CPE 150 Laboratory 11: Arrays III

Department of Computer Engineering
Yarmouk University

Summer 2017

1 Objectives

- To declare arrays, initialize arrays and refer to individual array elements.
- To be able to pass arrays to functions.
- To understand basic sorting techinques.
- To understand basic searching techinques.
- To be able to declate, initialize and manipulate multiple-subscript arrays.

2 Lab Exercise 1 - 2D Array Operations

Write a program that creates a two-dimensional array initialized with test data. Use any data type you wish. The program should have the following functions:

- `getTotal`: This function should accept a two-dimensional array as its argument and return the total of all the values in the array.
- `getAverage`: This function should accept a two-dimensional array as its argument and return the average of all the values in the array.
- `getRowTotal`: This function should accept a two-dimensional array as its first argument and an integer as its second argument. The second argument should be the subscript of a row in the array. The function should return the total of the values in the specified row.
- `getColumnTotal`: This function should accept a two-dimensional array as its first argument and an integer as its second argument. The second argument should be the subscript of a column in the array. The function should return the total of the values in the specified column.
- `getHighestInRow`: This function should accept a two-dimensional array as its first argument and an integer as its second argument. The second argument should be the subscript of a row in the array. The function should return the highest value in the specified row of the array.
- `getLowestInRow`: This function should accept a two-dimensional array as its first argument and an integer as its second argument. The second argument should be the subscript of a row in the array. The function should return the lowest value in the specified row of the array.
- `getTranspose`: This function computes the transpose of the two-dimensional array. There should be a `printTranspose` to print the output of the new transposed two-dimensional array.

Demonstrate each of the functions in this program. Your program should not use any global variables.

3 Lab Exercise 2 - Sales Person

Use a double-subscripted array to solve the following problem. A company has four salespeople (1 to 4) who sell five different products (1 to 5). Each salesperson passes in slips for each different type of product sold. Each slip contains the following:

- The salesperson number.
- The product number.
- The total dollar value of that product sold that day.

Assume the multiple slips are available. Write a program that reads all this information and summarize the total sales by salesperson by product. All totals should be stored in the double-subscripted array `sales`. After proessing all the sales information, print the results in tabular format with each of the columns representing a particular product and each of the rows representing a particular salesperson. Cross total each row to get the total sales of each product; cross total each column to get the total sales by salesperson. Your tabualr printout should indicate these cross totals to the right of the totaled rows and to the bottomm of the totaled columns. Your program should not use any global variables and the must be modularized to get all totals, input from user, output to screen.

```
Enter the salesperson (1 - 4), product number (1 - 5) and total sales. (-1 to end)
1 1 9.99
3 3 5.99
2 2 4.99
-1

The total sales for each sales person are displayed at the end of each row, and
the total sales for each product are displayed at the bottom of each column,
         1         2         3         4         5         Total
1      9.99      0.00      0.00      0.00      0.00         9.99
2      0.00      4.99      0.00      0.00      0.00         4.99
3      0.00      0.00      5.99      0.00      0.00         5.99
4      0.00      0.00      0.00      0.00      0.00         0.00

Total  9.99      4.99      5.99      0.00      0.00
```

4 Postlab Exercise

Write a program that allows two players to play a game of tic-tac-toe. Use a two-dimensional char array with three rows and three columns as the game board. Each element of the array should be initialized with an asterisk (*). The program should run a loop that:

- Displays the contents of the board array.
- Allows player 1 to select a location on the board for an X. The program should ask the user to enter the row and column number.
- Allows player 2 to select a location on the board for an O. The program should ask the user to enter the row and column number.

- Determines whether a player has won, or a tie has occurred. If a player has won, the program should declare that player the winner and end. If a tie has occurred, the program should say so and end.

Player 1 wins when there are three Xs in a row on the game board. The Xs can appear in a row, in a column, or diagonally across the board. A tie occurs when all of the locations on the board are full, but there is no winner.