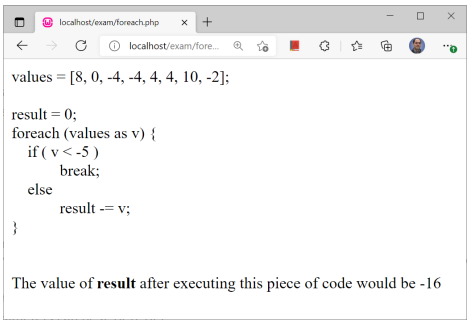
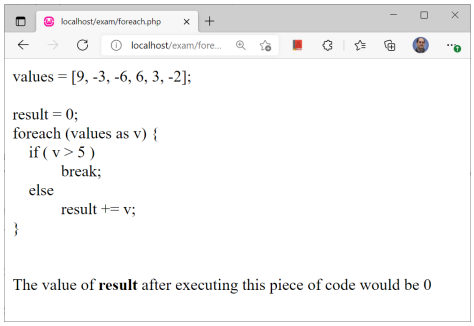
CSIS 3280 Midterm Exam- Practical October 29, 2021

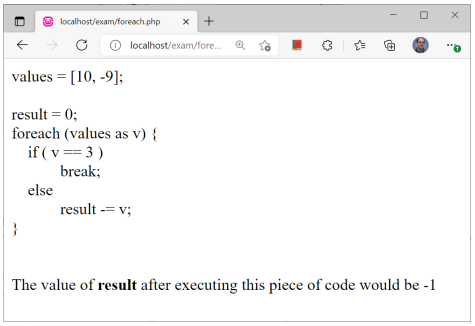
1. Complete work on your computer using Visual Studio or any other editor you choose.
2. You shall not collaborate or share your work with anyone in or out of the class. **Any student** (the programmer or the copier) taking part in any form of dishonesty will get a zero and reported to the Dean.
3. Create a folder called xy where x is your first name and y is your last name. Save all your files in this folder.
4. Make sure you regularly save your work. (Pay attention to the point 7 below!). **You must submit your progressing work to Blackboard every 40 minutes.**
5. After you complete the task, close all the files, then compress the folder xy and upload **xy_final.zip**. You can compress the folder by right-clicking the folder, and hit the Send To option, and select compressed (zipped) folder.
6. MAKE SURE YOU SUBMIT on blackboard (and double check your submission by downloading and extracting the files to check if everything works).
7. **You are completely responsible for the submission on blackboard. Any error in submission or compression will result in a ZERO.**
8. If you did not submit every 40-minutes or any other suspicious activity is detected, you will be invited for an interview and/or oral exam.

Random Foreach Loop Generator

You are asked to write a PHP program that generates a piece of code randomly, containing a random array and a random foreach loop. The code evaluates the generated code and displays to the user what the result would be if the piece of code executed.







Requirements

1. You must write your program as requested below. It must conform with the following requirements. If something has not been explicitly requested, you can choose your own style or the most efficient choice.
2. The name of the array is always fixed, and it is **values**. Therefore, all randomly generated arrays have the same name.
3. The number of elements of the array is random. It can at least 1 element to maximum 10 elements. The values are generated randomly of integers from -10 to 10.
4. After generating the array, there is always the statement **result = 0 ;**
5. Then the foreach loop is generated. There is always **foreach (values as v)**.
6. Inside the foreach loop, there is an if-else statement.
7. The condition (Boolean expression) in front of if-keyword is generated randomly. A comparison operator (>, <, >=, <=, ==, !=) is picked randomly among the list and the integer value after the comparison operator is generated randomly. This integer value can be from -8 to 8.
8. In the body of if-statement is always **break;**
9. The else-statement has always 1 expression using either += or -= operators. This operator must be picked randomly.
10. After the array and foreach loop definition, your program must display what the value of result would be if the piece of code executed.
    a. define a function called **evaluate_foreach_loop.** This function has 4 input parameters:
        i. The array of randomly generated values (9,-3,-6,6,3,-2 in the example top-middle)
        ii. a string indicating the comparison operator in front of if-keyword (in the first example above is '<')
        iii. the value after the comparison operator. (-5 in the first example on top-left)
        iv. the compound operator in the expression of the else-statement ('-=' in the first example, and '+=' in the second example above.
    b. The function returns 1 value: the evaluated value of result if the code executed.
    c. You may use two loops in this function. First loop to start from the beginning of the array to the point in the array where an element violates the condition of the Boolean expression in front of if-keyword. For example, in the middle example above 9 violates the condition (v > 5), and in the third example above, none of the elements (10 and -9) does not violate (v==3).
    The second loop calculates either += or -= on the elements of the array from the beginning of the array to the element that you found in the first loop.
    d. You must also define a function **evaluated_bool_expression**. This function is used in the first loop to evaluated if the condition in front of if-keyword is true or false
        i. This function has 3 inputs and returns 1 Boolean value (true or false)

ii. The 3 input parameters are: the value of an element of the array, comparison operator, and the integer value after the comparison operator.
iii. You must use a switch-case to find what the comparison operator is and if the boolean expression is evaluated to true or false.

11. For formatting you can use **
** for line break and ** ** for spaces. For example, define consts like the following and use them in your strings where you want to create a new line or :
    - const white_4space = "   ";
    - const new_line = "
";

Submission

Submit a zipped folder including all your files to the corresponding folder on Blackboard. Make sure you submit correct files and a valid zipped file. **Note the point 7 on the first page of this exam.**

Screenshots