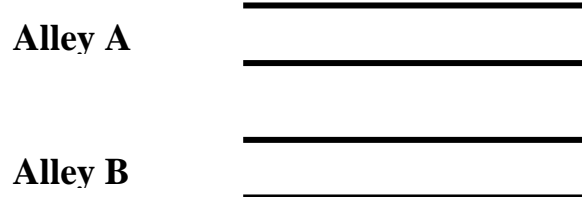


Parking lot simulation

This program simulates the operations of a parking lot. This particular parking lot consists of two alleys, each of which is wide enough for only one car and is closed at one end. You must use a Link List.



Alley A is the primary parking location. Alley B is used only as a place to move cars out of the way when retrieving a car parked in the middle of Alley A.

Program execution starts with both alleys empty. The program repeatedly prompts the user to specify one of four commands:

1. Park a car
2. Retrieve a car
3. Display the contents of Alley A
4. Terminate the program

To park a car, the program generates a new ticket number (1,2,...), issues a ticket stub to the customer, and parks the car at the front of Alley A.

To retrieve a car, the program prompts the user for the ticket stub number and begins to search Alley A. The program moves each car at the front of Alley A to Alley B until it finds the desired car. All of the cars that were temporarily placed in Alley B are then moved back to Alley A.

Input for the program

Each input command consists of a single lowercase or uppercase letter:

1. 'd' or 'D' (To display the alley contents)
2. 'p' or 'P' (To park a car)
3. 'r' or 'R' (To retrieve a car)
4. 'q' or 'Q' (to terminate the program)

C++ Programming, Comprehensive

Output from the program

The main prompt for user commands is the following:

D)isplay P)ark R)etrieve Q)uit:

In response to the Park command, the program generates a ticket number and displays this number, denoted below by <t>:

Ticket no. = <t>

In response to Retrieve command, the program displays the following prompt for the ticket stub number:

Ticket no. :

The output for the Display command is as follows, where each <t> represents the integer ticket number of a car in the alley:

Alley A: <t> <t> <t> ...

Error-Handling

1. The program ignores invalid user commands.
2. If the user tries to retrieve a car with a nonexistent ticket stub number, the program displays the message: "CAR NOT PARKED IN MY LOT."
3. If Alley A is full and the user wants to park a car, the program displays the message: "PARKING LOT FULL".

Assignment #2 Grading Policy

Category	Points Possible	Points Received
Use of a Link List.	20	
Error-Handling	15	
Use of enum type.	15	
Program Output	10	
Correctness and Efficiency	10	
Style and Code Readability	10	
Complete Documentation	10	
User-friendliness	10	
Total	100	

Things to Remember

- Program that is turned in late will lose 10% each week.
- Make sure that you do appropriate error checking. (User-friendliness)
- Do not turn in incomplete or crashing program.

Example

Below is a sample execution of this program with all inputs:

D) isplay P) ark R) etrieve Q) uit: P
Ticket no. = 1

D) isplay P) ark R) etrieve Q) uit: P
Ticket no. = 2

D) isplay P) ark R) etrieve Q) uit: P
Ticket no. = 3

D) isplay P) ark R) etrieve Q) uit: d
Alley A: 3 2 1

D) isplay P) ark R) etrieve Q) uit: r
Ticket no. : 2

D) isplay P) ark R) etrieve Q) uit: r
Ticket no. : 5
CAR NOT PARKED IN MY LOT