

**Universidad de las Américas, Puebla**

**Course:**

Data Structures

**Assignment 2:** “Graphic User Interface”

**Teacher:**

Ingrid Kirschning Albers

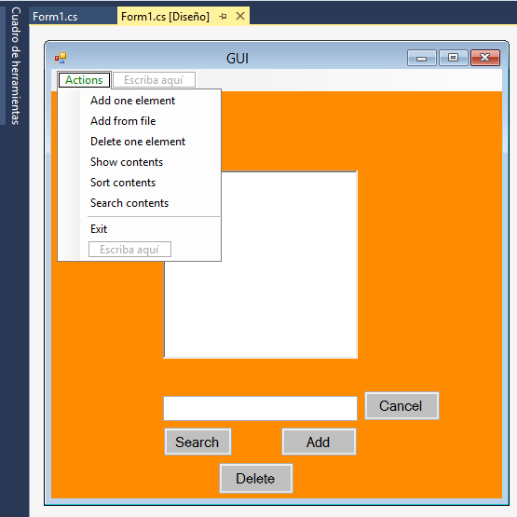
**Name:** Raúl González Cruz

**ID:** 151211

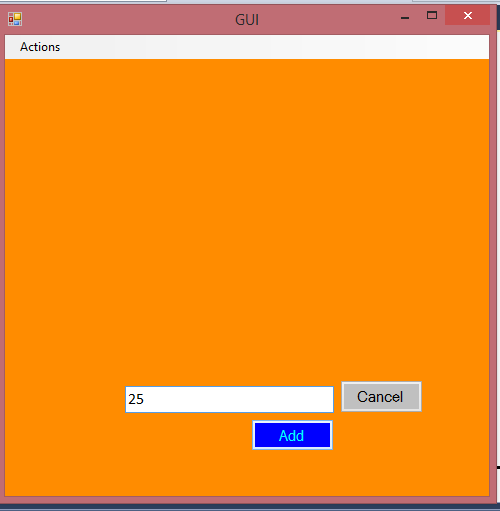
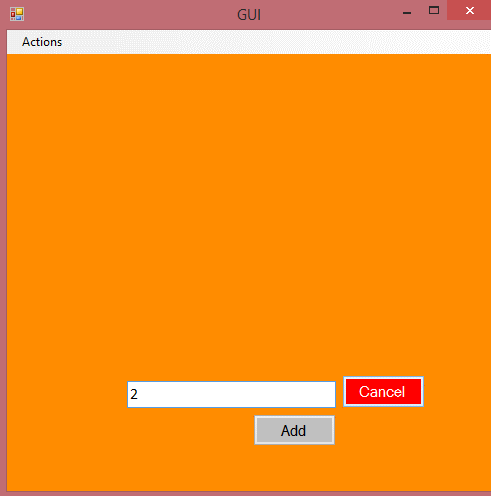
**Date:** 10/09/2015

**Explanation of the program**

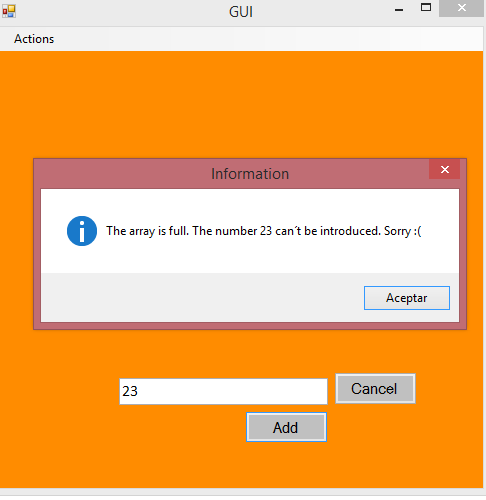
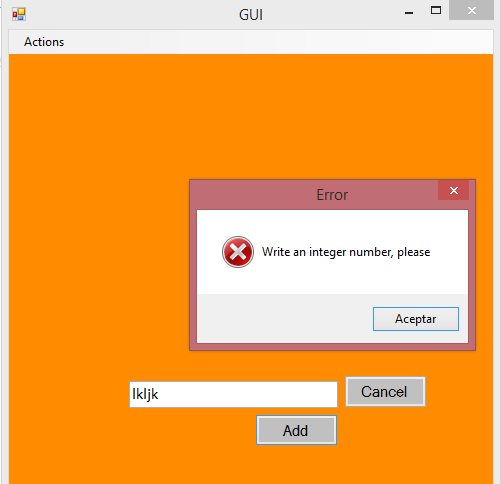
To begin with this work, the Graphic User Interface I have done in Visual Studio with C# is this:

