

CS 112 Assignment 7a

Submit your finished code to the Dropbox. You will need to pass off the assignments with the TA or tutor during their office hours (their information is in the CS 112 Course Information section of the Table of Contents), or the instructor in class. Follow the formatting guide which is also found in the CS 112 Course Information section.

Read the instructions carefully. Make sure your output matches the example run.

Objectives:

In this assignment you will learn how to do the following:

- Call a function from main()
- Call a function from another function (not main())
- Understand that the variables inside a function are different from variables in another function even if they are named the same.
- Print from inside a function other than main()

Program: functions `make_even()` and `add_numbers()`

Write a program that asks the user to enter three whole numbers. Use input validation to ensure the user entered three valid numbers. You may use either `.isdigit()` or `try-except` to perform input validation on the numbers.

Have your `main()` function call a second function that you write called `add_numbers()`, passing the three numbers as arguments. Your `add_numbers()` function will check to see if each number is even. If a number is not even, your `add_numbers()` function will call another function you write called `make_even()` which simply adds 1 to the odd number and returns the result to the calling function. Print out the three modified numbers and the sum of these number FROM YOUR `add_numbers()` function as shown in the example run.

After the program has finished, ask the user if they would like to play again. Use input validation to ensure the user answered with a 'Y' or 'N'. Keep the program running in an infinite loop until the user indicates they want to quit.

Key program requirements:

- Do input validation on all three numbers and the 'Y'/'N' play again question
- Write a function called `add_numbers()` which is called from `main()`
- Write a function called `make_even()` which is called from `add_numbers()`
- Use of global variables is NOT allowed
- The names of the three variables inside `main()` which are passed to `add_numbers()` as arguments **MUST BE THE SAME** as the names of the three parameters inside of your `add_numbers()` function

Technical Design:

The main() function will:

- Ask the user for three numbers and validate they are whole numbers
- Call the add_numbers() function passing the three numbers the user entered as arguments
- Repeat the program until the user wants to quit

The add_numbers() function will:

- Accept three numbers as parameters
- Evaluate each number to determine if it is odd or even
- If a number is odd, call the make_even() function passing the odd number as an argument
- Add the three (possibly modified) numbers together
- Print the result as shown in the example

The make_even() function will:

- Accept one number as a parameter
- Add 1 to it
- Return the result

Ponder – Discuss the following with your partner. Failure to do this activity and discuss these questions before you attempt to pass off your program will result in a failed pass off.

- On a separate piece of paper, draw boxes that represent the various memory locations that will be used by your program. MAKE SURE that you indicate which function the variables belong to.
- With your partner, use your finger or a pen to point at each line of code as you trace through the program from beginning to end. Yes, really use your finger or a pen.
- Step through each line IN THE ORDER THAT IT WILL BE EXECUTED from beginning to end.
- Make assumptions of what numbers the user will enter, then write those numbers in the appropriate memory locations as you trace through the code with your finger.
- Show the paper with your memory locations to the TA or instructors when you pass off the program.
- **DISCUSS** why your instructor required you to use the same names for both the variables inside of main() and the parameters inside of add_numbers().
- **DISCUSS** why you were instructed to print the result from inside of add_numbers() instead of main().

Example Run

This program will ask the user for 3 numbers.

If the user enters an odd number, the program will add 1 to that number to make it even.

Once all three numbers are even, the program will add them together and print the result.

Please enter the first number: ok

Invalid response. Please enter a whole number.

Please enter the first number: 4

Please enter the second number: 5

Please enter the third number: 6

$4 + 6 + 6 = 16$

Would you like to play again? (Y/N): y

Please enter the first number: 3

Please enter the second number: 5

Please enter the third number: 7

$4 + 6 + 8 = 18$

Would you like to play again? (Y/N): n

>>>