

GEOG 490/590

GIS Programming

Lab 2: GUI Controls and Numerical Approximation

In this lab, you will design and develop a simple add-in that approximates the value of Pi. Pi or  $\pi$  is the ratio of a circle's circumference to its diameter in Euclidean geometry.

Alternatively  $\pi$  can be also defined as the ratio of a circle's area (A) to the area of a square whose side is the radius:

$$\text{Pi} = A / r^2$$

(Equation 1)

The numerical value of  $\pi$  truncated to 50 decimal places is:

3.14159 26535 89793 23846 26433 83279 50288 41971 69399 37510

BACKGROUND:

There are many ways to calculate the value of  $\pi$ . One of the more intuitive approaches is based on the second definition mentioned above. You draw an imaginary circle of radius  $r$  and a square inscribing the circle (Figure 1).

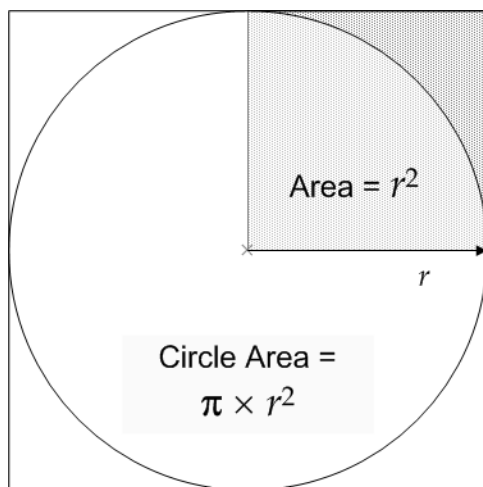


Figure 1

Finding a collection of points inside the square and the circle allows the square's area ( $4 * r^2$ ) and the circle's areas (A) to be approximated. Since the area A of a circle is  $\pi$  times the radius squared,  $\pi$  can be approximated by using Equation 1.

To implement this method, you:

1. Generate a huge number of random points (x, y) that fall within a, say, 1000 by 1000 (square unit) area. That is, the range of x and y values is between 0 and 1000.

2. Check any point (x,y) whose distance d from the origin is less than r (i.e., 1000). Those points will be inside the circle. The **distance** d is derived by the pythagorean theorem:

$$d = \text{sqr}(x^2 + y^2)$$

(Equation 2)

3. Pi is estimated using the following equation(s).

$$\text{Pi} = \#\_points\_with\_d\_less\_than\_r * 4 / \text{Total\_}\#\_of\_points\_generated$$

Or

$$\text{Pi} = 4 * (\#\_points\_with\_d\_less\_than\_r / \text{Total\_}\#\_of\_points\_generated)$$

The reason you multiply the numerator by 4 is because you only generate random points in the first quadrant of the circle.

## PART 1: CREATING THE GUI

1. Start by creating a new ArcGIS Desktop Add-In project for ArcMap. Include your last name when naming your Visual Basic project: for example, BrossLab2.
2. On the Welcome page, list yourself as the author.
3. Add a button to your add-in with the following required parameters. Be creative with the image that you choose for your button. Many icons come pre-installed with ArcMap at <ArcMap install directory>\ArcGIS\Desktop10.0\Bin\Icons. This button will be used to display the new GUI when it is pressed.

Class Name: BtnCalcPi

Caption: Calculate Pi

Category: GEOG 590 Add-In Controls (**this is important so the instructor can find your work**)

4. When the New Project Wizard has finished creating the project use File > Save All to save your project. Verify you are using the same name you used in step 1 when naming the project. Save your project in a folder in the C:\Users folder and regularly back up your data to your H:\ drive or portable device.

5. Add a DockableWindow to your project by right-clicking on the project name and selecting Add... New Item. In the New Item... dialog box select an ArcGIS desktop add-in component.
6. Name the component FrmCalcPi.vb
7. Select the Dockable Window component. This window will be your GUI. Set the following required parameters for your dockable window:

Designer Class: FrmCalcPi

Caption: Numerical approximation of Pi

8. Right-click on FrmCalcPi.vb in your project explorer and select Designer Mode. Add four TextBox controls, four Labels, and one Button.

