Ассоциативные контейнеры.

Если тип элементов контейнера не указан, то предполагается, что элементами являются целые числа.

Множества. Теоретико-множественные алгоритмы

(set, multiset)

1. Дан вектор V с четным количеством элементов. Если все значения, содержащиеся во второй половине вектора, входят хотя бы один раз в его первую половину, то вывести true, иначе вывести false. Использовать алгоритм includes, применив его к двум множествам (контейнерам типа set), созданным на основе вектора V.
2. Дана строка name и вектор V с четным количеством элементов. Найти все различные числа, которые одновременно входят и в первую, и во вторую половину исходного вектора, и записать их в текстовый файл с именем name в возрастающем порядке, добавляя после каждого числа символ пробела. Использовать алгоритм set\_intersection для двух вспомогательных множеств и итератора ostream\_iterator.
3. Дана строка name и вектор V с четным количеством элементов. Найти все различные числа, которые входят во вторую половину исходного вектора и при этом отсутствуют в первой половине. Записать найденные числа в текстовый файл с именем name в убывающем порядке, выводя каждое число на новой строке. Использовать алгоритм set\_difference для двух вспомогательных множеств и итератора ostream\_iterator. Чтобы обеспечить вывод чисел в нужном порядке, при создании множеств и в алгоритме использовать функциональный объект greater.
4. Даны векторы V1 и V2. Найти все числа (с учетом повторений), которые входят хотя бы в один из исходных векторов, и вывести их в порядке возрастания; при этом если, например, некоторое число входит в один из векторов 3 раза, а в другой 5 раз, то его надо вывести 5 раз. Использовать алгоритм set\_union для двух вспомогательных мультимножеств и итератора ptout\_iterator.
5. Даны векторы V1 и V2 с различным количеством элементов. Найти все числа (с учетом повторений), которые входят в один из исходных векторов и отсутствуют в другом, и вывести их в порядке убывания; при этом если, например, некоторое число входит в один из векторов 3 раза, а в другой 5 раз, то его надо вывести 2 раза. Использовать алгоритм set\_symmetric\_difference для двух вспомогательных мультимножеств и итератора ptout\_iterator. Чтобы обеспечить вывод чисел в нужном порядке, при создании множеств и в алгоритме использовать функциональный объект greater.
6. Дан вектор V, содержащий не менее трех различных чисел. Вывести все его различные элементы, кроме максимального и минимального, в порядке убывания. Использовать вспомогательное множество и алгоритм copy с обратными итераторами, указывающими на предпоследний и первый элементы множества, и с итератором ptout\_iterator.
7. Дан вектор V, содержащий не менее трех различных чисел. Вывести все его элементы (с учетом повторений), кроме минимального и максимального, в порядке возрастания. Использовать вспомогательное мультимножество, его функции-члены lower\_bound и upper\_bound и алгоритм copy с итератором ptout\_iterator.
8. Дан вектор V. Определить количество повторений каждого числа в векторе V и вывести все различные элементы вектора V вместе с количеством их повторений (в порядке возрастания значений элементов); количество повторений выводить сразу после значения соответствующего элемента. Использовать вспомогательное мультимножество и цикл, в котором вызывается функция-член upper\_bound для мультимножества и функция difference для его итераторов.

Отображения. Группировка и объединение данных

(map, multimap)

1. Дан вектор V. Определить количество повторений каждого числа в векторе V и вывести все различные элементы вектора V вместе с количеством их повторений (в порядке возрастания значений элементов); количество повторений выводить сразу после значения соответствующего элемента. Использовать вспомогательное отображение M (класс map), ключами которого являются различные элементы вектора V, а значениями – количество повторений этих элементов. При заполнении отображения M не использовать условные конструкции (достаточно операций индексирования [] и инкремента). Элементы вектора V (при заполнении отображения M) и элементы отображения M (при выводе полученных результатов) перебирать в цикле с параметром-итератором соответствующего контейнера.
2. Дан вектор V. Выполнить группировку элементов вектора V, используя в качестве ключа группировки последнюю (т. е. правую) цифру элемента: в одну группу должны входить все элементы вектора V, оканчивающиеся на одну и ту же цифру (сгруппированные элементы должны располагаться в том же порядке, в котором они располагались в исходном векторе). Представить результат группировки в виде отображения M, ключами которого являются ключи группировки, а значениями – векторы, содержащие сгруппированные элементы (таким образом, отображение M должно иметь тип map>). Вывести полученное отображение (для каждого элемента отображения M вначале выводить ключ, а затем элементы связанного с ним вектора). Для перебора элементов контейнеров использовать циклы с параметрами-итераторами.
3. Дан вектор V, элементами которого являются английские слова, набранные заглавными буквами. Выполнить группировку элементов вектора V, используя в качестве ключа группировки первую букву элемента: в одну группу должны входить все элементы вектора V, начинающиеся с одной и той же буквы (сгруппированные элементы должны располагаться в алфавитном порядке с учетом повторений). Представить результат группировки в виде отображения M, ключами которого являются ключи группировки, а значениями – мультимножества, содержащие сгруппированные элементы (таким образом, отображение M должно иметь тип map>). Вывести полученное отображение (для каждого элемента отображения M вначале выводить ключ, а затем элементы связанного с ним мультимножества). Для перебора элементов контейнеров использовать алгоритм for\_each или, если компилятор поддерживает стандарт C++11, цикл for по элементам контейнера.
4. Дан вектор V. Выполнить группировку элементов вектора V, используя в качестве ключа группировки последнюю (т. е. правую) цифру элемента: в одну группу должны входить все элементы вектора V, оканчивающиеся на одну и ту же цифру (сгруппированные элементы должны располагаться в том же порядке, в котором они располагались в исходном векторе). Представить результат группировки в виде мультиотображения M (класса multimap), ключами которого являются ключи группировки, т. е. последние цифры элементов вектора V, а значениями – элементы вектора, оканчивающиеся на соответствующую цифру (таким образом, отображение M должно иметь тип multimap). Вывести полученное отображение (для каждого элемента отображения M вначале выводить ключ, а затем связанный с ним элемент вектора V; ключи могут повторяться). Для перебора элементов контейнеров использовать циклы с параметрами-итераторами.