## 5. Розробка власних контейнерів. Ітератори

**Мета:** Набуття навичок розробки власних контейнерів.

Використання ітераторів.

**1 ВИМОГИ**

**1.1 Розробник**

Інформація про розробника:

* Шарма Олександр Раджнішович
* НТУ “ХПІ” КІТ-119б
* Варіант 25

**1.2 Загальне завдання**

1. Розробити клас-контейнер, що [ітерується](http://docs.oracle.com/javase/8/docs/api/java/lang/Iterable.html) для збереження початкових даних завдання [л.р. №3](https://oop-khpi.github.io/" \l "task_03_app) у вигляді **масиву рядків** з можливістю додавання, видалення і зміни елементів.
2. В контейнері реалізувати та продемонструвати наступні методи:
   * String toString() повертає вміст контейнера у вигляді рядка;
   * void add(String string) додає вказаний елемент до кінця контейнеру;
   * void clear() видаляє всі елементи з контейнеру;
   * boolean remove(String string) видаляє перший випадок вказаного елемента з контейнера;
   * Object[] toArray() повертає масив, що містить всі елементи у контейнері;
   * int size() повертає кількість елементів у контейнері;
   * boolean contains(String string) повертає true, якщо контейнер містить вказаний елемент;
   * boolean containsAll(Container container) повертає true, якщо контейнер містить всі елементи з зазначеного у параметрах;
   * public Iterator<String> iterator() повертає ітератор відповідно до [Interface Iterable](http://docs.oracle.com/javase/8/docs/api/java/lang/Iterable.html).
3. В класі ітератора відповідно до [Interface Iterator](http://docs.oracle.com/javase/8/docs/api/java/util/Iterator.html) реалізувати методи:
   * public boolean hasNext();
   * public String next();
   * public void remove().
4. Продемонструвати роботу ітератора за допомогою циклів *while* и *for each*.
5. Забороняється використання контейнерів (колекцій) і алгоритмів з [Java Collections Framework](https://docs.oracle.com/javase/8/docs/technotes/guides/collections/).

**1.3 Задача**

Ввести декілька рядків. Упорядкувати, а потім вивести рядки за алфавітом (перший пріоритет) та в порядку зростання їх довжини (другий пріоритет).

**2 ОПИС ПРОГРАМИ**

**2.1 Засоби ООП**

У даній програмі відсутні об’єктно-орієнтовані методи.

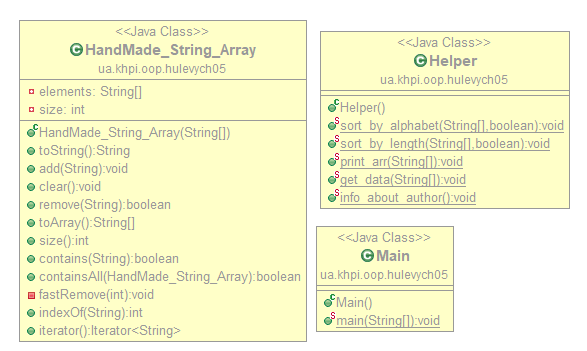
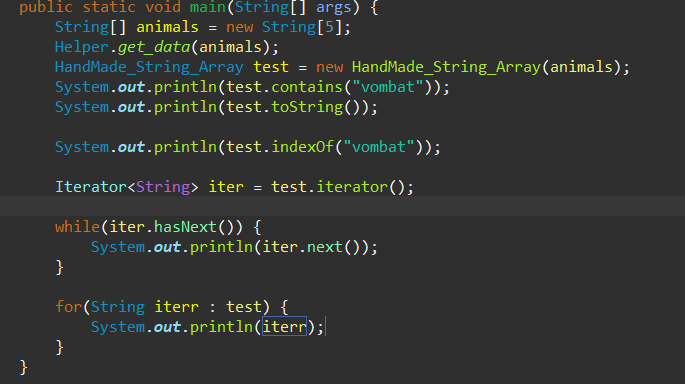
**2.2 Ієрархія та структура даних** Рисунок 1 – діаграма класів **2.3 Важливі фрагменти програми** 

Рисунок 1 – тестування контейнеру згідно з завданням

**3 ВАРІАНТИ ВИКОРИСТАННЯ** Програма створена для тестування роботи власне розробленого контейнера, перевіряється імплементація інтерфейсу Iterator та коректна робота методів, що відповідають вимогам контейнера.

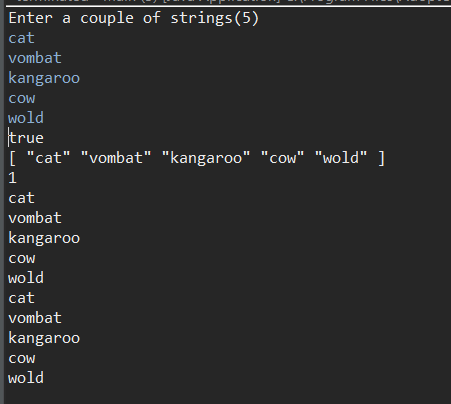


Рисунок 2 – результати роботи програми

**ВИСНОВКИ**

В даній лабораторній роботі було розроблено власний контейнер та набуто навичок у використанні ітераторів.