The TextBox controls are for entering/displaying:

- (i) the number of points to be generated
- (ii) the radius of the circle
- (iii) the number of runs
- (iv) the approximate Pi value. This TextBox should be read-only as your program will be calculating the value

It is easier to write code for your controls when you use a consistent naming scheme:

- a. TextBox names should start with Txt
- b. Label names should start with Lbl.
- c. Button names should start with Btn

When the 'Run' button is clicked, your add-in will calculate the value of Pi. Please use labels to label the text boxes in your form as shown in Figure 2.

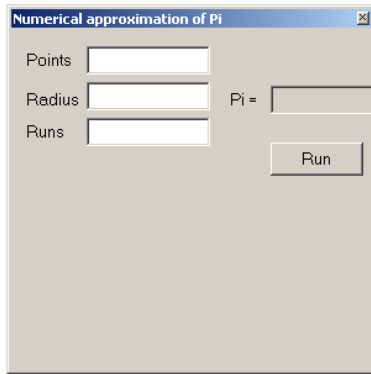


Figure 2

9. Copy and paste the following code into the `OnClick()` sub of the `BtnCalcPi` class. This code toggles the display of the dockableWindow when you click on `BtnCalcPi`.

Protected Overrides Sub `OnClick()`

```

    My.ArcMap.Application.CurrentTool = Nothing
    ' Declare the IDockableWindow class
    Dim dockWindow As ESRI.ArcGIS.Framework.IDockableWindow
    ' Declare the UID class
    Dim dockWinID As ESRI.ArcGIS.esriSystem.UID = New
    ESRI.ArcGIS.esriSystem.UIDClass()
    ' Set value property of UID to the form we wish to display
    dockWinID.Value = My.ThisAddIn.IDs.FrmCalcPi
    ' Get the dockable window containing the form frm the
    DockableWindowManager
    dockWindow =
    My.ArcMap.DockableWindowManager.GetDockableWindow(dockWinID)
    ' Toggle the dockable window visible/invisible depending on current
    setting
    dockWindow.Show((Not dockWindow.IsVisible()))
End Sub

```

10. Verify that the height and width of the `DockableWindow` in the `Config.esriaddinx` file match the height and width properties of `FrmCalcPi`.
11. Start ArcMap from inside Visual Basic Express using the debug functionality. Use the `Customize > Customize Mode...` dialog to add the `BtnCalcPi` command button to one of your toolbars. Click the button to verify that your form looks as expected.

## PART 2: IMPLEMENTING AND TESTING THE ALGORITHM

Here is the pseudo-code for the algorithm:

- Reset the pointsInCircle counter to zero
- Loop through the numberOfPoints
  - Generate a set of positive random numbers x and y whose values are smaller than the radius
  - Calculate the distance of x, y to the origin (i.e., 0, 0)
  - If the distance is shorter than the radius then increase the value of pointsInCircle by one
  - Next point
- $Pi = \text{pointsInCircle} * 4 / \text{numberOfPoints}$

To improve the accuracy and precision of the approximation, scientists usually average the results from multiple runs. You will need to incorporate another loop into your code so that the program can average the Pi values derived from multiple runs.

You will use the VB Random class to generate random numbers. The Random class has the option of returning an integer less than the maximum that you specify. Please look up the class in the online help for more information. This article on working with random numbers may also be helpful: [http://articles.techrepublic.com.com/5100-10878\\_11-5663283.html](http://articles.techrepublic.com.com/5100-10878_11-5663283.html).

Write code that implements the pseudo-code and add it to the Click event of the Run button on your form. Don't forget to verify that the user has entered valid values in all of the fields on your form.

## Questions

Q1) When creating a form, you can format the first control and then copy/paste additional controls of the same type to be more efficient (T/F)

Q2) What is an infinite loop? How could you accidentally create one in this program?

Q3) How do you set the name of the dockable window that appears in the top bar of the dockable window?

Q4) Experiment using different parameters in your new Pi calculator and fill the blanks in the table below. Describe how you might get a better approximation of Pi.

test #	radius	points	runs	pi
1	500	500	10	
2	500	500	100	

3	500	1000	10	
4	500	1000	100	
5	1000	500	100	
6	1000	1000	100	
7				
8				
9				
10				