

Міністерство освіти і науки України
Національний університет «Львівська політехніка»



Лабораторна робота №9
з курсу:
“ОБ’ЄКТНО ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ”

Виконав:
ст. гр. КН-110
Шаварський
Максим
Прийняв:
Гасько Р.Т.

Львів – 2018 р.

Лабораторна робота №9

Завдання: . Розробити клас-контейнер, що ітерується для збереження початкових даних Вашого варіанту завдання з попередньої роботи (лаб.№8: Прикладні задачі. Список з 1-15 варіантів) у вигляді масиву рядків з можливістю додавання, видалення і зміни елементів.

Виконання роботи

GitHub link - https://github.com/MaxShavarsky/OOP_Labku

Код програми

```
import java.util.Scanner;

class Container
{
    private int size;
    private String[] arr;
    private int currentPos = 0;
    public Container(int size)
    {
        setsize(size);
        arr = new String[size];
    }
    public String toString()
    {
        String temp = "";
        for (int i = 0; i < getsize()-1; i++)
        {
            temp = temp + arr[i];
        }
        return temp;
    }
    public void add(String str)
    {
        if(currentPos < size-1) {
            arr[currentPos] = str;
            currentPos++;
        }
        else {
            arr[currentPos] = str;
            String[] temp = new String[size];
            for(int i=0; i < size;i++)
            {
                temp[i] = arr[i];
            }
            size+=1;
            arr = new String[size];
            for(int i=0; i < size-1;i++)
            {
                arr[i] = temp[i];
            }
            currentPos++;
        }
    }
}
```

```

public void clear()
{
    currentPos = 0;
    arr = new String[2];
    setsize(2);
}
public boolean remove(String str)
{
    for (int i = 0; i < getsize()-1; i++)
    {
        if (str.equals(arr[i]))
        {
            arr[i] = arr[i+1];
            size--;
            currentPos--;
            return true;
        }
    }
    return false;
}
public int getsize()
{
    return size;
}
public void setsize(int Size)
{
    size = Size;
}
public boolean contains(String str)
{
    for (int i = 0; i < getsize()-1; i++)
    {
        if (str.equals(arr[i]))
        {
            return true;
        }
    }
    return false;
}
public boolean containsAll(Container container)
{
    int counter = 0;
    for (int i = 0; i < getsize()-1; i++)
    {
        for (int j = 0; j < container.getsize()-1; j++)
        {
            if (arr[i].equals(container.arr[j]))
            {
                counter++;
                break;
            }
        }
    }
    if (counter == getsize()-1)
    {
        return true;
    }
    else
    {
        return false;
    }
}
public String[] toArray()
{
    String[] temp = new String[getsize()];

```

```

        for (int i = 0; i < getsize()-1; i++)
        {
            temp[i] = arr[i];
        }
        return temp;
    }
    public void show()
    {
        for (int i = 0; i < getsize()-1; i++)
        {
            System.out.println(arr[i]+ ' ');
        }
        System.out.println("\n");
    }
    public int begin()
    {
        return i.begin();
    }
    public int end()
    {
        return i.End(getsize()-1);
    }
    public String next()
    {
        return arr[i.next()];
    }
    public void Delete()
    {
        remove(arr[i.remove()]);
    }
    public boolean hasNext()
    {
        end();
        return i.hasNext(i.next());
    }
    public String getIndex(int index)
    {
        return arr[index];
    }
    public boolean compape(String length1,String length2)
    {
        if(length1.length()==length2.length())
        {
            return true;
        }else {
            return false;
        }
    }
    public String find(String str){
        for(int i=0; i < getsize()-1;i++){
            if(arr[i].equals(str)){
                return arr[i];
            }
        }
        return "Element missing";
    }
    public void sort()
    {
        int count = 0;
        String temp = "";
        while(count != getsize()-1)
        {
            for(int i = 0; i < getsize()-2;i++)
            {
                if(arr[i].charAt(0)>arr[i+1].charAt(0))

```

```

        {
            temp = arr[i];
            arr[i] = arr[i+1];
            arr[i+1] = temp;
            count = 0;
            break;
        }else
        {
            count++;
            if(count == getsize()-1){
                break;
            }
        }
    }
}

class Iterator
{
    private int startposition = 0;
    private int currentposition = 0;
    private int lastindex;
    public int begin()
    {
        return startposition;
    }
    public int End(int size)
    {
        lastindex = size;
        return lastindex;
    }
    public boolean hasNext(int currentposition)
    {
        if (currentposition + 1 != lastindex)
        {
            PlusOne();
            return true;
        }
        else
        {
            return false;
        }
    }
    public int next()
    {
        return currentposition;
    }
    public int remove()
    {
        return currentposition;
    }
    public void PlusOne()
    {
        currentposition++;
    }
    public void SetCurrent (int index)
    {
        currentposition = index;
    }
};

Iterator i = new Iterator();
};

```

```

public class lab {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int anotherSize;
        System.out.print("Enter start size of arrays:");
        anotherSize = sc.nextInt();
        Container c = new Container(anotherSize);
        Container C = new Container(anotherSize);
        String temp1 = "";
        System.out.println("Fill array number 1:");
        for(int i = 0; i < anotherSize+1;i++)
        {
            temp1 = sc.nextLine();
            C.add(temp1);
        }
        String temp2 = "";
        System.out.println("Fill array number 2");
        for(int i = 0; i < anotherSize;i++)
        {
            temp2 = sc.nextLine();
            c.add(temp2);
        }
        String[] str;
        System.out.println("Method toString() " + c.toString());
        System.out.println("Method show() for array number 2: ");
        c.show();
        System.out.print("Method clear() for array number 2: ");
        c.clear();
        c.show();
        System.out.println("Fill array number 2 again");
        for(int i = 0; i < anotherSize;i++)
        {
            temp2 = sc.nextLine();
            c.add(temp2);
        }
        String check;
        System.out.println("Enter string to check if array number 2 contains
it:");
        check = sc.nextLine();
        System.out.println("Method getsize() " + (c.getsize()-1));
        System.out.println("Method contains()");
        if (c.contains(check) == true)
        {
            System.out.println("Container contains string "+ check);
        }
        else
        {
            System.out.println("Container doesnt contains string " + check);
        }
        System.out.println("Method containsAll()");
        if (c.containsAll(C) == true)
        {
            System.out.println("c contains all elem of C ");
        }
        else
        {
            System.out.println("c doesnt contains all elem of C ");
        }
        System.out.println("Method toArray() for array number 2");
        str = c.toArray();
        for (int i = 0; i < c.getsize()-1; i++)
        {
            System.out.print(str[i]+' ');

```

```

    }
    System.out.println("\n");
    System.out.println("Method hasNext() for array number 1");
    while (C.hasNext())
    {
        System.out.print(C.next()+ ' ');
    }System.out.println("\n");
    c.clear();
    System.out.println("Fill array number 2 for method sort()");
    for(int i = 0; i < anotherSize;i++)
    {
        temp2 = sc.nextLine();
        c.add(temp2);
    }
    c.sort();
    c.show();
}
}

```