

ENGG1340 / COMP2113

Problem Solving Session 2 - Functions and Array Basics

Notes to Student TAs:

- It is important to show the coding process, which includes debugging, so there is no problem that you make a mistake. Sometimes it's good for students to know how to trace a bug too.
 - It'd be good to let students see how you break down a problem and solve smaller problems (and make sure that they work) one at a time.
 - Coding style (i.e., indentation & proper comments) is important too.
-

Problem 1

Write a `void` function `OrderThreeIntegers()` that takes three `int` arguments by reference. The function should determine the order of the values, and upon return of the function, it should modify the values in the arguments so that the first argument contains the largest value, the second the second-largest, and the third the smallest value.

- You may need another function to swap the values of two `int` arguments
 - Write a simple driver program (i.e., `main()`) to test your functions. You should print out the values before and after calling `OrderThreeIntegers()` to show it works correctly.
-

Notes to Student TAs:

Concepts to reinforce here:

- modularity by defining functions which serve specific purposes (or solve a smaller problem) in a program
- how a function prototype is defined based on the function purpose: return types and input arguments
- pass-by-value & pass-by-reference

Question: how many cases do you need to test if your program is correct?

Answer: the input of 3 numbers can assume $3! = 6$ permutations (or orders)

e.g., `1 2 3`, `1 3 2`, `2 1 3`, `2 3 1`, `3 1 2`, `3 2 1`

they all should result in an output `3 2 1`

Problem 2

Write a program to read in a sequence of integers in the range from 0 to 10, and to find the mode of the input numbers (i.e., the number which occurs most frequently).

- write a function `int FindMax(int a[], int aSize)` which returns the index of the largest element in the array `a[]`. The size of `a[]` is given by `aSize`.
 - if multiple elements have the same largest value, return the largest index of these elements.
- write the `main()` function of the program:
 - define an array `freq[]` of 11 integers; each element will store the frequency of occurrences of an integer. For example, `freq[i]` stores the frequency of the number `i` in the input sequence.
 - read a sequence of integers from user input which is terminated by a sentinel value `-1`. For each input integer, increment the corresponding frequency counter in the array `freq[]`
 - outputs the frequencies of all numbers from 0 to 10, and output the mode of the input numbers.

Notes to Student TAs:

Concepts to reinforce here:

- array as function parameter
- loop through an array for some processing

Question: Modify `FindMax()` so that if multiple elements have the same largest value, it returns the smallest index of these elements.

Answer: change line 15 from `if (a[i] >= a[maxIdx])` to `if (a[i] > a[maxIdx])`