

# CISS 160 Homework – Arrays A

The grade you earn will be based on a number of factors, including overall quality/correctness/validity of your application, following the file naming guidelines, following the compressed folder guidelines (see below and Syllabus), and following the other guidelines such as comments for your name, programming comments, test cases etc.

## General guidelines for each program:

> Include three comment lines within each program's Form1.cs file with your name, student id number, date, and goal/purpose of the program:

//Author: Your NAME

//ID: Your Student ID Number (NOT YOUR SOCIAL SECURITY NUMBER)

//Date:

//Goal-Purpose of the Program: (...your description...)

> within any program file where you write source code, include comments throughout your code describing in your own words, what the various sections of your code are doing; single line comments can be preceded by two forward slashes  
//COMMENT...

> for each program create a Text file in the folder where your project files are located and call it TestPlan.txt – edit that file for each program below to include a description of the different 'Tests' that you performed to give yourself assurance that your program is valid/is working correctly; this can include validating data to be numeric, validating that required data was entered, exception handling, other tests to verify specific calculations, functionality, verifying User Interface events etc.

## Write the C# programs described below:

> the requirements for each program will be described as a paragraph below OR the requirements will be represented as a compiled .exe of a program (note: this is not source code) that I include in the assignment folder so you can see a running version of how your program should work

> we will cover general aspects of various programming elements during lectures, but it is up to you to combine the lecture learning components along with the textbook reading to create a specific solution

> any images/graphics that are needed will be included in the assignment folder or I will provide in Angel

(see next page)

# CISS 160 Homework – Arrays A

1. (100 Points) Work with Arrays in a Program written in C#. Create a C# windows form that demonstrates how to work with Arrays. Be sure to refer to the general programming guidelines listed above. Name your project:

## **ArrayOfStudents - Your Name**

THINK: Building Blocks... get incremental parts of the logic working and then keep adding incremental functionality until you have it all done... break it down! AVOID the 'Big Bang' - trying to code it all-at-once approach!

**- Use the enclosed .exe as an example and create a program that looks and functions the same**

Create a program that will allow the entry of one or more student's names and course numeric grades; that also will display the information from the array in a Label on the form.

One student at a time, allow the user to enter a Student's Name and Course Numeric Grade and click the 'Add Student to Array' button. The Course Numeric Grade is to be entered as a number such as 100 (meaning 100 percent).

After entering up to 5 students, click on the Display Array button below to display the stored Student Names and Grade Numbers one line per student; then display a line with the total number of students, and calculate an average Grade across all the students entered.

### **A) ABOVE/OUTSIDE OF YOUR METHODS, AT THE CLASS LEVEL:**

- declare one array variable (string type) for the Student Names
- declare a second separate array variable for the Student Course Grades (decimal or int type)
- the maximum array size for both arrays is 5 (think: this means array index/subscripts of [0] thru [4])
- also keep a class level field int variable to track the current student number/array index (needs to be at the class level to retain its value after each Add button click)

### **B) Create a button to Add the Student data to the array variables:**

- when clicked, put the Student Name and Grade entered into the next open array index/slot/subscript for each array
- fyi after 3 students have been entered, for example, array slots [0] thru [2] should be populated
- popup a message box telling the user that the student data has been added to array index/slot/subscript [??] where ?? is the current array index value
- after each student's data has been added to the array, clear out the Student Name and Grade text boxes on the form
- if the array has been filled to it's maximum number of array elements, do not add any more data to the array, and popup a message box to the user stating that the array is full
- validate that a Student Name is entered; validate that the Course Numeric Grade is entered, numeric, less than or equal to 100

### **C) Create a button to Display the Data from the Array in a large Label field on the Form:**

- when clicked, iterate/go through each array starting at array index/slot/subscript [0] and going through the last array index [4] and retrieve each array's data stored during the data entry process above
- see the .exe example, but basically display one line per array entry showing the stored Student Name and Grade
- after all students have been displayed, show a Total line with the number of students entered and the average Grade for the students entered (think calculating the average may require explicit casting of an int variable to a decimal)

### **D) Create a button to Clear everything:**

- 1) Clear the Textboxes and Labels
- 2) Clear out the Arrays
- 3) Reset the current student counter to zero so that a new group of student names and grades can be entered

Take all of your completed project folders above and copy them into a folder named:

## **ArraysA - Your Name**

Zip up this folder and submit the zip file in this dropbox by the due date and time.