**Костин Андрей Константинович, 3530904/20001, Лабораторная работа №1 «Однонаправленный упорядоченный список. Словари»**

**Общая постановка задачи:**

**Словарь** – динамическое множество, поддерживающее операции:  
• добавить элемент (insert);  
• удалить элемент (delete);  
• проверить, есть ли элемент в словаре (search);  
Словарь не может иметь записи с одинаковыми ключами (словами).  
АТД словарь (DictionaryList) необходимо реализовать на основе однонаправленного  
упорядоченного списка.  
  
**Словарные операции:**  
1) *Создать «пустой» словарь*2) *Добавить слово в словарь*, обеспечивая лексикографическую упорядоченность и  
уникальность ключей (слов).  
3) *Найти слов*о в словаре.  
4) *Удалить слово* из словаря.

**Действия со словарями:**   
При выполнении действий необходимо учитывать лексикографическую упорядоченность и уникальность ключей (слов).  
Первый словарь – объект, к которому применяется метод, второй – параметр метода.  
***1) Объединить словари (merge).*** Метод класса для добавления в первый словарь слов,  
содержащихся во втором. Элементы второго словаря, уже присутствующие в первом не  
добавлять; после выполнения операции второй словарь должен быть пустым. При  
выполнении задания нельзя копировать списки, копировать узлы списков. Для вставки  
узлов нужно корректировать ссылки.  
***2) Удалить словарь (delete).*** Метод класса для удаления из первого словаря слов,  
встречающихся во втором. В процессе выполнения метода второй словарь не меняется.  
***3) Получить пересечение словарей (getIntersection*).** *Дружественная функция*,  
формирующая новый словарь, содержащий слова, присутствующие одновременно в двух  
словарях. Исходные словари остаются без изменения.

*Для эффективной работы со списком реализуйте конструктор перемещения и оператор  
перемещающего присваивания.*

**Таблица с детальными требованиями и тест планом:**

|  |  |  |  |
| --- | --- | --- | --- |
| Требования | Детальные требования | Данные | Ожидаемый результат |
| 1. Тест метода insert() | | | |
| Если элемент вставлен успешно, выводится сообщение «Вставлен элемент (элемент)». Если данный элемент уже есть в списке, выводится сообщение «Элемент (…) уже существует» |  | 9 2 5 1 7 5 | Рис.1 |
| 1. Тест метода search() | | | |
| Если элемент уже есть в списке, выводится сообщение «Элемент (…) - найдено». Если элемента нет в словаре, выводится сообщение «Элемент (…) - не найдено» |  | 1 5 9  -8 4 15 | Рис. 2 |
| 1. Тест метода deleteElement() | | | |
| Если элемент существует в списке и был успешно удален, выводится сообщение «Элемент (…) - удалено». Если нет такого элемента, выводится сообщение «Элемент (…) - не найдено» |  | 5 1 9  **15**  2 7  **-** | Рис. 3 |
| 1. Тест метода merge() | | | |
| *Метод класса для добавления в первый словарь слов, содержащихся во втором.* Элементы второго словаря, уже присутствующие в первом не добавлять; после выполнения операции второй словарь должен быть пустым. Если словарь пустой, выводится сообщение «Список пустой.» |  | 789 345 567 321 82  81 123 456 9 987 **567**  - | Рис. 4 |
| 1. Тест метода deleteDictionary() | | | |
| *Метод класса для удаления из первого словаря слов, встречающихся во втором.* В процессе выполнения метода второй словарь не меняется. |  | 81 123 456 9 987 888 777 444 | Рис. 5 |
| 1. Тест метода getIntersection() | | | |
| *Дружественная функция, формирующая новый словарь, содержащий слова, присутствующие одновременно в двух словарях.* Исходные словари остаются без изменения. |  | 789 345 567  589 467 | Рис. 6 |

**Список иллюстраций**

[Рисунок 1 5](#_Toc127874899)

[Рисунок 2 5](#_Toc127874900)

[Рисунок 3 5](#_Toc127874901)

[Рисунок 4 6](#_Toc127874902)

[Рисунок 5 7](#_Toc127874903)

[Рисунок 6 7](#_Toc127874904)

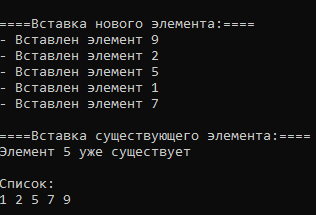


Рисунок 1

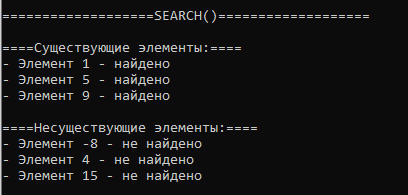


Рисунок 2

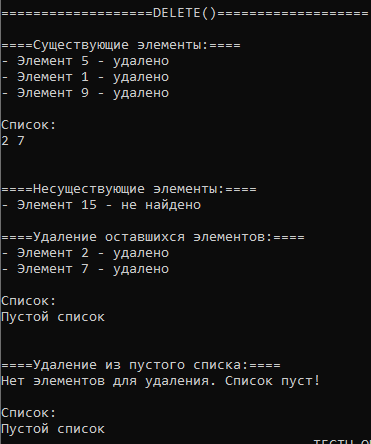


Рисунок 3

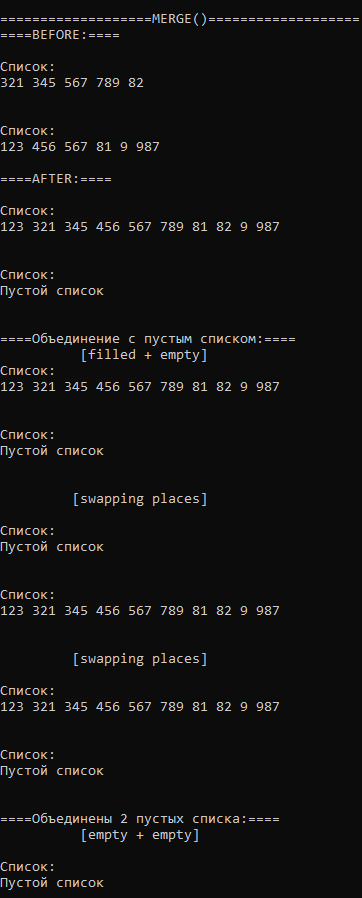


Рисунок 4

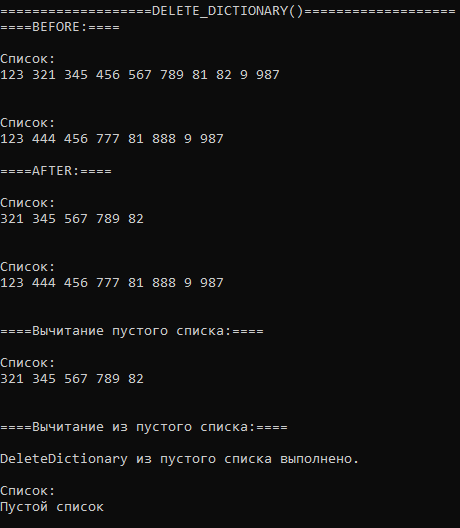


Рисунок 5

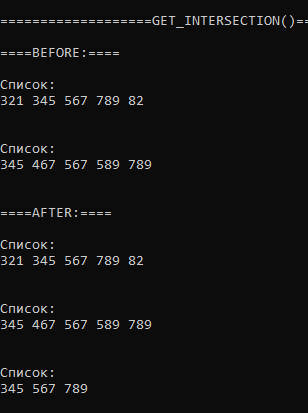


Рисунок 6