

## Programming Assignment: Односвязный список

✓ Passed · 1/1 points

**Deadline** The assignment was due on May 5, 11:59 PM PDT  
You can still pass this assignment before the course ends.

**Instructions** My submission

Discussions

Односвязный список — классический пример структуры данных, для реализации которой нужно пользоваться памятью в куче. В этой задаче вам нужно реализовать шаблон класса `LinkedList`, который представляет собой односвязный список и обладает следующим интерфейсом:

```
1  template <typename T>
2  class LinkedList {
3  public:
4      struct Node {
5          T value;
6          Node* next = nullptr;
7      };
8
9      ~LinkedList();
10
11     void PushFront(const T& value);
12     void InsertAfter(Node* node, const T& value);
13     void RemoveAfter(Node* node);
14     void PopFront();
15
16     Node* GetHead() { return head; }
17     const Node* GetHead() const { return head; }
18
19 private:
20     Node* head = nullptr;
21 };
```

- Метод `GetHead` возвращает указатель на голову списка, он используется для перебора элементов списка (см. шаблон `ToVector` в заготовке решения)
- Метод `PushFront` добавляет новый элемент в голову списка.
- Метод `InsertAfter` вставляет новый элемент в список так, чтобы он шёл после узла `node`. Если `node == nullptr`, метод эквивалентен `PushFront`
- Метод `PopFront` удаляет элемент из головы списка и освобождает выделенную под него память. Если список пуст, метод завершается корректно. Если после выполнения метода список стал пустым, метод `GetHead` должен возвращать `nullptr`
- Метод `RemoveAfter` должен удалять из списка элемент, который следует за узлом `node`, и освобождать выделенную под него память. Если `node == nullptr`, метод эквивалентен `PopFront`. Если `node->next == nullptr`, то метод должен корректно завершаться.
- Все методы, перечисленные выше, должны работать за  $O(1)$
- Деструктор класса `LinkedList` освобождает всю память, выделенную для хранения элементов списка.

Файл с заготовкой решения

linked\_list.cpp

### How to submit

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