Abstract:

This project aims to create a parking management system using C programming language. The program provides the user with four parking options, i.e., bus, car, bike, and auto. The user can choose any of the options and park their vehicle by paying the parking fees. The program maintains the parking slot availability and the number of vehicles parked in each slot. The program generates a receipt after the payment is successful.

Introduction:

The parking management system is essential in modern cities to manage the increasing number of vehicles on the roads. This project provides a simple yet effective solution to manage parking slots and help vehicle owners park their vehicles conveniently. The system uses C programming language and provides an easy-to-use interface for the users. The program keeps track of the parking slot availability and generates a receipt after the payment.

Methodology:

The program uses a modular approach to handle parking operations. The program has four functions for each parking option. Each function handles the parking operation and maintains the parking slot availability. The program uses a structure to store customer data such as customer name, vehicle number, and date. The program uses a switch case to allow the user to choose the parking option. The program uses a print receipt function to generate a receipt after the payment.

Components:

The components of the program are as follows:

Header files: The program uses standard C header files such as stdio.h, stdlib.h, and string.h.

Global variables: The program uses global variables to maintain the parking slot availability and the number of vehicles parked in each slot.

Structure: The program uses a structure to store customer data such as customer name, vehicle number, and date.

Functions:

In this project, we have created 8 user defined functions whose names are as follows - park_vehicle function

check_vehicle function.

check_insight function

bus_park function car_park function

bike_park function

auto_park function

print_receipt function

In this project we have created these 8 user defined functions. Now we will know about these functions in a little more detail, which function will do what work, for which it has been made.

main function :- You must know that it is necessary to have main function in every C program. So in this project also there is main function inside which we have done some coding related to printing some messages and creating menu. So its job is to print the menu message and execute the related operation from the same menu which the user chooses. From main function we will call three user defined functions. Which will be called based on the menu.

First user defined function is park_vehicle second is check_vehicle and third is check_insight. If the user chooses a menu related to parking a vehicle, the park_vehicle function will be called. If the user selects a menu related to checking his vehicle, the check_vehicle function will be called. And if the user chooses the related menu to see the insight of today's parking then the check_insight function will be called.

1.park_vehicle function : Inside the park_vehicle function, we have done coding related to parking the vehicle like asking the user which vehicle he wants to park. And then will call the related function from that vehicle. And will complete the operation of parking that vehicle.

For example, if the user wants to park the bus, this function will call the bus_park function. If the user wants to park the car, the car_park function will be called. If user wants to park bike then bike_park function will be called and if user wants to park auto then auto_park function will be called. After this these functions will do their work.

2.check_vehicle function :- Inside the check_vehicle function, we have done the coding related to checking the vehicle. This function will first ask you some details and then check your details whether there is any vehicle park related to these details or not. This function will do just that.

3.check_insight function :- Inside this function, we have done coding related to checking the insight. Insight means that how much money was earned today, how many buses were parked, how many cars were parked, how many bikes were parked and how many autos were parked. So we have done the coding related to printing this inside this function.

4.bus_park function :- Inside this function, we have done the coding related to parking the bus. This function will ask some details from the user while parking the bus, this function will handle the same and will also call the print_receipt function. So the job of this function is to handle the operations related to parking the bus.

5.car_park function :- Inside this function, we have done the coding related to parking the car. This function will ask some details from the user while parking the car, this function will handle the same and will also call the print_receipt function. So the job of this function is to handle the operations related to parking the car.

6.bike_park function :- Inside this function, we have done coding related to parking the bike. While parking this function bike some details will be asked from the user, this function will handle the same and will also call the print_receipt function. So the job of this function is to handle the operations related to parking the bike.

7.auto_park function :- Inside this function, we have done the coding related to parking the auto. While parking this function auto, some details will be asked from the user, this function will handle the same and will also call the print_receipt function. So the job of this function is to handle the operations related to parking the auto.

8.print_receipt function :- Inside this function, we have done coding to print the receipt. This function will handle the operation related to printing the receipt. User's name, vehicle number, date and parking fee will be inside the receipt and this function will print

the same

Implementation:

The implementation of the program is straightforward. The program provides an easy-to-use interface for the users. The user can choose the parking option, and the program handles the parking operation. The program maintains the parking slot availability and the number of vehicles parked in each slot. The program generates a receipt after the payment is successful. The program can be improved by adding more features such as a database to store customer data and a graphical interface for the users.

Conclusion:

The parking management system is essential in modern cities to manage the increasing number of vehicles on the roads. This project provides a simple yet effective solution to manage parking slots and help vehicle owners park their vehicles conveniently. The program uses C programming language and provides an easy-to-use interface for the users. The program maintains the parking slot availability and generates a receipt after the payment. The program can be improved by adding more features such as a database to store customer data and a graphical interface for the users.