SPECIMEN COVER PAGE

PARKING SYSTEM IN C PROGRAMMING

A PROJECT REPORT

Submitted by

BIDYASAGAR BEHERA (230720100128)
SUBRAT NAYAK (230720100134)
SAMBHUNATH BEHERA (230720100141)
SOUMYA RANJAN PARIDA (230720100144)
JIBAN JAGANNATH JENA (230720100147)

in partial fulfillment for award of the degree of

MASTER OF COMPUTER APPLICATION

Centurion
UNIVERSITY
Shaping Lives...
Empowering Communities...

SCHOOL OF ENGINEERING AND TECHNOLOGY

BHUBANESWAR CAMPUS

CENTURION UNIVERSITY OF TECHNOLOGY AND MANAGEMENT

ODISHA

AUGUST 2023 / JANARY 2024

SPECIMEN COVER PAGE

PARKING SYSTEM IN C PROGRAMMING

A PROJECT REPORT

Submitted by

BIDYASAGAR BEHERA (230720100128)

in partial fulfillment for award of the degree of

MASTER OF COMPUTER APPLICATION

SCHOOL OF ENGINEERING AND TECHNOLOGY

BHUBANESWAR CAMPUS

CENTURION UNIVERSITY OF TECHNOLOGY AND MANAGEMENT

ODISHA

AUGUST 2023 / JANARY 2024

SPECIMEN CERTIFICATE

DEPARTMENT OF MASTER OF COMPUTER APPLICATION

SCHOOL OF ENGINEERING AND TECHNOLOGY

BHUBANESWAR CAMPUS

BONAFIDE CERTIFICATE

Certified that this project report **PARKING SYSTEM IN C PROGRAMING** is the bonafide work of **“BIDYASAGAR BEHERA”** who carried out the project work under my supervision. This is to further certify to the best of my knowledge, that this project has not been carried out earlier in this institute and theuniversity.

SIGNATURE

(Professor Harapriya Senapati)

(Assistant Professor)

Certified that the above mentioned project has been duly carried out as perthe norms of the college and statutes of the university.

SIGNATURE
(Dr. Rakesh Kumar Ray)

HEAD OF THE DEPARTMENT

HOD of Master of computer Application

DEPARTMENT SEAL

DECLARATION

I hereby declare that the project entitled “**PARKING SYSTEM IN C PROGRAMING**” submitted for the “Minor Project” of 1ST semester in Master of Computer Application is my original work and the project has not formed the basis for the award of any Degree / Diploma or any other similar titles in any other University / Institute.

Name of the Students: BIDYASAGAR BEHERA

Signature of the Student:

Registration No: 230720100128

Place: Bhubaneswar

Date:

ACKNOWLEDGEMENTS

We wish to express our profound and sincere gratitude to Professor Harapriya Senapati, Department of Master of Computer Application, SoET, Bhubaneswar Campus,who guided me into the intricacies of this project nonchalantly with matchless magnanimity.

I thank Dr. Rakesh Kumar Ray, Head of the Dept. of Master of Computer Application, SoET, Bhubaneswar Campus and Dr. Sujata Chakravarty, Dean,School of Engineering and Technology, Bhubaneswar Campus for extending their support during Course of this investigation.

I would be failing in my duty if I don't acknowledge the cooperationrendered during various stages of image interpretation by Professor Harapriya Senapati.

I am highly grateful to Professor Harapriya Senapati who evinced keen interest and invaluable support in the progress and successful completion of my project work.

I am indebted to Professor Harapriya Senapati for their constant encouragement, co- operation and help. Words of gratitude are not enough to describe the accommodation and fortitude which they have shown throughout my endeavor.

