

Ассоциативные контейнеры

Во всех заданиях данной группы заготовки решения уже содержат операторы, обеспечивающие заполнение исходных контейнеров. Если тип элементов контейнера не указан, то предполагается, что элементами являются целые числа.

Множества.

Теоретико-множественные алгоритмы

STL5Assoc1. 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 .

STL5Assoc2. Дан вектор V_0 , целое число $N (> 0)$ и набор векторов V_1, \dots, V_N .

Известно, что размер вектора V_0 не превосходит размера любого из векторов V_1, \dots, V_N . Найти количество векторов $V_i, I = 1, \dots, N$, в которых содержатся все элементы вектора V_0 (без учета их повторений). Использовать алгоритм `includes`, применяя его в цикле к двум множествам, одно из которых создано на основе вектора V_0 , а другое на очередной итерации содержит элементы очередного из векторов $V_i, I = 1, \dots, N$.

STL5Assoc3. Дан вектор V_0 , целое число $N (> 0)$ и набор векторов V_1, \dots, V_N .

Известно, что размер вектора V_0 не превосходит размера любого из векторов V_1, \dots, V_N . Найти количество векторов $V_i, I = 1, \dots, N$, в которых содержатся все элементы вектора V_0 (с учетом их повторений). Использовать алгоритм `includes`, применяя его в цикле к двум *мультимножествам* (контейнерам типа `multiset`), одно из которых создано на основе вектора V_0 , а другое на очередной итерации содержит элементы очередного из векторов $V_i, I = 1, \dots, N$.

STL5Assoc4. 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`.

STL5Assoc5. 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`.

STL5Assoc6. Даны векторы V_1 и V_2 .

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`.

STL5Assoc7. Даны векторы V_1 и V_2 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 вы-

вод чисел в нужном порядке, при создании множеств и в алгоритме использовать функциональный объект `greater`.

STL5Assoc8. 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 сорту с обратными итераторами, указывающими на предпоследний и первый элементы множества, и с итератором `ptout_iterator`.

STL5Assoc9. Решить задачу STL5Assoc8, не используя вспомогательное множество. Вместо него последовательно использовать для исходного вектора алгоритмы `sort`, `unique` и `sort`.

STL5Assoc10. 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 сорту с итератором `ptout_iterator`.

STL5Assoc11. Решить задачу STL5Assoc10, не используя вспомогательное мультимножество. Вместо него последовательно использовать для исходного вектора алгоритмы `sort`, `lower_bound`, `upper_bound` и `sort`.

STL5Assoc12. 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.

STL5Assoc13. Решить задачу STL5Assoc12, не используя вспомогательное мультимножество и функцию `difference`. Вместо это использовать для исходного вектора алгоритм `sort` и в цикле алгоритм `upper_bound` и операцию разности для итераторов вектора.

STL5Assoc14. Решить задачу STL5Assoc12, не используя функцию `upper_bound` и цикл. Вместо этого использовать вспомогательное мультимножество M и вспомогательное множество S (создав их на основе исходного вектора) и алгоритм `for_each` для множества S с параметром — функциональным объектом, в котором использовать функцию-член `count` мультимножества M .

Отображения. Группировка и объединение данных

STL5Assoc15. 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.

STL5Assoc16. Решить задачу STL5Assoc15, используя для перебора элементов вектора V и отображения M вызовы алгоритма `for_each` или, если компилятор поддерживает стандарт C++11, циклы `for` по элементам контейнера.

STL5Assoc17. Дан вектор V , элементами которого являются английские слова, набранные заглавными буквами. Определить суммарную длину слов, начинающихся с одной и той же буквы, и вывести все различные буквы, с которых начи-

наются элементы вектора V , вместе с суммарной длиной этих элементов (в алфавитном порядке букв); длину выводить сразу после соответствующей буквы. Использовать вспомогательное отображение M , ключами которого являются начальные буквы элементов вектора V , а значениями — суммарная длина этих элементов. При заполнении отображения M не использовать условные конструкции (достаточно операций индексирования [], инкремента и функции-члена `size` для строк). Элементы вектора V (при заполнении отображения M) и элементы отображения M (при выводе полученных результатов) перебирать в цикле с параметром-итератором соответствующего контейнера.

