

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТУ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

**Кафедра систем штучного інтелекту**

**Лабораторна робота № 9**  
з дисципліни  
«Об’єктно-орієнтоване програмування»  
**Варіант - 11**

**Виконав:**  
студент групи КН-108  
Яковлев В.К.  
**Викладач:**  
Гаскьо Р.Т.

Львів – 2018 р.

**Тема:**Розробка власних контейнерів. Ітератори.  
Серіалізація/десеріалізація об'єктів.  
Бібліотека класів користувача

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

- Використання ітераторів.
- Тривале зберігання та відновлення стану об'єктів.
- Ознайомлення з принципами серіалізації/десеріалізації об'єктів.
- Використання бібліотек класів користувача.

Вимоги

1. Розробити клас-контейнер, що ітерується ([docs.oracle.com/javase/8/docs/api/java/lang/Iterable.html](http://docs.oracle.com/javase/8/docs/api/java/lang/Iterable.html)) для збереження початкових даних Вашого варіанту завдання з роботи №8 (**Прикладні задачі. Список з 1-15 варіантів**) 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/>

6. Реалізувати і продемонструвати тривале зберігання/відновлення розробленого контейнера за допомогою серіалізації/десеріалізації . <https://docs.oracle.com/javase/8/docs/technotes/guides/serialization/index.html>

7. Обмінятися відкомпільованим (без початкового коду) службовим класом

(Utility Class) рішення одного варіанту задачі **(Прикладні задачі.**

**Список з 1-15 варіантів)**з сусіднім номером. *1 міняється з 2, 2 з 3, 3 з 4,*

*4 з 5 і т.д. Останній, 15 міняється з 1 варіантом і далі аналогічно.*

8. Продемонструвати послідовну та вибірккову обробку елементів розробленого контейнера за допомогою власного і отриманого за обміном службового класу.

9. Реалізувати та продемонструвати порівняння, сортування та пошук елементів у контейнері.

10. Розробити консольну програму та забезпечити діалоговий режим роботи

з користувачем для демонстрації та тестування рішення.

Код:

```
package com.week;
import java.io.*;
import java.util.*;
import java.util.Scanner;

public class Lab9 {
    public static void main(String[] args) throws Exception
    {

        Set<String> set = new HashSet<String>();

        Scanner in = new Scanner(System.in);

        System.out.println("1)Enter some words: ");
        String s1 = in.nextLine();

        System.out.println("Enter some other words: ");
        String st = in.nextLine();
```

```

        System.out.println(st);

        ITERATOR iterator1 = new ITERATOR();
        iterator1.funcs = "Your serialization is completed";
        FileOutputStream fileOutputStream = new
FileOutputStream("tempfile");
        ObjectOutputStream objectOutputStream = new
ObjectOutputStream(fileOutputStream);
        objectOutputStream.writeObject(iterator1);
        objectOutputStream.close();

        FileInputStream fileInputStream = new
FileInputStream("tempfile");
        ObjectInputStream objectInputStream = new
ObjectInputStream(fileInputStream);

        ITERATOR newFuncs = (ITERATOR)
objectInputStream.readObject();
        objectInputStream.close();
        System.out.println(newFuncs.funcs);

    }

}

package com.week;
import java.io.*;
import java.util.ArrayList;
import java.util.Iterator;
import java.util.List;
import java.util.Scanner;

public class ITERATOR implements Serializable{
    public Object funcs;

    public static void main(String[] args) {

        List<String> list = new ArrayList<String>();
        Scanner in = new Scanner(System.in);
        System.out.println("1)Enter some words: ");
        String s1 = in.nextLine();
        list.add(s1);

        System.out.println("2)Enter some other words: ");

```

```

        String s2 = in.nextLine();
        list.add(s2);

        System.out.println("3)And again, enter some other
words: ");
        String s3 = in.nextLine();
        list.add(s3);

        list.remove(s2);
        System.out.println("Your choicable list: " +
list.get(0) +" "+ list.get(1));
        //System.out.println(list.get(1));

        boolean empty = list.isEmpty();
        System.out.println("IF your collection contains some
elements, it will print: false; " + "Else: true; " + "Yours:
" + empty);

        Iterator<String> iterator = list.iterator();

        while(iterator.hasNext()){
            System.out.println("Now iterator print an element: "
+ iterator.next());
        }
        //list.forEach(System.out::println);

        // System.out.println(list.toString());

        System.out.println("Your list size is: " +
list.size());
        // System.out.println(list.sort());

        list.toArray();
        System.out.println("Now your list is an array: " +
list.get(0) +" "+ list.get(1));
        list.clear();

        System.out.println("List clear");
        System.out.println("IF your collection contains some
elements, it will print: false; " + " Else: true; " + "Yours:
true");
        int funcs;

    }

```

}