Name of the Student: BIDYASAGAR BEHERA

Signature of the Student:

Registration No.: 230720100128

Place: Bhubaneswar

Date:

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	CERTIFICATE	i
	DECLARATION	ii
	ACKNOWLEDGEMENT	iii
	ABSTRACT	iv
CHAPTER – 1	INTRODUCTION	
CHAPTER – 2	PROJECT OVERVIEW	
2.1. WE CAN PARK OUR VEHICLE IN OUR OWN SLOT BY PAYING		
2.2. PROBLEM DEFINITION		
CHAPTER – 3	BACKGROUND & OBJECTIVE	
3.1. BACKGROUND		
3.2. OBJECTIVE		
CHAPTER – 4	METHODLOGY & ANALYTICAL COMPUTATION	
4.1. SYSTEM REQUIREMENT		
4.2. SOFTWARE		
4.3 .HARDWARE		
4.4. ANALYTICAL COMPUTATION		
CHAPTER – 5	SOURCE CODE	
CHAPTER – 6	OUTPUT	
CHAPTER – 7	CONCLUSION	
CHAPTER – 8	FUTURE SCOPE	

REFERENCE

ABSTRACT

The main aim of this project is to reduce the traffic in the parking place. Normally we can see in the multiplexes, cinema halls, large industries, and function halls there is problem they have to go and search which line is empty and which line having place to park the vehicle, for parking then they need workers for parking in correct position it is the money consumed process. So to avoid this problem Car Parking System project is implemented.

The Parking System project aims to revolutionize conventional parking management by leveraging advanced technologies to enhance efficiency, user experience, and overall urban mobility. In response to the escalating challenges associated with limited parking spaces and increasing vehicular traffic, this system integrates smart algorithms, real-time monitoring, and user-friendly interfaces to optimize parking operations.

KEY FEATURES

Smart Utilization: The project employs intelligent algorithms to maximize the utilization of available parking spaces, minimizing congestion and improving overall traffic flow within parking facilities.

Real-time Monitoring: By incorporating real-time monitoring capabilities, the system provides instant updates on parking space availability, ensuring users can make informed decisions and reducing the time spent searching for parking.

User-Friendly Interface: The intuitive user interface caters to both administrators and users, simplifying navigation and interaction with the system. This fosters a positive user experience and encourages widespread adoption.

CHAPTER -1 : INTRODUCTION

Parking Management System for managing the records of the incoming and outing vehicles in an Parking Institution.

It’s an easy for Admin to retrieve the data if the Vehicle has been visited through number.
He can get the data.

Now days in many public places such as malls, multiplex system, hospitals, offices market areas there is a crucial problem of vehicle parking. The vehicle parking area has many lanes/slots for car parking. So to park a vehicle one has to took for all the lanes. Moreover this involves a lot of manual labour and investment. Instead of vehicle caught in towing the vehicle can park on safe and security with low cost.

Parking control system has been generated in such a way that it is filled with many secure devices such as, parking control gates, toll gates, time and attendance machine car counting system etc. These features are hereby very necessary nowadays to secure your car and also to evaluate the fee structure for every vehicles entry and exit.

The objective of this project is to build a Vehicle Parking Management System that enables the time management and control of vehicles using Number plate recognition. The system that will track the entry and exit of cars, maintain a listing of cars within the parking lot, and determine if the parking lot is full or not. It will determine the cost of per vehicle according to their time consumption.

CHAPTER -2 : PROJECT OVERVIEW

We can Park our vehicle in our own slot by Paying.

- Because of that there is no towing problems.
- And our vehicle has been parked as a secure condition.
- There is no risk for vehicle owner for parking the vehicle.
- In case of any damages and problem of vehicle that will claim by parking management.
- As the world is facing many threads daily, robberies are done easily with no track , bomb blasts occur with the use of vehicle, so if a proper system is adopted each and every record can be saved and anyone can be track easily therefore mainly is to make a better and fast software, most important user-friendly.
- Maintain records in short time of period.
- Determines the parking area is full or not.
- Enhances the Visitor's experience.

PROBLEM DEFINITION

- Now a days in parking like valet parking they maintain just with the tokens and they have records the vehicle details in books so that during some critical situation like police enquiry of Terrorist car or vehicle roberrer that case it is difficult to find the details of particular vehicle but in this case is easy to find in 1 to 2 seconds.
- By parking the vehicle in public place the vehicle can be claimed by towing person but in this case there is no towing problems and no need to give fine for anything we can park our vehicle with securely.

