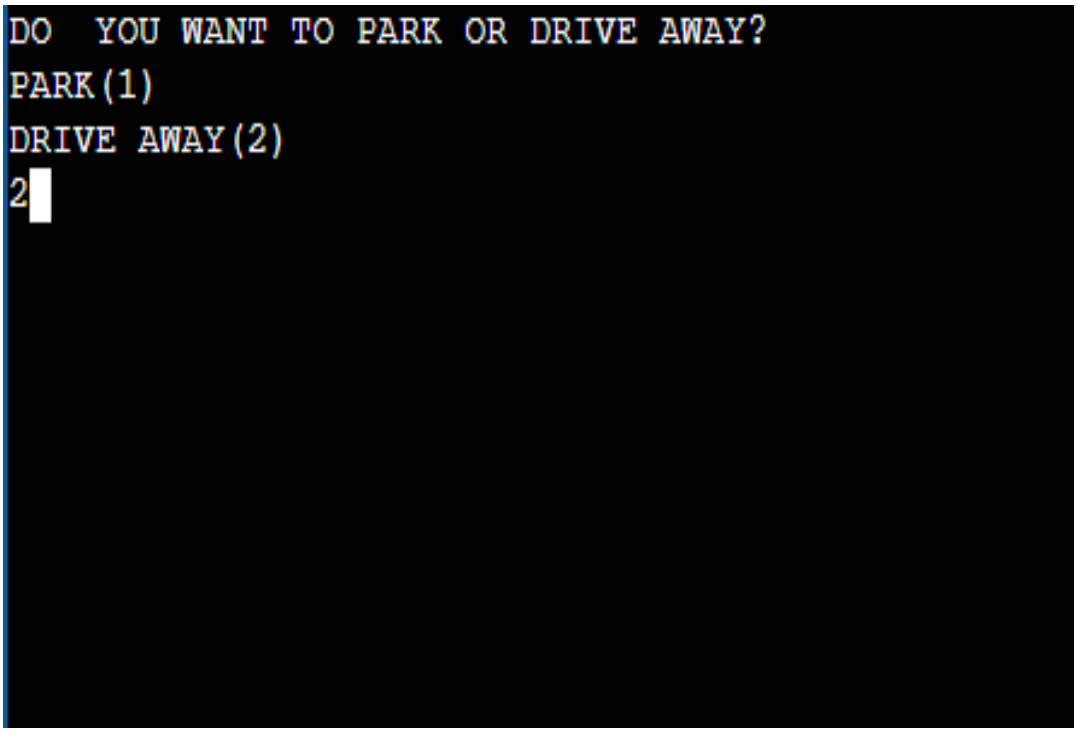
```
----WELCOME TO PARKING AREA----
DO  YOU WANT TO PARK OR DRIVE AWAY?
PARK(1)
DRIVE AWAY(2)
1
```

```
Enter the details:
NAME: REGINA
BRAND: ROLLSROYCE
TYPE(CAR 🚗/ BIKE 🏍️): CAR
NUMBER PLATE: TN88M8888
```

```
PARKING VIEW :

|0| - REPRESENTS EMPTY SLOT
|1| - REPRESENTS OCCUPIED

--------------------------------------------------------------------------------
|1| -> |1| -> |1| -> |0| -> |0| -> |1| -> |0| -> |1| -> |0| -> |0| -> |end|
 |      |      |      |      |      |      |      |      |      |
 V      V      V      V      V      V      V      V      V      V
|1| -> |1| -> |0| -> |1| -> |1| -> |0| -> |0| -> |1| -> |0| -> |1| -> |end|
 |      |      |      |      |      |      |      |      |      |
 V      V      V      V      V      V      V      V      V      V
|0| -> |0| -> |1| -> |1| -> |0| -> |1| -> |1| -> |0| -> |0| -> |1| -> |end|
 |      |      |      |      |      |      |      |      |      |
 V      V      V      V      V      V      V      V      V      V
|0| -> |1| -> |0| -> |0| -> |1| -> |1| -> |0| -> |1| -> |1| -> |0| -> |end|
 |      |      |      |      |      |      |      |      |      |
 V      V      V      V      V      V      V      V      V      V
|0| -> |1| -> |0| -> |1| -> |0| -> |0| -> |1| -> |1| -> |0| -> |1| -> |end|
--------------------------------------------------------------------------------


THERE IS AN EMPTY SLOT IN ROW: 1 COLUMN: 4.
DO YOU WANT TO PARK?(YES(1)/NO(0)):0


THERE IS AN EMPTY SLOT IN ROW: 1 COLUMN: 5.
DO YOU WANT TO PARK?(YES(1)/NO(0)):0


THERE IS AN EMPTY SLOT IN ROW: 1 COLUMN: 7.
DO YOU WANT TO PARK?(YES(1)/NO(0)):1
```

