

COP2360

Assignment #3

Due at start of class, Monday, February 27

- A. (**20 points**) Print isosceles triangles. For each triangle, allow the user to input two values: a character to be used for printing the triangle and the size of the peak for the peak for the triangle. Test the input for valid characters. The size of the triangle should be larger than 10. If an invalid non-numeric character is entered for size or if the value entered for size is larger than 10, use 3 as the default value. If an invalid entry is entered for the character, use an asterisk (\*) as the default character. Allow multiple triangles to be printed. For example, if user inputs # for the character and 6 the peak, you should produce the following display:

The image consists of three vertically stacked screenshots of a terminal window. The top screenshot shows the program's instructions: "You will be prompted to enter a character to be used for printing an isosceles triangle AND an integer between 1 and 10. The integer will be the length for the two sides of the isosceles triangle. Whatever character is entered will be the character used for printing. You will be able to enter as many sets as you wish." Below this, a message says "Press any key when you are ready to begin...". The middle screenshot shows the user input: "How large of an isosceles Triangle do you want to print? Please enter a positive integer (between 1 and 10) : 6 What character do you want to use for printing? #". The bottom screenshot shows the resulting output: an isosceles triangle made of '#' characters with a peak size of 6. At the bottom of the terminal window, there is a prompt: "Would you like to enter a new character and size? Please enter 'y' for yes or any other letter for no".

- B. (**40 points**) Create three arrays of type **double**. Do a compile-time initialization and place different values in two of the arrays. Write a program to store the product of the two arrays in the third array. Produce a display using the *MessageBox* class that shows the content of all

three arrays using a single line for an element from all three arrays. Design your solution so that the two original arrays have a different number of elements. Use 1 as the multiplier when you produce the third array.

Program sample output:

Parallel Arrays		
Array1 * Array2 = Array3		
Array1	Array2	Array3
1	22	22
2	32	64
3	14	42
4	54	216
5	23	115
6	71	426
7	234	1638
8	63	504
9	91	819
10	11	110

- C. (40 points) Write an application that creates a two-dimensional array. Allow the user to input the size of the array (number of rows and number of columns). Fill the array with the random numbers between 0 and 100. Search the array for the largest value. Display the array values, numbers aligned, and the indexes where the largest value is stored.

Program sample output:

```
How many rows do you want to create? 5
How many columns do you want to create? 6

Random Generated Values

Col      1      2      3      4      5      6
Row 1    55     51     55     33     90     21
Row 2    11     58     89     3     4     17
Row 3    35     89     89     90     47     25
Row 4    18     18     41     34     1     30
Row 5    55     70     13     39     88     16

Largest value: 90
Row 3
Col 2
```

### Important notes:

1. Total points: 100
2. Automatic 50 points deducted for files that do not work or produce incorrect outputs.
3. Put your name, source files names, and program description at the top as comments.
4. Submit C# source files (*IsoscelesTriangleApp.cs*, *ParallelArrayApp.cs*, *RandomValuesApp.cs*).
5. Email me your questions at *hardanis@palmbeachstate.edu*.