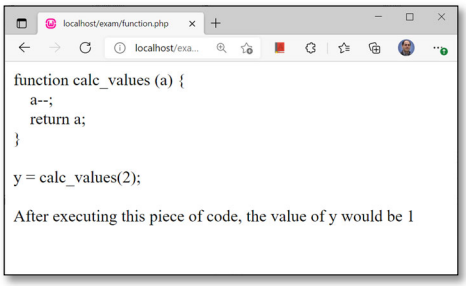
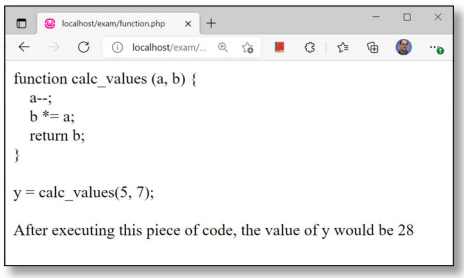
CSIS 3280 Midterm Exam- Practical October 29, 2021

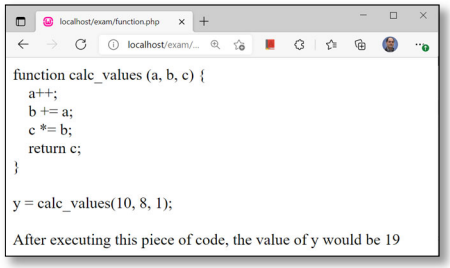
1. Complete work on your computer using Visual Studio or any other editor you choose.
2. You shall not collaborate or share your work with anyone in or out of the class. **Any student** (the programmer or the copier) taking part in any form of dishonesty will get a zero and reported to the Dean.
3. Create 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 rename it to **xy_final** and upload **xy_final.zip**. If your computer is Windows, 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.
9. You must stay on Zoom meeting and keep the speaker of your computer on. I may announce some changes, clarifications, or explanations.

Random Function Statements Generator

You are asked to write a PHP program that generates a piece of code randomly, containing a function definition and function invoking and then finds out what the value of y 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 function is always fixed, and it is calc_values. Therefore, all randomly generated functions have the same name.
3. The number of the function input parameters are random. It can at least 1 parameter to maximum 3 parameters. The name of the parameters can be a, or b, or c.
4. If the function has 1 input parameter, it has 1 statement (plus return statement). If the function has 2 parameters, the function has 2 statements (plus return), and if there are 3 statements, the function has 3 statements, plus return
5. The first statement of the function is either increment(++) or decrement(--) of the first parameter (i.e., a)
6. **UPDATED:** The second statement is an expression of the ~~first~~ second parameter (~~a~~b) on the left and the ~~second~~ first parameter (~~b~~a) on the right. The operator can be +=, -=, *=, or /=, selected by your program randomly.
7. **UPDATED:** The third statement is like the second statement, but the second parameter (b) is on the ~~left~~ right and the third parameter (c) on the ~~right~~left. The operator is randomly selected one of +=, -=, *=, or /=
8. **UPDATED:** The final statement is always a return statement. It returns the variable on the left of the last expression (a or b or c depending on the number of the input parameters as you can see in the examples above).
9. After the function definition, there is one statement to call the function and pass values to the parameters and store the returned value in the variable y.
10. The number of values passed to the function must match with the number of parameters. For example, if there are 3 parameters in the function definition, 3 values must be passed to the function. The values are chosen randomly between 0 and 10.
11. After generating randomly, the function definition and invoking the function, your program must display what the value of x would be if the piece of code executed.
    a. define a function called **evaluate_function.** This function has 3 input parameters:
        i. an array containing the value of the arguments passed to the function (3,8,3 in the example top-right)
        ii. a string indicating if the operator of the first statement is increment (++) or decrement(--) (in the first example above is '- -')
        iii. an array containing two operators' symbols, the operator of the second and third statements (*= and /= for the example on top-right)
    b. the function returns 1 value: the evaluated value of x if the code executed.
    c. You must use if-else statements in the body of this function to determine what the number of function's input parameters are then based on the operators calculate the expressions accordingly.

d. First you check if ++ or – is chosen for the first statement in the body of function, apply operation to the value of the first argument, then check which operator chosen in the second statement (+= or -= or …), apply the operation to the values in the array accordingly, and the same for the third statement.
e. At the end, return the evaluated value.
f. NOTE: you must use at least 1 switch-case in this function.

12. 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