Assignment 4

F2021

Multi-Dimensional Arrays

Complete the project described below.

Create a menu-driven program that allows the user to select some data from the results from a series of races. Use a two dimensional array to store the results, and a parallel array to store the competitor names (because it's a different data type). Here is some sample data from the four-man bobsleigh event from the 2018 Olympics (in order of their final ranking)[1]. **You may use any other data as long as it can be presented in a grid and the calculations below can still be carried out**.

Team	*Run 1*	*Run 2*	*Run 3*	*Run 4*	*Total*
Germany	48.54	49.01	48.76	49.54	
Germany	48.74	49.16	48.90	49.58	
South Korea	48.65	49.19	48.89	49.65	
Switzerland	49.05	49.16	48.87	49.51	
Latvia	48.82	49.39	48.91	49.53	
Canada	48.85	49.28	48.95	49.61	
Austria	49.10	49.21	49.03	49.56	
Germany	48.95	49.26	49.10	49.80	
United States	49.09	49.26	49.08	49.77	
Latvia	49.18	49.26	49.34	49.63	
Average					

Technical Requirements

You must use the special version of the "for each" loop designed for iterating through arrays at least once in your solution.

You must use pointer arithmetic at least once in your solution.

You may hard code the data.

Menu Validation

The menu will validate all entries. If invalid menu items are entered the program will give an appropriate message and reprint the menu.

Menu Actions

The menu allows the user to get:

- One race result for a team
    - After menu choice the user will need to enter two coordinates
    - Output will be one value (ex. Canada Run 3: 48.95)

[1] Wikipedia 15-Oct-2019:
https://en.wikipedia.org/wiki/Bobsleigh_at_the_2018_Winter_Olympics_%E2%80%93_Four-man

- All the results from one team
    - After menu choice the user will need to enter one value representing the team they wish
    - Output will be a list of values (ex. Latvia: 48.82, 49.9, 48.91, 49.53)
- All the results from one race
    - After menu choice the user will need to enter one value representing the race they want
    - Output will be a list of values (ex. Run 2: 49.01, 49.16, 49.19, 49.16, 49.39, 49.28, 49.21, 49.26, 49.26, 49.26)
- The average of one race
    - After menu choice the user will need to enter one value representing the race they want
    - Use stream manipulators to show only two decimal places.
    - Output will be one number (ex Run 3: 48.98)
- The total for any team across the entire series
    - After menu choice the user will need to enter one value representing the team they want
    - Output will be one number (ex. for Latvia it would be 197.41)

Testing

Make sure that all of your code has been tested

- Every method in your code should be tested at least once.
- Remember when testing ranges of values to test above and below the allowable range, and exactly at the end points of the allowable range.

Style

Your program must be neat, well formatted, and readable (see the style guide on iLearn). Remember you can lose up to 30% for poor style. Don't forget comments that identify the programmer, the program, and the date the program was written.