CHAPTER -3 : BACKGROUND AND OBJECTIVE

The provided code is a simple parking management system written in C. The objective of the project appears to be the creation of a program that manages a parking stand by allowing users to park their vehicles, check for vehicle availability, and provide insights into the day's activities.

BACKGROUND :

The project seems to be designed for a parking stand at CUTM (presumably a university or institution) to manage and monitor the parking of different types of vehicles

Here is a breakdown of the main components and features of the project:

Vehicle Types and Parking Slots:

The system supports four types of vehicles: Bus, Car, Bike, and Auto.

Each vehicle type has a designated number of parking slots (bus_slot, car_slot, bike_slot, auto_slot).

Customer Data Structure:

The struct customer is used to store information about customers, including their name, vehicle number, and date.

Payment and Receipt:

The code handles the payment process based on the type of vehicle and prints a receipt with the customer's name, vehicle number, date, and the amount paid.

Parking Operations:

Separate functions (bus_park, car_park, bike_park, auto_park) handle the parking operations for each type of vehicle.

Users are prompted to enter their name, vehicle number, date, and parking fees. Input validation is performed to ensure that valid fees are entered.

Parking Slot Availability:

Before allowing a vehicle to be parked, the code checks if there are available slots for that specific vehicle type.

Vehicle Check:

Users can check if their vehicle is available in the parking stand by entering their name, vehicle number, and date.

Insight Display:

The system provides insights into the day's activities, including total earnings and the count of parked vehicles for each type.

User Interface:

The main function presents a simple text-based menu for users to choose from various options, such as parking their vehicle, checking their vehicle, viewing insights, or exiting the program.

Looping Menu:

The program runs in a loop, allowing users to perform multiple operations until they choose to exit.

Exit Option:

Users can exit the program gracefully with the option to display a thank-you message.

OBJECTIVE :

The primary objective is to create a functional and user-friendly program for managing parking operations, providing users with necessary information, and generating insights for the parking stand's daily activities.

CHAPTER – 4 : METHODLOGY AND ANALYTICAL COMPUTATION

System Requirement :

Operating System : Windows

Software Requirement : Dev C++

Hardware Requirement : PC

Software :

Dev C++ is a free , oprn-source integrated development environment (IDE) for programming in C and C++ . It uses the MinGW port of GCC (GNU Compiler Collection) as its compiler. Dev C++ provides an environment for writing , compiling and debugging C and C++ programs**.**

Hardware :

A Desktop helps to fast run the source code and display a clear output picture for the reference.

Here is an analysis of the methodology and computational aspects of the project:

Parking Slots and Counts:

The code uses integer variables (bus_slot, car_slot, bike_slot, auto_slot) to represent the available parking slots for each type of vehicle.

Separate count variables (bus_count, car_count, bike_count, auto_count) keep track of the total number of each type of vehicle parked.

Customer Data:

Customer data is stored in a structure (struct customer) with arrays to hold customer names, vehicle numbers, and dates.

The index variables (index1, index2, index3) are used to keep track of the current position in the arrays when storing customer data.

User Input and Validation:

The code uses gets and scanf for user input, with some basic validation for parking fees.

Receipt Printing:

The print_receipt function is responsible for printing a receipt with customer details and the parking fee.

Parking Functions:

Separate functions (bus_park, car_park, bike_park, auto_park) handle the parking operations for each type of vehicle.

The use of a repeat label and goto statement is employed for fee validation, but this could be improved for better readability.

Menu and Control Flow:

The main menu is displayed using a do-while loop, and the user's choice is processed using a switch statement.

Insight Display:

The check_insight function displays insights such as total earnings and the count of each type of vehicle parked.

