

Programming Assignment: Контейнер с приоритетами

You have not submitted. You must earn 1/1 points to pass.

**Deadline**

The assignment was due on May 19, 11:59 PM PDT  
You can still pass this assignment before the course ends.

**Instructions**

My submissionDiscussions

Условие задачи

Реализуйте шаблонный контейнер `PriorityCollection`, позволяющий хранить объекты не копируемого типа, а также совершать с ними следующие операции:

- добавить элемент с нулевым приоритетом;
- добавить набор элементов с нулевыми приоритетами;
- увеличить на 1 приоритет заданного элемента;
- извлечь из контейнера элемент с максимальным приоритетом.

Поскольку добавление элемента заключается в его перемещении в контейнер, метод увеличения приоритета не может принимать сам элемент. Поэтому метод, добавляющий элементы, для каждого из них должен вернуть некий идентификатор, который затем можно использовать для вызова других методов.

Тип идентификатора — `PriorityCollection::Id` — вы должны выбрать самостоятельно. Он должен быть легко копируемым: например, числом, указателем или итератором. Примеры определения и использования типа `Id` вы можете найти в шаблоне решения.

Структура класса

```
1 template <typename T>
2 class PriorityCollection {
3 public:
4     using Id = /* тип, используемый для идентификаторов */;
5
6     // Добавить объект с нулевым приоритетом
7     // с помощью перемещения и вернуть его идентификатор
8     Id Add(T object);
9
10    // Добавить все элементы диапазона [range_begin, range_end)
11    // с помощью перемещения, записав выданные им идентификаторы
12    // в диапазон [ids_begin, ...).
13    template <typename ObjInputIt, typename IdOutputIt>
14    void Add(ObjInputIt range_begin, ObjInputIt range_end,
15            IdOutputIt ids_begin);
16
17    // Определить, принадлежит ли идентификатор какому-либо
18    // хранящемуся в контейнере объекту
19    bool IsValid(Id id) const;
20
21    // Получить объект по идентификатору
22    const T& Get(Id id) const;
23
24    // Увеличить приоритет объекта на 1
25    void Promote(Id id);
26
27    // Получить объект с максимальным приоритетом и его приоритет
28    pair<const T&, int> GetMax() const;
29
30    // Аналогично GetMax, но удаляет элемент из контейнера
31    pair<T, int> PopMax();
32
33 private:
34    // Приватные поля и методы
35 };
```

Гарантии

1. Вы можете полагаться на то, что для элементов типа `T` определены перемещение и сравнение на `==`, `!=` и `<`. Все операции над элементами имеют константную сложность.
2. Гарантируется, что все добавляемые элементы различны, однако добавленный и затем извлечённый элемент может быть добавлен вновь.
3. Методы `Get` и `Promote` всегда вызываются от валидных с точки зрения метода `IsValid` идентификаторов.
4. Методы `GetMax` и `PopMax` вызываются только при наличии элементов в контейнере.
5. При наличии нескольких элементов с одинаковым максимальным приоритетом необходимо выбирать и извлекать тот, который был добавлен позже остальных.

Файл с заготовкой решения

```
priority_collection_2.cpp
```

Ограничения

Контейнер должен обрабатывать  $10^6$  запросов не более чем за 1 секунду. Гарантируется, что суммарная длина диапазонов, с которыми вызывается вторая версия метода `Add`, также не превосходит  $10^6$ .

Подсказки

Если задача вызывает непреодолимые трудности, вы можете воспользоваться подсказками. Они помогут быстрее сдать задачу, но в итоге вы получите меньше пользы от процесса поиска решения.

Не нужно открывать все подсказки сразу. Используйте их последовательно.

- priority-collection-1.pdf
- priority-collection-2.pdf
- priority-collection-3.pdf
- priority-collection-4.pdf

How to submit

When you're ready to submit, you can upload files for each part of the assignment on the "My submission" tab.