STL5Assoc18. Решить задачу STL5Assoc17, используя для перебора элементов вектора V и отображения M вызовы алгоритма `for_each` или, если компилятор поддерживает стандарт C++11, циклы `for` по элементам контейнера.

STL5Assoc19. 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<int, vector<int>>`). 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.

STL5Assoc20. 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 повторов). 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<char, multiset<string>>`). 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.

STL5Assoc21. 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<int, int>`). 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.

STL5Assoc22. Дан вектор V , элементами которого являются английские слова, набранные заглавными буквами. Выполнить группировку элементов вектора V , используя в качестве ключа группировки последнюю букву элемента: в одну группу должны входить все элементы вектора V , оканчивающиеся одной и той же буквой (сгруппированные элементы должны располагаться в порядке, обратном порядку их расположения в исходном векторе). Представить результат группировки в виде мультиотображения M , ключами которого являются ключи группировки, т. е. последние буквы элементов вектора V , а значениями — элементы вектора, оканчивающиеся на соответствующую букву (таким образом, отображение M должно иметь тип `multimap<char, string>`). 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 алгоритм `for_each` или, если компилятор поддерживает стандарт C++11, цикл `for` по элементам контейнера.

Указание. Для расположения элементов в группе в порядке, обратном исходному порядку их расположения в векторе, достаточно при формировании мультиотображения M перебирать элементы вектора V в обратном порядке (используя обратные итераторы). Требуемый результат можно получить и при прямом переборе элементов вектора, если использовать вариант функции-члена `M.insert` с первым параметром-итератором («подсказкой»), в качестве которого указывать возвращаемое значение функции-члена `M.lower_bound` для соответствующего ключа.

STL5Assoc23. Дан вектор V . В каждой группе его элементов, оканчивающихся на одну и ту же цифру, найти сумму значений этих элементов, за исключением начального элемента группы (предполагается, что элементы группы располагаются в том же порядке, что и в исходном векторе). Если группа состоит из единственного элемента, то сумма должна равняться 0. Для каждой группы вывести соответствующую ей цифру и найденную сумму, упорядочивая выводимые пары по возрастанию цифр. Использовать вспомогательное отображение M и вариант группировки, описанный в задаче STL5Assoc19. Для нахождения сумм использовать алгоритм `accumulate`.

STL5Assoc24. Решить задачу STL5Assoc23, используя вспомогательное мультиотображение M и вариант группировки, описанный в задаче STL5Assoc21. Для нахождения сумм использовать алгоритм `accumulate`.

STL5Assoc25. Решить задачу STL5Assoc23, не прибегая к этапу предварительной группировки. Использовать отображение M с ключом — последней цифрой числа и значением — суммой требуемых чисел (ср. с задачей STL5Assoc15). При формировании отображения M использовать, наряду с операциями индексирования и инкремента, функцию-член `find`.

STL5Assoc26. Дан вектор V , элементами которого являются английские слова, набранные заглавными буквами. В каждой группе его элементов, оканчивающихся на одну и ту же букву, получить строку, являющуюся суммой всех слов из этой группы, кроме последнего слова (предполагается, что элементы группы располагаются в том же порядке, что и в исходном векторе). При построении строки добавлять после каждого слова пробел. Если группа состоит из единственного элемента, то строка должна остаться пустой. Для каждой группы вывести соответствующую ей букву и найденную строку, упорядочивая выводимые пары в алфавитном порядке букв. Использовать вспомогательное отображение M и вариант группировки, описанный в задаче STL5Assoc20. Для нахождения сумм использовать алгоритм `accumulate`.

STL5Assoc27. Решить задачу STL5Assoc26, используя вспомогательное мультиотображение M и вариант группировки, описанный в задаче STL5Assoc22. Для нахождения сумм использовать алгоритм `accumulate`.