ANALYTICAL COMPUTATION

In a parking system, analytical computation can be applied for tasks like optimizing parking space allocation, calculating fees based on duration, or analyzing usage patterns. For example, algorithms can help determine the most efficient way to utilize available parking spaces, ensuring an organized and effective system. Additionally, analytical computation can assist in predicting peak usage times and adjusting pricing models accordingly**.**

CHAPTER -5 : SOURCE CODE :

```
#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int bus_slot = 5, car_slot = 5, bike_slot = 20, auto_slot = 15;

int earings = 0, bus_count = 0, car_count = 0, bike_count = 0, auto_count = 0;

int index1 = 0, index2 = 0, index3 = 0;

struct customer

{

    char customerName[10][20], date[10][13];

    int vehicleNumber[10];

} c_data;

void print_receipt(int fees)

{

    //this function use to print receipt

    printf("\nHere you receipt!\n");

    printf("***************************************\n");

    printf("Name: %s\n", c_data.customerName[index2]);

    printf("Vehicle numebr: %d\n", c_data.vehicleNumber[index1]);

    printf("Date: %s\n", c_data.date[index3]);

    printf("Your Fees :%d\n", fees);

    printf("***************************************\n");

}

void bus_park()

{
 //this function use to handle bus parking operations
    int fees;
    printf("Enter your name: ");
    fflush(stdin);
    gets(c_data.customerName[index2]);
    printf("Enter vehicle number: ");
    scanf("%d", &c_data.vehicleNumber[index1]);
    printf("Enter today's date: ");
    fflush(stdin);
    gets(c_data.date[index3]);
repeat:
    printf("Please enter fees: ");
    scanf("%d", &fees);
    if (fees < 200 || fees > 200)
    {
        printf("Please enter valid amount");
        goto repeat;
    }
    else
    {
        earings = earings + fees;
        printf("Your payment is successfully done\n");
    }
    print_receipt(fees);
```

```
  index1++;
    index2++;
    index3++;
    bus_count++;
}
void car_park()
{
    //this function use to handle car parking operations
    int fees;
    printf("Enter your name: ");
    fflush(stdin);
    gets(c_data.customerName[index2]);
    printf("Enter vehicle number: ");
    scanf("%d", &c_data.vehicleNumber[index1]);
    printf("Enter today's date: ");
    fflush(stdin);
    gets(c_data.date[index3]);
repeat:
    printf("Enter parking fees: ");
    scanf("%d", &fees);
    if (fees < 150 || fees > 150)
    {
        printf("Please enter valid amount");
        goto repeat;
    }
```

```
 else
   {
          earings = earings + fees;
       printf("Your payment is successfully done\n");
   }
   print_receipt(fees);
   index1++;
   index2++;
   index3++;
   car_count++;
}
void bike_park()
{
   //this function use to handle bike parking operations
   int fees;
   printf("Enter your name: ");
   fflush(stdin);
   gets(c_data.customerName[index2]);
   printf("Enter vehicle number: ");
   scanf("%d", &c_data.vehicleNumber[index1]);
   printf("Enter today's date: ");
   fflush(stdin);
   gets(c_data.date[index3]);
repeat:
```

```
printf("Enter parking fees: ");
    scanf("%d", &fees);
    if (fees < 50 || fees > 50)
    {
        printf("Please enter valid amount");
        goto repeat;
    }
    else
    {
        earings = earings + fees;
        printf("Your payment is successfully done\n");
    }
    print_receipt(fees);
    index1++;
    index2++;
    index3++;
    bike_count++;
}
void auto_park()
{
    //this function use to handle auto parking operations
    int fees;
    printf("Enter your name: ");
    fflush(stdin);
    gets(c_data.customerName[index2]);
```

```
 printf("Enter vehicle number: ");
   scanf("%d", &c_data.vehicleNumber[index1]);
   printf("Enter today's date: ");
   fflush(stdin);
   gets(c_data.date[index3]);
repeat:
   printf("Enter parking fees: ");
   scanf("%d", &fees);
   if (fees < 100 || fees > 100)
   {
      printf("Please enter valid amount");
      goto repeat;
   }
   else
   {
         earings = earings + fees;
      printf("Your payment is successfully done\n");
   }
   print_receipt(fees);
   index1++;
   index2++;
   index3++;
   auto_count++;
}
```

```
void park_vehicle()
{
    //this function use to handle parking operations
    int choose;
again:
    printf("\n1.Bus                    2.Car\n");
    printf("3.Bike                   4.Auto\n");
    printf("Which vehicle you want to park: ");
    scanf("%d", &choose);
    switch (choose)
    {
    case 1:
        if (bus_slot == 0)
        {
             printf("\nSorry! Bus parking slot is not available\n");
        }
        else
        {
            bus_park();
            bus_slot--;
        }
        break;
    case 2:
        if (car_slot == 0)
        {
```

```
printf("\nSorry! Car parking slot is not available\n");
       }
       else
       {
           car_park();
           car_slot--;
       }
       break;
   case 3:
       if (bike_slot == 0)
       {
           printf("\nSorry! Bike parking slot is not available\n");
       }
       else
       {
           bike_park();
           bike_slot--;
       }
       break;
   case 4:
       if (auto_slot == 0)
       {
           printf("\nSorry! Auto parking slot is not available\n");
       }
```

