**\_\_\_\_\_**

Imagine a **tollbooth** at a bridge. Cars passing by the booth are expected to **pay a 50 hlala** toll. sometimes a car goes by without paying.

The tollbooth keeps track of the number of cars that have gone by, and of the total amount of money collected.

لنتخيل ان لدينا كشكات لتحصيل الرسوم على السيارات التي تمر على جسر

كل سياره تمر يجب عليها ان تدفع 50 هلله كرسوم. يحصل في بعض المرات ان تمر بعض السيارات بدون ان تدفع .

هذه الكشكات تقوم بحساب عدد السيارات التي مرت بالاضافه لكمية المال التي تم جمعها .

قومي بعمل برنامج يمثل وظيفة هذه الكشكات كالتالي :

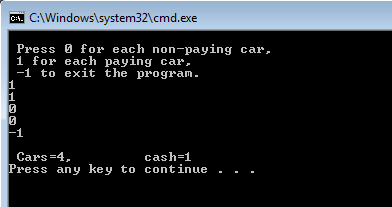
* Model this tollbooth with a **class** called **tollBooth**.
* The class contained **two** data items , the first one is from type **int** to hold the total number of cars.
* The second one is from type **double** to hold the total amount of money collected.
* A **constructor** initializes both of these variables to 0.
* A member function called **payingCar()** increments the car total and adds 0.50 to the cash total.
* A member function called **nopayCar**(), increments the car total but adds nothing to the cash total.
* a member function called **display()** displays the two totals.

Write a main() test this class.

This program should allow the user to enter (1) count a paying car, and( 0) to count a non-paying car.

Entering (-1) should cause the program to print out the total cars and total cash and then exit.

The run for the programme could be like this:



1. create a class called array manipulations that contains:

private variables :

* int array called X
* max (to determine the capacity) , size (size of array )
* constructor : to initialize the variables and create the array with a given max size.
* Display : to print the array content
* Full : return true if array is full and false other wise.
* Empty : return true if array is full and false other wise.
* Retrieve : retrieve the value for a given index
* Update : update the value for a given index
* Insert : insert a value in the array.
* remove : remove the last added value
* remove : remove a value in a given index(replace the value with zero);

write a test class to test your class