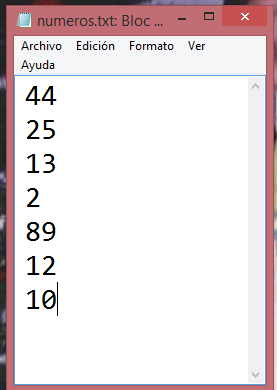


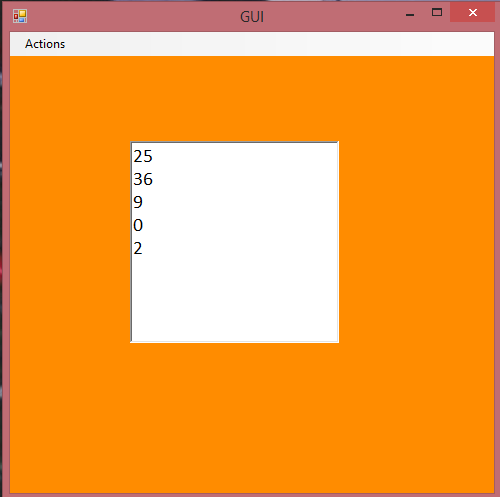
Then, we start working with the action “Add one element” adding the number 25. The, we add 36, 9, 0 and 2. As you can see, this option functions well.

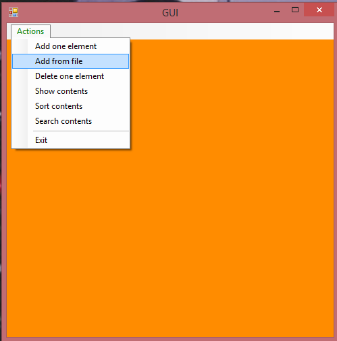
And remeber, the button “Cancel” prevents to add more values if the user has picked up by mistake.

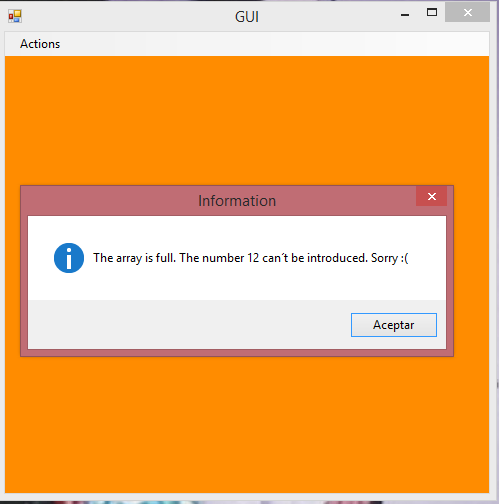
If the array is full and we want to add another element, the program do this:

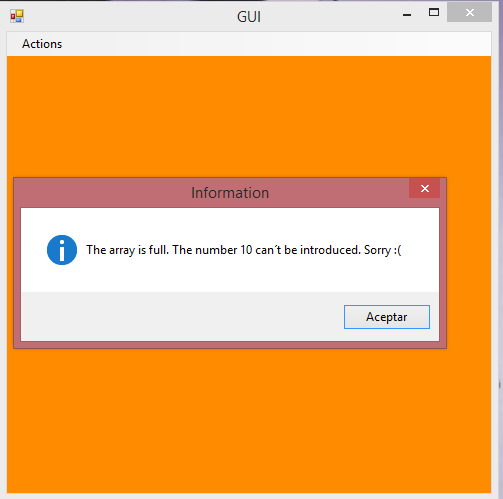
And remember to write a number, because the program would do this:

Another action that can be performed it´s when the user pick up the option “Add from File”. When someone pick up this option, the program call a text file that contains a column of numbers. Then, our array add these numbers, if there is space. So, the program can add 5 more numbers, beacuse the array can contains only 10 numbers.