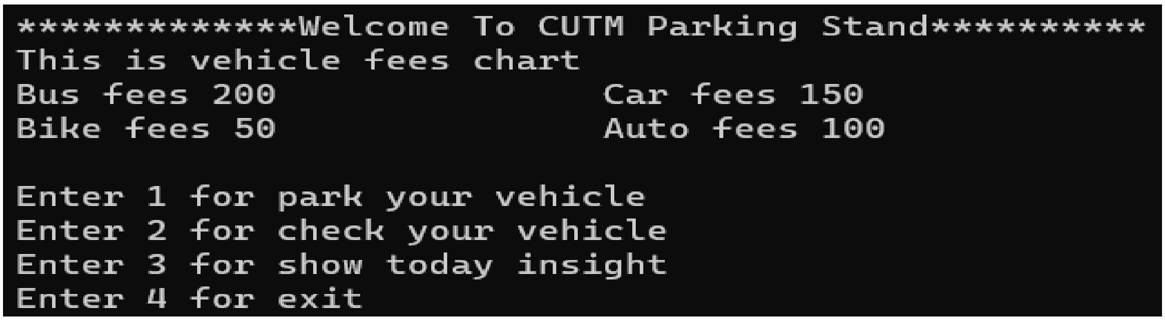
```
 else
        {
            auto_park();
            auto_slot--;
        }
        break;
        default:
        printf("Invalid number try again\n");
        goto again;
    }
}
void check_vehicle()
{
    //This function use to check your vehicle is available or not
    char customerName1[20], date1[13];
    int vehicleNumber;
    int i = 0, check1, check2, check3;
    printf("\nPlease enter some details\n");
    printf("Enter Name: ");
    fflush(stdin);
    gets(customerName1);
    printf("Enter vehicle number: ");
    scanf("%d", &vehicleNumber);
    printf("Enter date: ");
    fflush(stdin);
```

```
gets(date1);
  while (1)
  {
      check1 = strcmp(customerName1, c_data.customerName[i]);
      check3 = strcmp(date1, c_data.date[i]);
      if (vehicleNumber == c_data.vehicleNumber[i])
      {
          check2 = 0;
      }
      if (check1 == 0 && check2 == 0 && check3 == 0)
      {
          printf("\nYour vehicle is available here\n");
          break;
      }
      else
      {
          if (index1 == 9)
          {
              printf("\nSorry! your vehicle is not available here\n");
              break;
          }
      }
      i++;
  }
  }
```

```
void check_insight()
{
   //This function use to check parking insight
   printf("\n*************Today's insight***************\n");
   printf("Total earnings: %d\n", earings);
   printf("Total Bus parked: %d\n", bus_count);
   printf("Total Car parked: %d\n", car_count);
   printf("Total Bike parked: %d\n", bike_count);
   printf("Total Auto parked: %d\n", auto_count);
}
void main()
{
   //This function use to handle start menu
   int choose;
   do
   {
      printf("\n*************Welcome To CUTM Parking Stand**********\n");
      printf("This is vehicle fees chart\n");
      printf("Bus fees 200            Car fees 150\n");
      printf("Bike fees 50            Auto fees 100\n");
      printf("\nEnter 1 for park your vehicle\n");
      printf("Enter 2 for check your vehicle\n");
      printf("Enter 3 for show today insight\n");
      printf("Enter 4 for exit\n");
   again:
```

