МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

Кафедра систем штучного інтелекту

Лабораторна робота №12

з дисципліни
«Об’єктно-орієнтоване програмування»

Виконав:
студент групи КН-107
Шиманський П.С.
Прийняв:
Старший викладач
Гасько Р.Т.

Львів – 2018 р.

Код програми:

1)

```
package week12;

public class Info {
	public String passport;
	public String education;
	public String price;
	public String career_date;
	public String career_position;
	public String career_viddil;
	public String[] characteristics = new String[3];

	public void setPassport(String passport) {
		this.passport = passport;
	}

	public String getPassport() {
		return passport;
	}

	public void setEducation(String education) {
		this.education = education;
	}

	public String getEducation() {
		return education;
	}

	public void setPrice(String price) {
		this.price = price;
	}

	public String getPrice() {
		return price;
	}

	public void setDate(String career_date) {
		this.career_date = career_date;
	}

	public String getDate() {
		return career_date;
	}

	public void setPosition(String career_position) {
		this.career_position = career_position;
	}

	public String getPosition() {
		return career_position;
	}

	public void setViddil(String career_viddil) {
		this.career_viddil = career_viddil;
	}

	public String getViddil() {
		return career_viddil;
```

```
	}

	public void setCharacteristics(String[] characteristics) {
		this.characteristics = characteristics;
	}

	public String[] getCharacteristics() {
		return characteristics;
	}
}
```

2)

```
package week12;

import java.util.Iterator;
public class LinkedContainer<E> implements Iterable<E>{
		private Node<E> tail;
	private Node<E> head;
	private int size = 0;

	public LinkedContainer() {

	}

	public void add(String place) {
	    Node<E> newNode = new Node<E>();
	    newNode.setData(place);

	    if (size == 0)
	        head = newNode;
	    else
	        tail.setNext(newNode);

	    tail = newNode;
	    size++;
	}
	public void add(String[] place) {
	    Node<E> newNode = new Node<E>();
	    newNode.setData(place);

	    if (size == 0)
	        head = newNode;
	    else
	        tail.setNext(newNode);

	    tail = newNode;
	    size++;
	}


	public E get(int index) {
	    return findNodeByIndex(index).getData();
	}

	public boolean delete(int index) {
	    if (findNodeByIndex(index) != null ) {

	        if (index != 0)
	            findNodeByIndex(index - 1).setNext(findNodeByIndex(index+1));
	        else
```

```
                head = head.getNext();

            size--;
            return true;
        }
        return false;
    }

    public int size() {
        return size;
    }

    private Node<E> findNodeByIndex(int index) {
        if (index < size && index >= 0) {
            Node<E> curNode = head;
            int curIndex = 0;
            while (curIndex < index){
                if (curNode.getNext() != null) {
                    curNode = curNode.getNext();
                    curIndex++;
                }

            }
            return curNode;
        }
        else
            return null;
    }

    public String toString() {
        StringBuilder resultString = new StringBuilder("[");

        for (int i = 0; i < size; i++) {
            resultString.append(get(i));

            if (i < size-1) resultString.append(", ");
        }
        resultString.append("]");
        return resultString.toString();
    }

    public void clear() {
       size = 0;
       tail = null;
       head = null;
    }


      public Iterator<E> iterator() {
            return new Iterator() {

                    @Override
                    public boolean hasNext() {
                            // TODO Auto-generated method stub
                            return false;
                    }

                    @Override
                    public Object next() {
                            // TODO Auto-generated method stub
                            return null;
                    }
```

```
                        };

                }
}
```

3)

```
package week12;
public class Node<E> {
	private Node<E> next;
	private E data;

	public Node() {
	}

	public Node<E> getNext() {
		return next;
	}
	public void setNext(Node<E> next) {
		this.next = next;
	}
	public E getData() {
		return data;
	}
	public void setData(String place) {
		this.data = (E) place;
	}
	public void setData(String[] place) {
		this.data = (E) place;
	}



}
```

