

Постановка задачі

Розробити 32-розрядну програму, яка демонструє можливості мови C++, а також його стандартної бібліотеки по роботі з потоками, контейнерами та ітераторами.

Використання усіх вищезгаданих можливостей є обов'язковим. Повинні бути реалізовані основні можливості стандартних шаблонів: ітератори, пошук, впорядкування або інше.

Напишіть програму, яка заносить різні слова в файл в алфавітному порядку.

Опис розв'язку

Для розв'язку задачі був створений шаблонний клас `MyVector<class T>` - аналог `std::vector`:

- `MyVector()` - конструктор за замовчуванням
- `MyVector(int initial_capacity)` - конструктор, в якому параметр визначає кількість об'єктів під які необхідно виділити пам'ять
- `MyVector(MyVector<T>& vec)` - конструктор копіювання
- `int getCount()` - повертає кількість об'єктів
- `void add(const T& elem)` - додає новий об'єкт до контейнеру
- `void insert(int i, const T& elem)` - додає новий об'єкт до контейнеру за вказаним індексом
- `void insert(const T& t, const T& elem)` - додає новий об'єкт `elem` до контейнеру і розміщує його після об'єкту `t` (він має бути в контейнері)
- `void removeAt(int i)` - видаляє об'єкт з контейнеру за вказаним індексом
- `void remove(const T& elem)` - видаляє вказаний об'єкт з контейнеру
- `void removeAll()` - видаляє всі об'єкти з контейнеру

Для використання ітераторів в класі були перевизначені класи наступним чином:

```
typedef T value_type;  
typedef value_type* iterator;  
typedef const value_type* const_iterator;  
typedef std::size_t size_type;  
typedef value_type& reference;  
typedef const value_type& const_reference;
```

Для роботи з ітераторами визначені наступні функції:

- `iterator begin()` - повертає ітератор на початковий елемент контейнеру
- `iterator end()` - повертає ітератор на останній елемент контейнеру

Для розв'язку задачі занесення слів в файл в алфавітному порядку всі слова з файлу зчитувались у строку та додавались у контейнер типу `MyVector<string>`. Далі користувачу пропонується вводити слова (для виходу потрібно ввести "exit"). Після введення кожного слова здійснюється ітерація по всім словам, які вже знаходяться в контейнері, знаходить місце, де це слово повинне знаходитись і додає його у потрібне місце в контейнері. Після завершення вводу слів всі слова з контейнеру виводяться у файл.

Вихідний текст програми розв'язку задачі

див. Додаток 1

Опис інтерфейсу (керівництво користувача)

Виідний файл зі словами повинен знаходитись в одній папці із програмою і називатись "words.txt". Після запуску програми всі слова, що були в файлі виводяться на екран. Далі користувачу пропонується вводити слова через Enter. Для виходу потрібно ввести "exit". Після закінчення введення слів виводяться всі слова в упорядкованому вигляді.

Опис тестових прикладів

reading words from words.txt

readed words:

a
aa
aab
ab
aba
ba
baa
bab
bac
bba

input words(to exit type enter):

abb
c
exit
words input ended
result list:
a
aa
aab
ab
aba
abb
ba
baa
bab
bac
bba
c