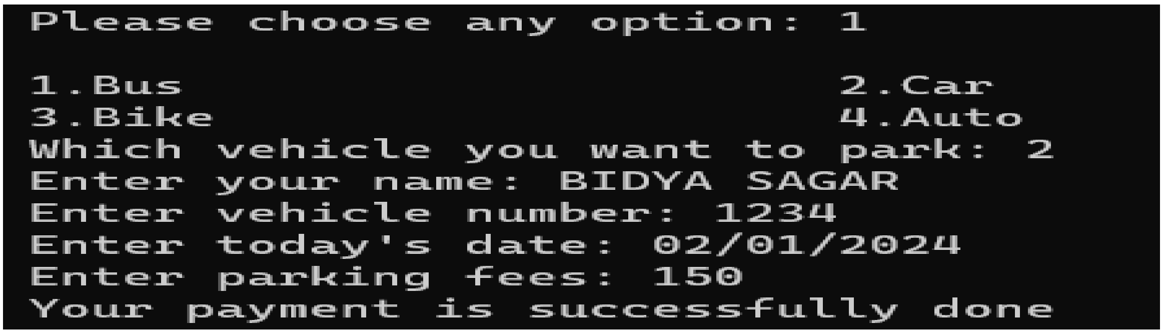
```
 printf("Please choose any option: ");
        scanf("%d", &choose);
        switch (choose)
        {
        case 1:
            park_vehicle();
            break;
        case 2:
            check_vehicle();
            break;
        case 3:
            check_insight();
            break;
        case 4:
            printf("Thank you for using our parking stand\n");
            exit(0);
            break;
            default:
            printf("Invalid number try again\n");
            goto again;
            break;
        }
    } while (choose!=4);
}
```

CHAPTER -6 : OUTPUT



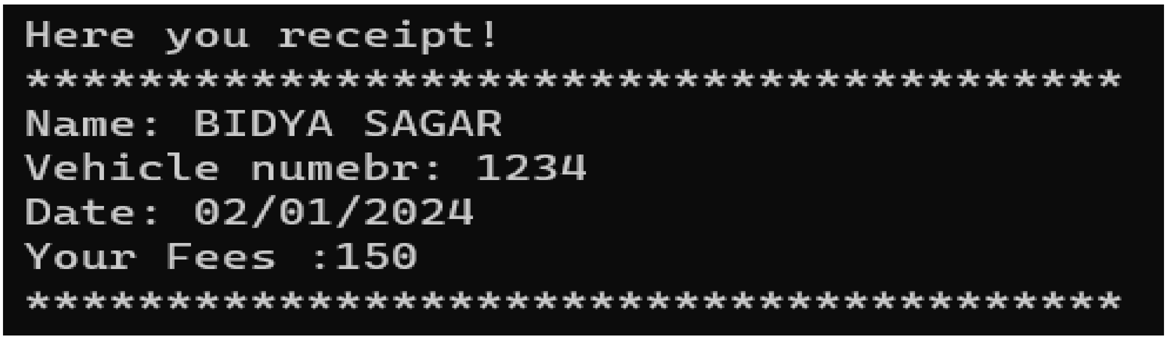
```
*************Welcome To CUTM Parking Stand**********
This is vehicle fees chart
Bus fees 200                Car fees 150
Bike fees 50                Auto fees 100

Enter 1 for park your vehicle
Enter 2 for check your vehicle
Enter 3 for show today insight
Enter 4 for exit
```