STL5Assoc28. Решить задачу STL5Assoc26, не прибегая к этапу предварительной группировки. Использовать отображение M с ключом — последней буквой слова и значением — суммой требуемых слов (ср. с задачей STL5Assoc17). При формировании отображения M организовать перебор исходного вектора V с конца и использовать, наряду с операциями индексирования и инкремента, функцию-член `find`.

STL5Assoc29. Даны векторы V_1 и V_2 ; все элементы в каждом векторе различны. Получить вектор V , содержащий пары чисел (типа `pair<int, int>`), удовлетворяющие следующим условиям: первый элемент пары принадлежит вектору V_1 , второй принадлежит вектору V_2 , и оба элемента оканчиваются одной и той же цифрой. Полученный набор пар называется *внутренним объединением* векторов V_1 и V_2 по ключу, определяемому последними цифрами исходных чисел. Порядок следования пар определяется исходным порядком элементов вектора V_1 , а для равных первых элементов — исходным порядком элементов вектора V_2 . Для построения вектора V выполнить группировку элементов вектора V_2 по ключу — последней цифре, используя вариант группировки со вспомогательным отображением M , описанный в задаче STL5Assoc19, после чего в цикле по элементам вектора V_1 сформировать требуемое внутреннее объединение, перебирая для каждого элемента вектора V_1 соответствующие ему элементы отображения M . Вывести размер полученного вектора V и все его элементы.

STL5Assoc30. Решить задачу STL5Assoc29, выполняя группировку элементов вектора V_2 способом, описанным в задаче STL5Assoc21 (с использованием вспомогательного мультиотображения M). Для поиска в мультиотображении M элементов с требуемым ключом использовать функцию-член `equal_range`.

STL5Assoc31. Даны векторы V_1 и V_2 , элементами которых являются английские слова, набранные заглавными буквами, причем все слова в каждом векторе различны. Получить вектор V , являющийся внутренним объединением векторов V_1 и V_2 (см. STL5Assoc29), каждая пара которого содержит слова одинаковой длины. Порядок следования пар определяется алфавитным порядком первых элементов пар, а для равных первых элементов — порядком вторых элементов, обратным алфавитному. Для построения вектора V выполнить группировку элементов вектора V_2 по ключу — длине слова, используя вариант группировки со вспомогательным отображением M типа `map<int, set<string>>` (ср. с задачей STL5Assoc20), после чего в цикле по отсортированным элементам вектора V_1 сформировать требуемое внутреннее объединение, перебирая для каждого элемента вектора V_1 соответствующие ему элементы отображения M . Вывести размер полученного вектора V и все его элементы.

STL5Assoc32. Даны векторы V_1 и V_2 , элементами которых являются английские слова, набранные заглавными буквами, причем все слова в каждом векторе различны. Получить вектор V , являющийся внутренним объединением векторов V_1 и V_2 (см. STL5Assoc29), в каждой паре которого первое слово начинается с буквы, которой оканчивается второе слово. Порядок следования пар определяется алфавитным порядком первых элементов пар, а для равных первых элементов — порядком вторых элементов, обратным порядку их следования в векторе V_2 . Для построения вектора V выполнить группировку элементов вектора V_2 по ключу — последней букве слова, используя вариант группировки со вспомогательным мультиотображением M , описанный в задаче STL5Assoc22,

после чего в цикле по отсортированным элементам вектора V_1 сформировать требуемое внутреннее объединение, перебирая для каждого элемента вектора V_1 соответствующие ему элементы отображения M . Вывести размер полученного вектора V и все его элементы.