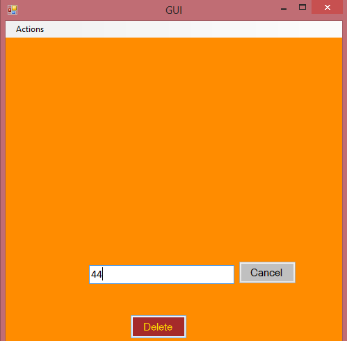
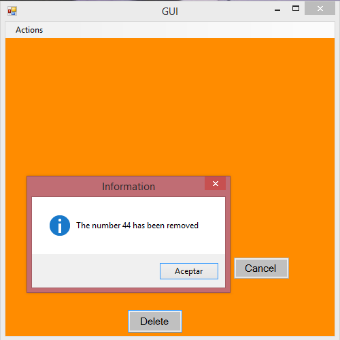
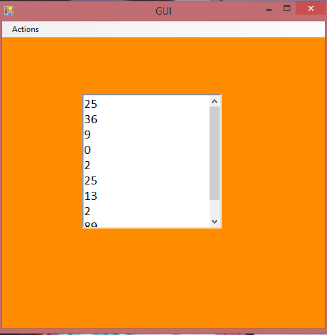


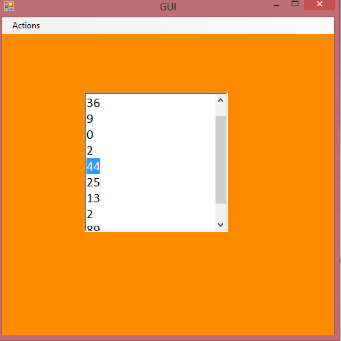






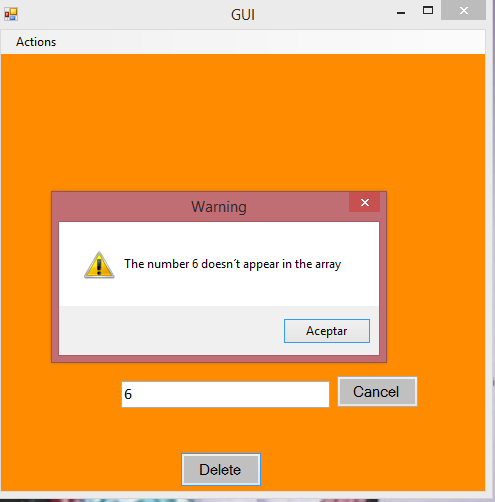
The program can´t add 12 and 10, because we add first 5 numbers (25,36,9,0,2) and the 44,25,13,2 and 89. At this point, the array contains 10 elements; and it can´t add one more number.

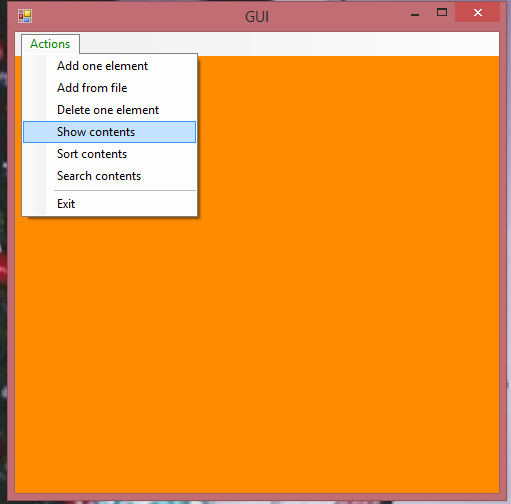
The next option is chosen when the user wants to delete an element. The user needs to define a number that is contained in the array. For example:



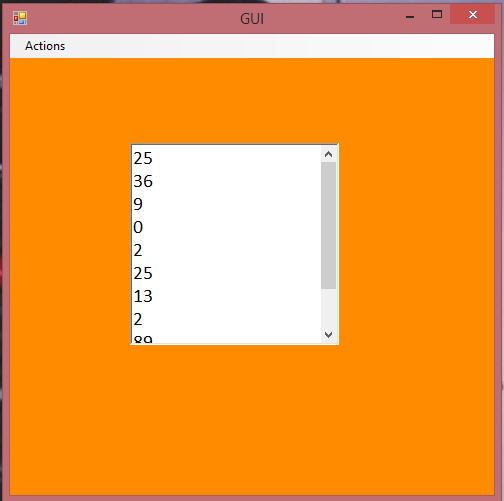
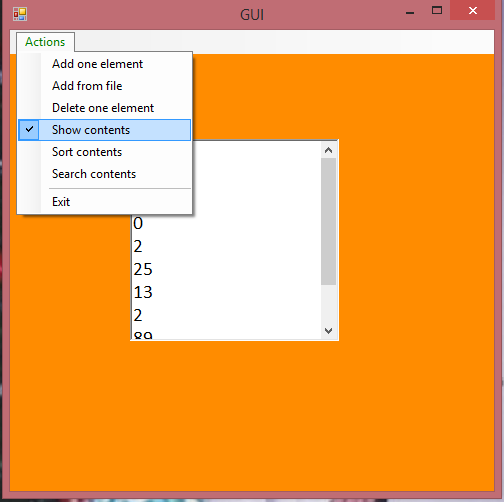
But what happen if the user wants to delete a number that doesn´t appear in the array:

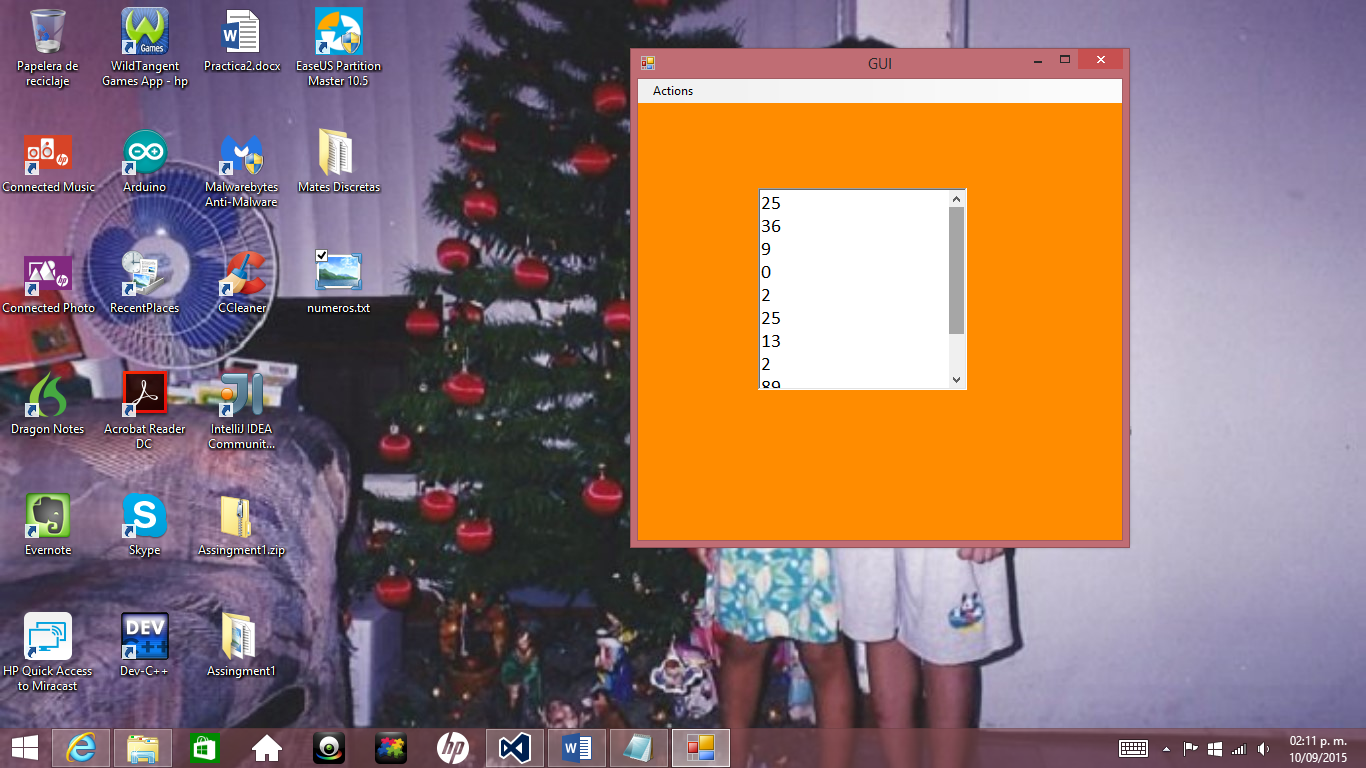
Finally, we asume that this option functions as we want, and if we want, we can add a number. So, for this example, I will add 1.