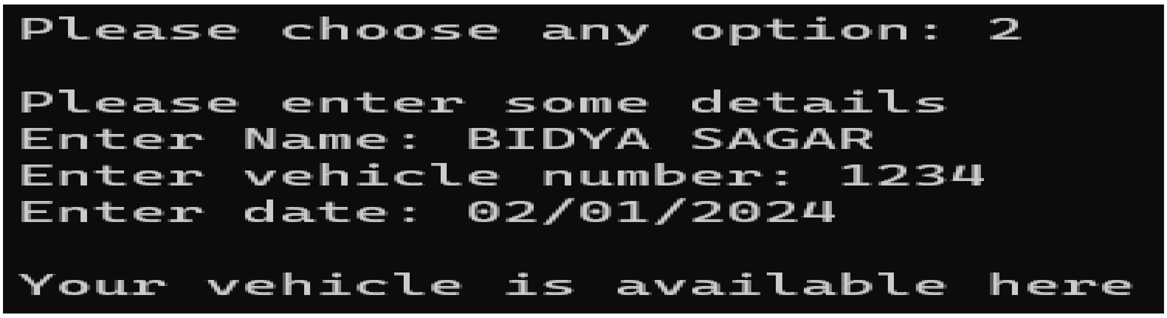


```
Please choose any option: 1

1.Bus                         2.Car
3.Bike                        4.Auto
Which vehicle you want to park: 2
Enter your name: BIDYA SAGAR
Enter vehicle number: 1234
Enter today's date: 02/01/2024
Enter parking fees: 150
Your payment is successfully done
```



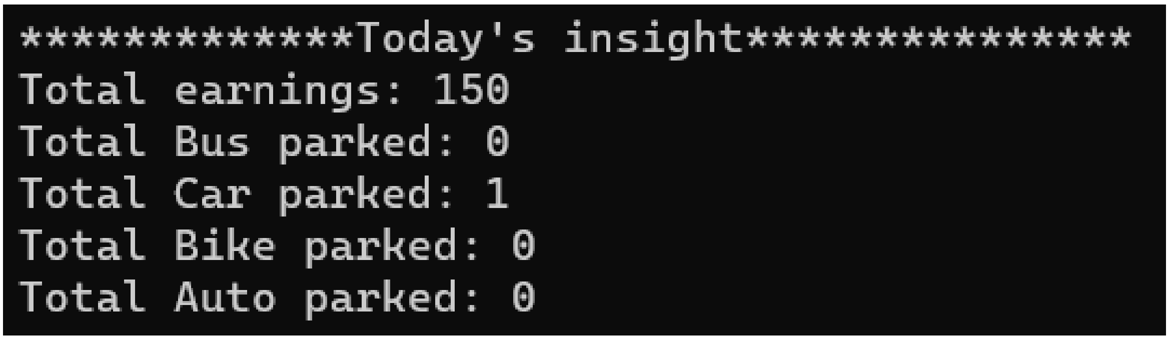
```
Here you receipt!
****************************************
Name: BIDYA SAGAR
Vehicle numebr: 1234
Date: 02/01/2024
Your Fees :150
****************************************
```



```
Please choose any option: 2

Please enter some details
Enter Name: BIDYA SAGAR
Enter vehicle number: 1234
Enter date: 02/01/2024

Your vehicle is available here
```



```
*************Today's insight****************
Total earnings: 150
Total Bus parked: 0
Total Car parked: 1
Total Bike parked: 0
Total Auto parked: 0
```

CHAPTER -7 : CONCLUSION

In conclusion, the Parking System project has successfully addressed the pressing need for efficient and organized parking management. The implementation of this system has significantly improved the overall parking experience for users, providing real-time information, streamlined operations, and enhanced security.

Parking management systems can provide smart and easy solutions for parking challenges. They can help with convenience, safety, and the environment. Parking management systems can also provide flexibility and convenience, such as allowing employees to reserve parking spaces in advance.

Multi-Vehicle Parking: The system allows parking for different vehicle types, including buses, cars, bikes, and autos.

Slot Management: The program tracks and manages available slots for each vehicle type, preventing overbooking.

Customer Information: Customer details such as name, vehicle number, and date are stored in the c_data structure.

Payment Processing: The system calculates and records earnings based on parking fees. It includes a receipt feature for each transaction.

Security Measures: Enhance security by incorporating user authentication or validation mechanisms.

Input Validation: Implement robust input validation to handle various scenarios and prevent invalid user inputs.

Dynamic Memory Allocation: Consider using dynamic memory allocation to handle a variable number of customers and improve scalability.

CHAPTER -8 : FUTURE SCOPE

Enhanced Automation: Future iterations could explore increased automation, potentially integrating with smart sensors and autonomous vehicle technologies to further optimize parking operations.