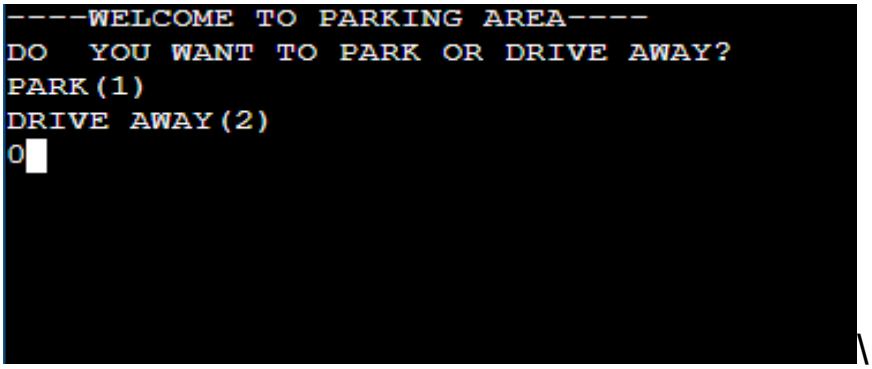
```
YOUR VEHICLE HAS BEEN PARKED IN ROW: 1 COLUMN: 7.
PARKING VIEW :

|0| - REPRESENTS EMPTY SLOT
|1| - REPRESENTS OCCUPIED

--------------------------------------------------------------------------------
|1| -> |1| -> |1| -> |0| -> |0| -> |1| -> |1| -> |1| -> |0| -> |0| -> |end|
 |      |      |      |      |      |      |      |      |      |
 V      V      V      V      V      V      V      V      V      V
|1| -> |1| -> |0| -> |1| -> |1| -> |0| -> |0| -> |1| -> |0| -> |1| -> |end|
 |      |      |      |      |      |      |      |      |      |
 V      V      V      V      V      V      V      V      V      V
|0| -> |0| -> |1| -> |1| -> |0| -> |1| -> |1| -> |0| -> |0| -> |1| -> |end|
 |      |      |      |      |      |      |      |      |      |
 V      V      V      V      V      V      V      V      V      V
|0| -> |1| -> |0| -> |0| -> |1| -> |1| -> |0| -> |1| -> |1| -> |0| -> |end|
 |      |      |      |      |      |      |      |      |      |
 V      V      V      V      V      V      V      V      V      V
|0| -> |1| -> |0| -> |1| -> |0| -> |0| -> |1| -> |1| -> |0| -> |1| -> |end|
--------------------------------------------------------------------------------
Enter any value to generate your recipt: YES
```


RECEIPT OF PARKING
Details of the car:
NAME: REGINA
BRAND: ROLLSROYCE
TYPE: CAR
NUMBER PLATE: TN88M8888
TOKEN NUMBER : 957
PARKING CHARGE : RS.50
Enter 1 to exit.1

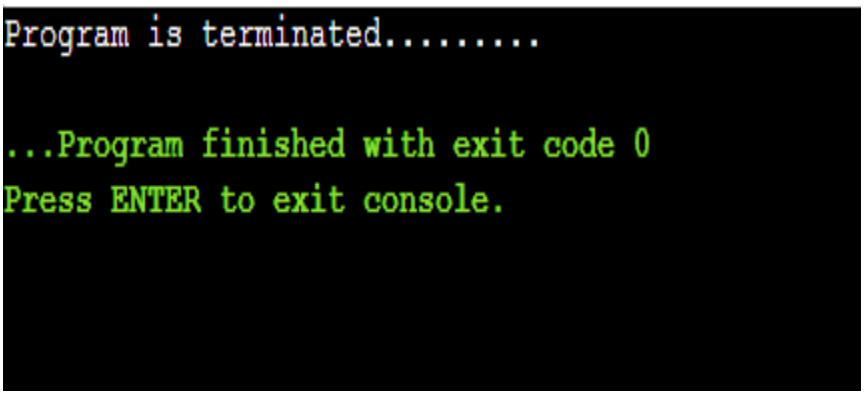

DO YOU WANT TO PARK OR DRIVE AWAY?
PARK(1)
DRIVE AWAY(2)
2

```
ENTER YOUR TOKEN NUMBER: 957

Details of the car:
NAME: REGINA
BRAND: ROLLSROYCE
TYPE: CAR
NUMBER PLATE: TN88M8888
IS THIS YOUR DETAILS?(YES(1)/NO(0)): 1
```

```
YOUR CAR IS IN ROW: 1 COLUMN: 7.PARKING VIEW :

|0| - REPRESENTS EMPTY SLOT
|1| - REPRESENTS OCCUPIED

--------------------------------------------------------------------------------
|1| -> |1| -> |1| -> |0| -> |0| -> |1| -> |0| -> |1| -> |0| -> |0| -> |end|
 |      |      |      |      |      |      |      |      |      |
 v      v      v      v      v      v      v      v      v      v
|1| -> |1| -> |0| -> |1| -> |1| -> |0| -> |0| -> |1| -> |0| -> |1| -> |end|
 |      |      |      |      |      |      |      |      |      |
 v      v      v      v      v      v      v      v      v      v
|0| -> |0| -> |1| -> |1| -> |0| -> |1| -> |1| -> |0| -> |0| -> |1| -> |end|
 |      |      |      |      |      |      |      |      |      |
 v      v      v      v      v      v      v      v      v      v
|0| -> |1| -> |0| -> |0| -> |1| -> |1| -> |0| -> |1| -> |1| -> |0| -> |end|
 |      |      |      |      |      |      |      |      |      |
 v      v      v      v      v      v      v      v      v      v
|0| -> |1| -> |0| -> |1| -> |0| -> |0| -> |1| -> |1| -> |0| -> |1| -> |end|
--------------------------------------------------------------------------------

THANKS FOR CHOOSING OUR PARKING
 VISIT AGAIN
Enter 1 to exit.1
```


----WELCOME TO PARKING AREA----
DO YOU WANT TO PARK OR DRIVE AWAY?
PARK(1)
DRIVE AWAY(2)
0

\


Program is terminated.....
...Program finished with exit code 0
Press ENTER to exit console.