STL5Assoc33. Даны векторы V_1 и V_2 ; все элементы в каждом векторе различны и являются положительными числами. Получить вектор V пар типа `pair<int, vector<int>>`, в которых первый элемент пары принадлежит вектору V_1 , а второй представляет собой набор тех элементов вектора V_2 , которые оканчиваются той же цифрой, что и первый элемент пары (если подходящие элементы в векторе V_2 отсутствуют, то второй элемент пары должен содержать единственный элемент, равный 0). Полученный набор пар называется *левым внешним объединением* векторов V_1 и V_2 по ключу, определяемому последними цифрами исходных чисел. Порядок следования пар определяется исходным порядком элементов вектора V_1 ; порядок чисел, входящих во вторые элементы пар, определяется исходным порядком элементов вектора V_2 . Для построения вектора V выполнить группировку элементов вектора V_2 по ключу — последней цифре, используя вариант группировки со вспомогательным отображением M , описанный в задаче STL5Assoc19, после чего в цикле по элементам вектора V_1 сформировать требуемое внешнее объединение, перебирая для каждого элемента вектора V_1 соответствующие ему элементы отображения M . Вывести полученный вектор V , указывая после первого элемента пары все числа, входящие во второй элемент данной пары.

STL5Assoc34. Даны векторы V_1 и V_2 , элементами которых являются английские слова, набранные заглавными буквами, причем все слова в каждом векторе различны. Получить левое внешнее объединение векторов V_1 и V_2 (см. STL5Assoc33) по ключу — длине слова: каждому элементу вектора V_1 должен соответствовать набор всех элементов вектора V_2 , имеющих ту же длину, или набор, содержащий только пустую строку, если требуемые элементы в векторе V_2 отсутствуют. Левое внешнее объединение должно быть упорядочено в алфавитном порядке элементов из вектора V_1 ; в каждом наборе элементов, соответствующих элементу из V_1 , порядок должен быть обратным алфавитному порядку. Реализовать левое внешнее объединение в виде отображения M с ключом — строкой и значением — строковым множеством. Для построения множества M выполнить группировку элементов вектора V_2 по ключу — длине слова, используя вариант группировки со вспомогательным отображением M_0 типа `map<int, set<string, greater<string>>>` (ср. с задачей STL5Assoc20), после чего в цикле по элементам вектора V_1 сформировать требуемое внутреннее объединение, перебирая для каждого элемента вектора V_1 соответствующие ему элементы отображения M_0 . Вывести полученное отображение M , указывая после каждого его ключа (т. е. элемента вектора V_1) все слова, входящие в связанное с этим ключом множество значений (т. е. все соответствующие ему элементы вектора V_2 или, при их отсутствии, пустую строку).

STL5Assoc35. Даны векторы V_1 и V_2 ; все элементы в каждом векторе различны и являются положительными числами. Получить вектор V пар типа `pair<int, int>`, в которых первый элемент пары принадлежит вектору V_1 , а второй представляет собой сумму значений тех элементов вектора V_2 , которые оканчиваются той же цифрой, что и первый элемент пары (если подходящие элементы в векторе V_2 отсутствуют, то второй элемент пары должен быть равен 0). Порядок следования пар должен быть обратным порядку следования элементов вектора V_1 . При построении вектора V использовать вспомогательное отображение M , построенное на основе вектора V_2 и сопоставляющее каждой цифре сумму элементов вектора V_2 , оканчивающихся этой цифрой (по поводу ва-

риантов построения отображения M см. задачи STL5Assoc23–STL5Assoc25; допускается использовать любой вариант).

STL5Assoc36. Даны векторы V_1 и V_2 , элементами которых являются английские слова, набранные заглавными буквами, причем все слова в каждом векторе различны. Получить отображение M , в котором ключом является элемент вектора V_1 , а значением — строка, полученная суммированием следующего набора элементов вектора V_2 : в набор включаются все слова, оканчивающиеся на ту же букву, что и ключ, за исключением последнего подходящего слова. Порядок слов в наборе должен соответствовать их порядку в векторе V_2 . При построении строки добавлять после каждого слова пробел. Если требуемый набор является пустым, то строка также должна остаться пустой. При построении отображения M использовать вспомогательное отображение M_0 , построенное на основе вектора V_2 и сопоставляющее каждой букве описанную выше сумму элементов вектора V_2 , оканчивающихся этой буквой (по поводу вариантов построения отображения M_0 см. задачи STL5Assoc26–STL5Assoc28; допускается использовать любой вариант).