Advanced Analytics: Implementing advanced analytics could provide valuable insights into usage patterns, helping optimize space allocation and improve overall system efficiency.

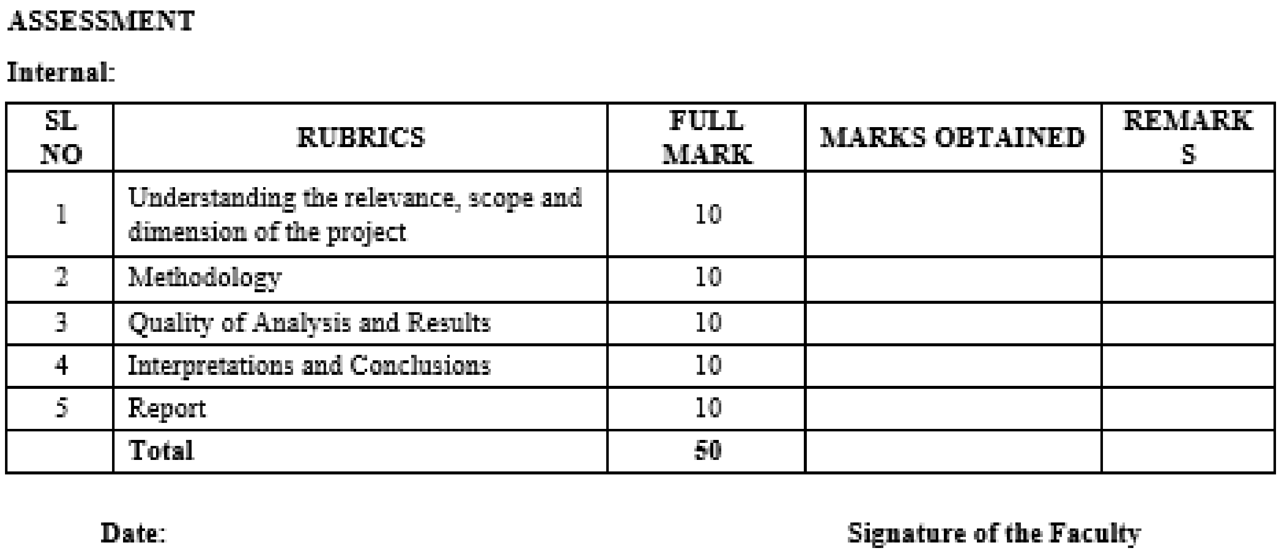
Scalability: As the system expands, considerations for scalability and adaptability to different environments will be crucial for its continued success.

Error Handling: Implement error-handling mechanisms to gracefully handle unexpected issues, such as memory allocation failures.

Code Refactoring: Organize the code into functions, making it more modular and readable. This can improve maintainability and ease of understanding.

REFERENCE

- ➢ https://youtu.be/TL7FDAtRLK4?si=sENi-_GamGdYMN90
- ➢ https://www.slideshare.net/RaushanKumar511746/parking-system-management-in-c-programing-language-project
- ➢ https://www.allprogramminghindi.in/2022/03/parking-system-project-in-c-language-in-hindi.html



ASSESSMENT

Internal:

SL NO	RUBRICS	FULL MARK	MARKS OBTAINED	REMARKS
1	Understanding the relevance, scope and dimension of the project	10		
2	Methodology	10		
3	Quality of Analysis and Results	10		
4	Interpretations and Conclusions	10		
5	Report	10		
	Total	**50**		

Date: **Signature of the Faculty**

COURSE OUTCOME (COs) ATTAINMENT

- **Expected Course Outcomes (COs):**

(Refer to COs Statement in the Syllabus)

____

____

____

____

- **Course Outcome Attained:**

How would you rate your learning of the subject based on the specified COs?

□	□	□	□	□	□	□	□	□	□
1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
LOW									**HIGH**

- **Learning Gap (if any):**

____

____

____

____

- **Books / Manuals Referred:**

____

____

____

____

Date: **Signature of the Student**

- **Suggestions / Recommendations:**

(By the Course Faculty)

____

____

____

____

Date: **Signature of the Faculty**