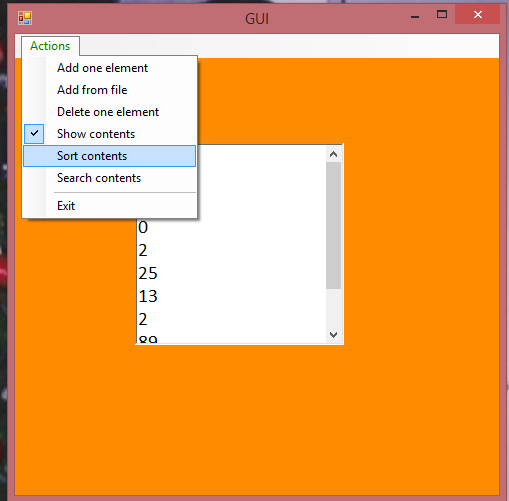
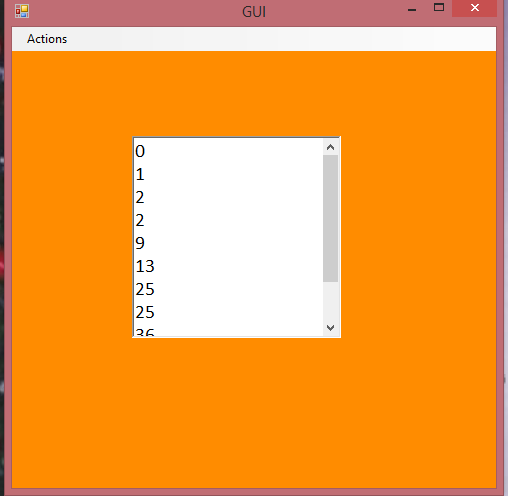
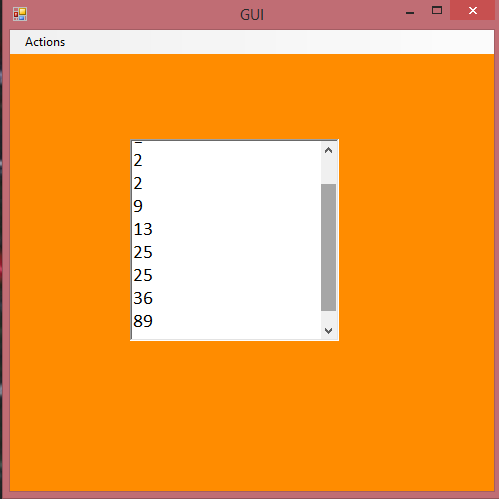


Another useful option is “Show contents”. This program can show the array, but also it can hide it; if the user want it. First, the program hide the array. So, we have to pick up this option if we want to see the array. If the array ist empty, a message is showed. And then, the list appears.

