02.Functions, Arrays and STL Vectors – Homework Exercises

Write C++ code for solving the tasks on the following pages.

Code should compile under the C++03 or the C++11 standard.

Please submit a single `.cpp` file for each task.

`.cpp` files for the tasks should be named with the task number followed by what you feel describes the exercise in a few words.

E.g. a good name for task 2 of this homework would be:
`2.longest-sequence.cpp`

Don't worry about the name too much, just make sure the number and the file extension are correct.

Problem 1 – Compare Arrays

Write a program that reads two integer arrays from the console and compares them element by element. For better code reusability, you could do the comparison in a
`bool areEqual(int arr1[], int length1, int arr2[], int length2)`
function, which returns **true** if they are equal and **false** if not.

Each array will be defined by two lines on the console – the first containing a number representing the length of the array, and the second containing the numbers in the array, separated by spaces.

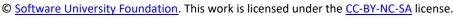Print **equal** if the arrays match, and **not equal** if the arrays don't match.

Examples

Input	Output
3 1 2 3 3 1 2 3	equal
3 1 2 3 3 2 1 3	not equal
2 10 1 3 42 13 69	not equal

Problem 2 – Longest Sequence

Write a program that finds the longest sequence of equal elements in an integer array and then prints that sequence on the console (integers separated by space on a single line). If there is more than one such sequence, print the last one. The input array will be entered on two lines – the first line will contain an integer representing the number of elements, the second will contain the elements separated by spaces

Examples

Input	Output
7 13 10 10 1 4 2 10	10 10
5 13 42 19 21 103	103
9 1 1 2 2 2 4 -1 3 3	2 2 2

Problem 3 – Above Average

Write a program that read an array of integer numbers from the console and prints all numbers which are larger than or equal to the mathematical average of the numbers in the array. The output should be printed on a single line, separating the output number by spaces. The output numbers should be in the same order as they were in the input.

The input array will be entered on two lines – the first line will contain an integer representing the number of elements, the second will contain the elements separated by spaces.

Examples

Input	Output	Explanation
5 1 2 3 4 5	3 4 5	Average = (1 + 2 + 3 + 4 + 5) / 5 = 3, only 3, 4 and 5 are larger than or equal to 3
1 42	42	The average of a single number is that number
7 -1 12 105 -105 1 0 9	12 105 9	Average = (-1 + 1 – 105 + 105 + 0 + 9 + 12) / 7 = 21 / 7 = 3, both 12 and 9 are larger

Problem 4 – Most Frequent Number

Write a program that finds the **most frequent number** in a given sequence of numbers.
- Numbers will be in the range [0…9].
- In case of multiple numbers with the same maximal frequency, print all of them, ordered from smallest to largest, separated by spaces

Examples

Input	Output	Output
13 **4** 1 1 **4** 2 3 **4 4** 1 2 **4** 9 3	4	The number **4** is the most frequent (occurs 5 times)
8 **2 2 2 2** 1 **2 2 2**	2	The number **2** is the most frequent (occurs 7 times)
11 **7 7 7** 0 2 2 2 0 **9 9 9**	2 7 9	The numbers **2, 7** and **9** have the same maximal frequence (each occurs 3 times).

Problem 5 – Cartesian Product

Write a program which reads an array from the console and prints the product of each of its elements with all elements. E.g. for the array {1, 7, 3}, the result would be {1 * 1, 1 * 7, 1 * 3, 7 * 1, 7 * 7, 7 * 3, 3 * 1, 3 * 7, 3 * 3}, which gives us the array {1, 7, 3, 7, 49, 21, 3, 21, 9}, so for the input
1 7 3
the program should print
1 7 3 7 49 21 3 21 9

Examples

Input	Output
3 1 7 3	1 7 3 7 49 21 3 21 9
2 -1 4	1 -4 -4 16
1 42	1764

Problem 6 – Closest Numbers

Write a program which finds the two closest (by value) integer numbers in an array and prints the absolute difference between them.

Examples

Input	Output	Output
5 1 105 10 100 3	2	The closest numbers are 1 and 3, abs(1 – 3) = 2
9 1 2 3 4 5 6 7 8 9	1	All numbers are exactly 1 unit apart
5 -1 -105 -10 -100 3	4	The closest numbers are -1 and 3, abs(-1 – 3) = 4, or equivalently abs(3 – (-1)) = abs(3 + 1) = 4