4)

```
package week12;
import java.util.Scanner;
import java.io.*;
public class RegularExpressinos {
	public static void main(String[] args) throws IOException {
		Info info = new Info();
		LinkedContainer<Info> list = new LinkedContainer<Info>();
		Scanner in = new Scanner(System.in);
		System.out.println("a. Зчитати з консолі\nb. Зчитати з файлу");
		String choise = in.nextLine();
		while(true) {
		switch(choise) {
		case "a":
			System.out.print("Ваші паспортні дані: ");
			info.passport = in.nextLine();
```

```
                    list.add(info.passport);
                    System.out.print("Ваша освіта: ");
                    info.education = in.nextLine();
                    list.add(info.education);
                    System.out.print("Ваша зарплата: ");
                    info.price = in.nextLine();
                    list.add(info.price);
                    System.out.print("Ваш відділ: ");
                    info.career_viddil = in.nextLine();
                    list.add(info.career_viddil);
                    System.out.print("Ваша посада: ");
                    info.career_position = in.nextLine();
                    list.add(info.career_position);
                    System.out.print("Дата призначення: ");
                    info.career_date = in.nextLine();
                    list.add(info.career_date);
                    System.out.print("Три ваші основні характеристики: ");
                    for(int i = 0; i < 3; i++) {
                        info.characteristics[i] = in.nextLine();
                        list.add(info.characteristics[i]);
                    }
                    System.out.println(list.toString());
                    break;
                case "b":
                    FileReader fr = new FileReader("info.txt");
                    BufferedReader into = new BufferedReader(fr);
                    while(into.ready()) {
                        System.out.println(into.readLine());
                    }
                    break;
                default :
                    System.out.println("Wrong char");

                }
                break;
        }

        }
}
```

Результат роботи програми:

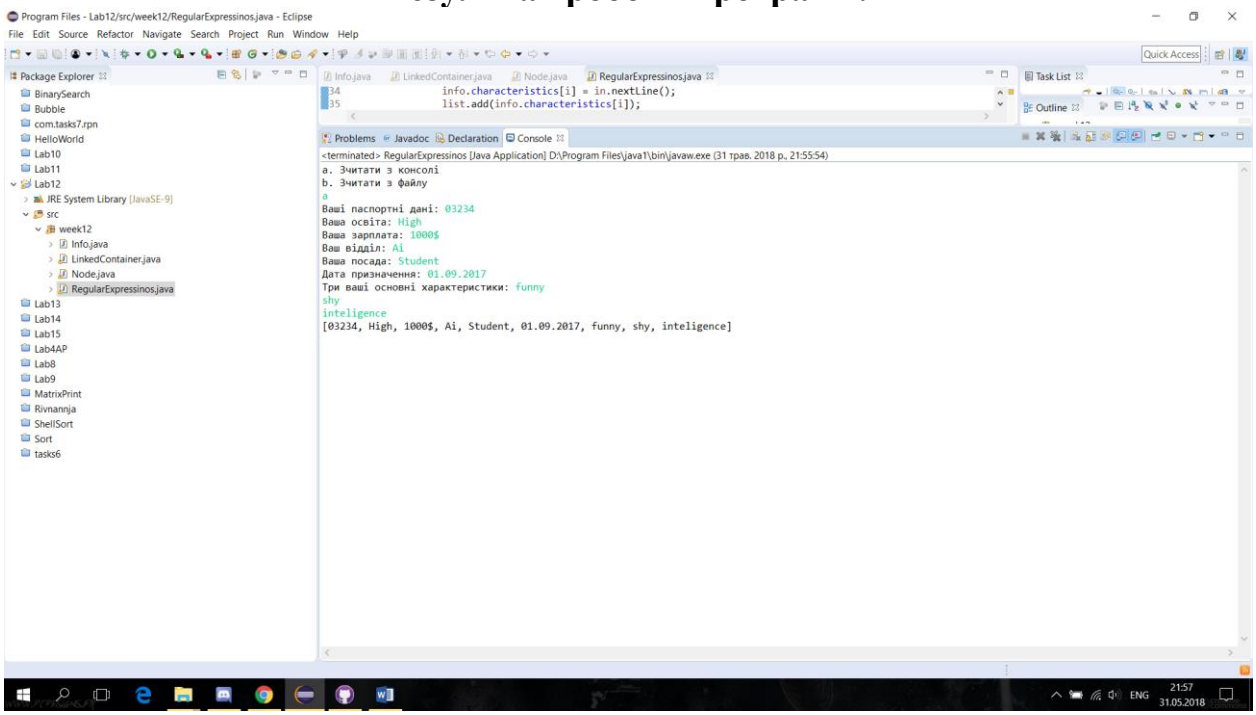

Рис.1 Результат роботи програми