CSCE221 Lab: Multimaps

Question:
Given an integer array nums and an integer k, return *the* k *most frequent elements*.

k always lies in the range [1, number of unique elements in the array given]

Example:

Input:
nums = [1,1,2,2,2,3]
k = 2
Output: [1,2]
1 occurs 2 times, 2 occurs 3 times, 3 occurs only once. The 2 most frequent elements are 1 and 2.

Input:
nums = [1]
k = 1
Output: 1

Reference:

Refer to this link below for further details on std::multimap
https://www.cplusplus.com/reference/map/multimap/multimap/

Task: Complete the function in the multimap.cpp file. Refer to the hint/idea given below.

Idea:

(a) Implement using std::map and std::multimap. Store each number and its corresponding frequency as key value pairs in a map -> **std::map<int, int>**
(b) Use **std::multimap<int,int,greater<int>>** to store the key-value pairs {f:n1,f:n2….} and so on. ***f*** is the frequency and value will be the numbers which correspond to that frequency. This utilizes the concept of multimap(same key can be mapped to different values).
(c) The greater<int> part in the multimap declaration is the std::greater. It is a functional object which is used for performing comparisons. Your multimap will be sorted based on the keys in descending order.
(d) Iterate through this sorted multimap to fetch the top k frequent elements.