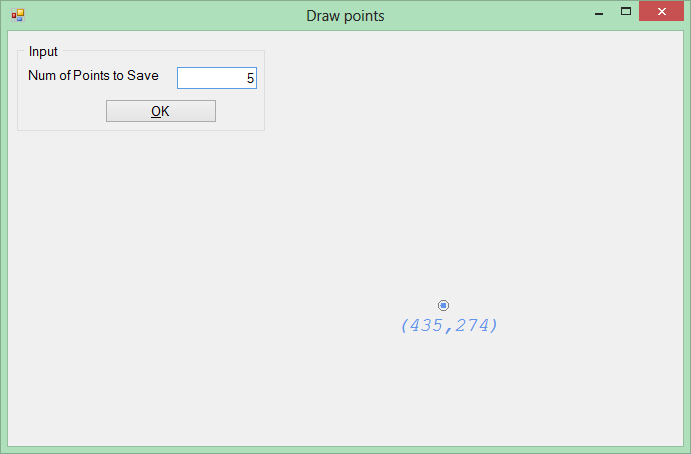
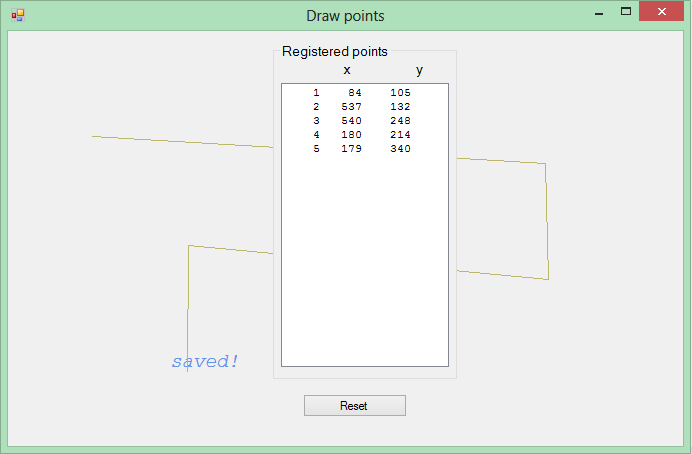
Exercise – Draw Polypoints

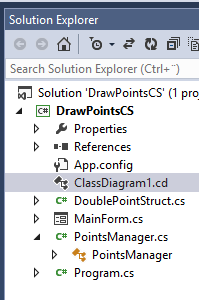
1. Objectives

The objectives of this exercise are to practice with the following:

* One-dimensional arrays
* Structs

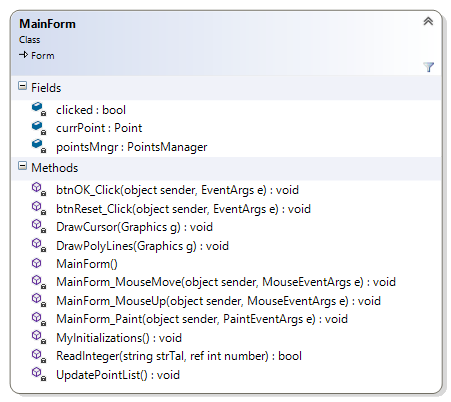
1. Description

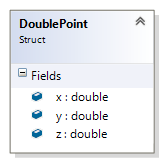
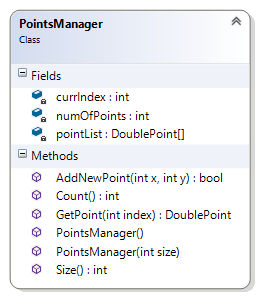
Write an application that keeps track of the mouse-clicking on a form. The user gives the number of points to be saved in an array, as in the image at left. After clicking the OK-button, a listbox is displayed and the coordinates (x, y) of the every clicking (up to the size of the array) on the form is recorded in the array. Meanwhile the, the points are draw on the drawing space of the form as in the image at right.

1. Instructions

Follow the class diagram given below.

* 1. Define a struct as shown in the class diagram below.
  2. Write a class PointManager in which you create an array of the struct.
  3. Handle the Mouse-up and Paint events of the form to perform the drawing.





1. Help and solution
   1. Usually, it is not the form on which you would want to make your drawings. A Panel controls very appropriate to draw on. It will then be the Panel's Paint-event that you should use to draw.
   2. Note also that you should not call the Paint-handler method to redraw. Controls have a method called Invalidate() that causes a redraw. So whenever you want to refresh your drawing, call the Invalidate() method (using a panel, call the panel's Invalidate).
   3. The Paint-handler method has a parameter **e** that contains valuable information. It has an object of Graphics that you can use for drawing lines, and other figures (and even strings).

private void MainForm\_Paint(object sender, PaintEventArgs e)

* 1. The MouseUp-handler method also has such a method parameter **e** that maintains the coordinate of the mouse position.

private void MainForm\_MouseUp(object sender, MouseEventArgs e)

* 1. A suggested solution is given via the course’s Its L. Download the project and run the application without looking at the source code to see how it works. Try then to create a similar program.

Good Luck!

*Programming is fun. Never give up. Ask for help!*

Farid Naisan,

Course Responsible and Instructor