Offline (Home Assignment) on 1D Array & Function

1. Write a C program that will take a non-negative integer *n* followed by *n* integers as input. You have to print the most frequent number among the given integers continuously while taking input. When there are multiple such numbers, you have to print all of them but you can print in any order. You can safely assume that all the given integers will be in [-10000, 10000] range.

Sample Input	Sample Output
5 3 1 1 5 5	Most frequent number = 3 Most frequent number = 3, 1 Most frequent number = 1 Most frequent number = 1 Most frequent number = 1, 5
4 -50 -50 10 20	Most frequent number = -50 Most frequent number = -50 Most frequent number = -50 Most frequent number = -50

2. A perfect number is a positive integer which is equal to the sum of its proper divisors (i.e. divisors excluding the number itself). For example, 6 is a perfect number since its proper divisors are 1, 2, 3 and their sum is 6. Now write a C program that will take a non-negative integer *n* as input and print the sum of perfect numbers upto *n*.
   You must write a C function to check if a number is perfect or not. This function should receive an integer as parameter and return 1 if the number is a perfect number and 0 otherwise.

Sample Input	Sample Output
4	0
10	6
30	34
50	34

3. Write a C program to print a bar chart with *. You have to take a positive integer *n* as input followed by *n* non-negative integers. Now considering each of these integers as height of the bars, print them with *. The width of each bar should be 2 asterisks (*) and two consecutive bars should be separated by a space. See sample input-output for further clarification. You can safely assume no input will be greater than 100.

Sample Input	Sample Output

4
2 1 6 3
	`        **`
	`        **`
	`        **`
	`        **  **`
	`**      **  **`
	`**  **  **  **`

2
1 4
	`    **`
	`    **`
	`    **`
	`**  **`

3
4 10 7
	``
	    **
	    **
	    **
	    ** **
	    ** **
	    ** **
	** ** **
	** ** **
	** ** **
	** ** **
	``

Submission Guideline:

i. Create a new folder in your local machine and rename it with your **7 digit student id** (for example, 1905001).
ii. Put the **source files (i.e. only the .c files)** in this folder (created in step i).
iii. Now **ZIP** the folder. The name of the zip file should look like 1905001.zip.
iv. Submit the .zip file at the offline submission link created on Moodle course page of CSE 102.

Special Instruction:

i. Please do not copy solutions from anywhere (your friends, seniors, internet etc.). If you do, you will get caught and will be given severe punishment.
ii. You should be able to explain your code during evaluation. Failure to do so may result in deduction of full marks.
iii. Try to write readable, re-usable, well-structured code with suitable comments. This will help you in the long run.

Submission Deadline: 10 October 2020 (Saturday) 8:00 pm