CS1050 – Lab 6

Spring 2020

Concepts to Practice

- Pointers
- Simulated "pass by reference" via pointers
- Strings
- Expand the prelab

Submission Information

Submit this assignment by following the instructions below. SUBMIT ONLY the .c file (no a.out or executable file is required). You must submit your lab by Friday, March 20, 2020 at 9 PM!

Use the following submit command:

```
mucs submit <class> <assignment_type> <filename>
```

For example:

```
mucs submit 1050 hw x-lab6.c
```

Filename must be: *sectionletter-lab6.c* (include your respective lab section, e.g., a-lab6.c).
Note that the command is "mucs submit 1050 hw …". All "labs" will now be submitted as homework (hw) so that you will have the full week to complete them.

Description

For the lab assignment, you need to only write two functions in addition to main. One function will get input from the user. The other function will print out the data that was received from the user. Both of these functions must have return type void.

Your main() function will:

- Declare any variables whose values will be input by the user.
- Print out a welcome message.
- Call GetInput() to get a string and an array of positive integers from the user.
- Call PrintOutput() to print out the string and the integers that you received from the user.

It is up to you how you will prototype these two functions. Your main() function should only declare variables, call printf(), and call the GetInput() and PrintOutput() functions. You may write other functions if you wish, but you may only call GetInputJ() and PrintOutput() from your main().

You may assume that no string will be longer than 256 characters (including the null-terminator). You may also assume that the integer array will have no more than 256 elements.

Functions

You may not use global variables (using global variables will result in a zero).

- **void GetInput()** – You should create an appropriate prototype for this function (it may have parameters) but it must be of void return type. This function requests a string and an array of positive integers from the user. You may use -1 as a sentinel value to cease inputting integers.

- **void PrintOutput()** – You should create an appropriate prototype for this function (it may have parameters) but it must be of void return type. This function prints out the string and the array which were entered by the user via the GetInput() function.

Sample Output

```
jimr@JimRHadesCanyon:~/CS1050/SP2020/labs/lab6$ compile x-lab6.c
jimr@JimRHadesCanyon:~/CS1050/SP2020/labs/lab6$ ./a.out

***********************
* Welcome to Lab 6    *
***********************

Please enter a string: CS1050IsFun(no,really)

Please enter positive integers, and -1 to end
        element #0: 999
        element #1: 123
        element #2: 15
        element #3: 1
        element #4: 2
        element #5: 3
        element #6: 4
        element #7: -1

***********************
The string you entered:
CS1050IsFun(no,really)
Array elements you entered:
        array[0]=999
        array[1]=123
        array[2]=15
        array[3]=1
        array[4]=2
        array[5]=3
        array[6]=4
```

Guidelines for Grading Lab 6
40 Points Possible

General

If your program does not compile or produce any input/output (I/O) because most of the source code is commented out then your lab will receive a grade of ZERO POINTS. Further, if your program does not actually follow the specifications, but merely prints out lines that make it appear to follow the specifications, you will receive a grade of ZERO POINTS. For partial credit your C program must not only compile but also produce some valid I/O that meets the lab specifications.

You program is expected to have a comment header at the top that includes your name, pawprint, the course you are taking, and the lab that you are solved (e.g., "Lab 6"). Your code should be nicely indented. **You will lose up to 10 points if you do not meet these basic requirements**.

If you do **ANYTHING** in your main() other than declaring variables, calling printf(), or calling one of the specified functions that you will write, you will lose 50% of the points you would have otherwise received. If you have any global variables, you will get zero points.

10 points: GetInput() correctly prompts for and receives a string.
10 points: GetInput() correctly prompts for and receives an array which is terminated by a -1.
10 points: PrintOutput() correctly prints the string that was entered and the array that was entered.
10 points: The output very closely matches the sample given.