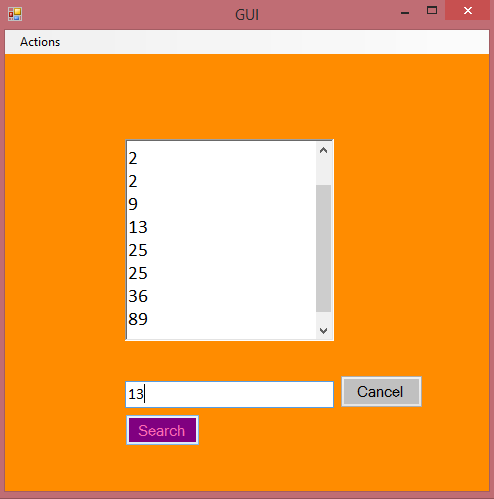
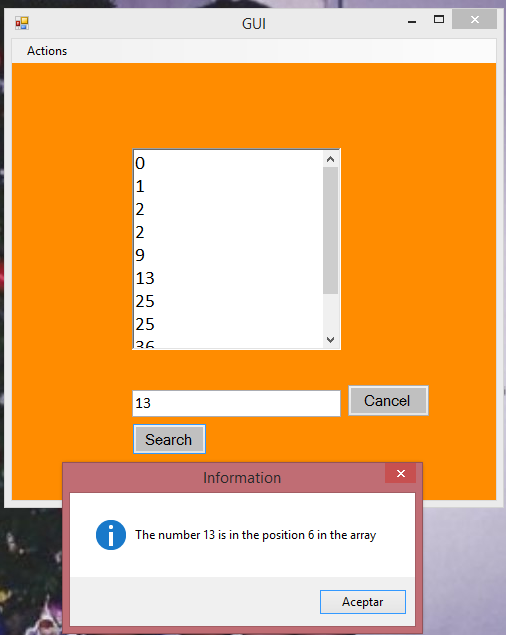
If you want to hide the list, you need to unselect this option. Finally, we conclude that this option works well, beacuse it only do visible or invisible the richtextbox that contains the elements of the array.

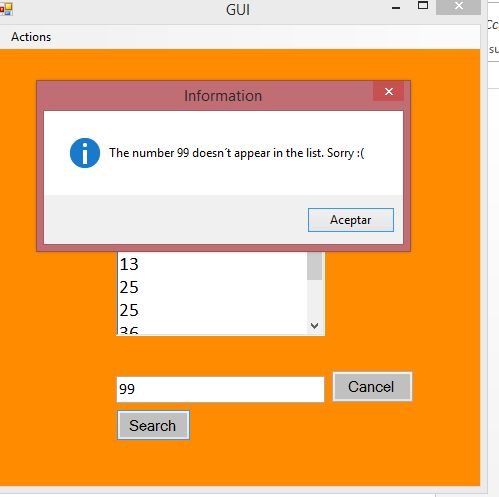


The “Sort” option has been programmed using the quicksort method, because I consider it´s the best option. The elements are sorted when this option is chosen. Considering the array we were working at this point, we have to show the option “sort elements”:



So the next option is “Search contents”. For this option, is not necessary to show the list, but if you want to check it, it´s a good idea to show the list. We have to write a number in the text box and pick up the button Search. We can try, for example, with 13:





And it shows that the number 13 is in the position 6 on the array, and that´s correct. And what happen if we want to serach the number 99:

Exactly! So, this option functions.

Finally, we have the option “Exit”. This option only finishes with the program, so it functions very well. Only we select it and that´s all. In this case, you have to prove it by yourself.

