

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТУ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

Кафедра систем штучного інтелекту

Лабораторна робота №3
з дисципліни
«Об'єктно-орієнтоване програмування»

Виконав:

студент групи КН-108

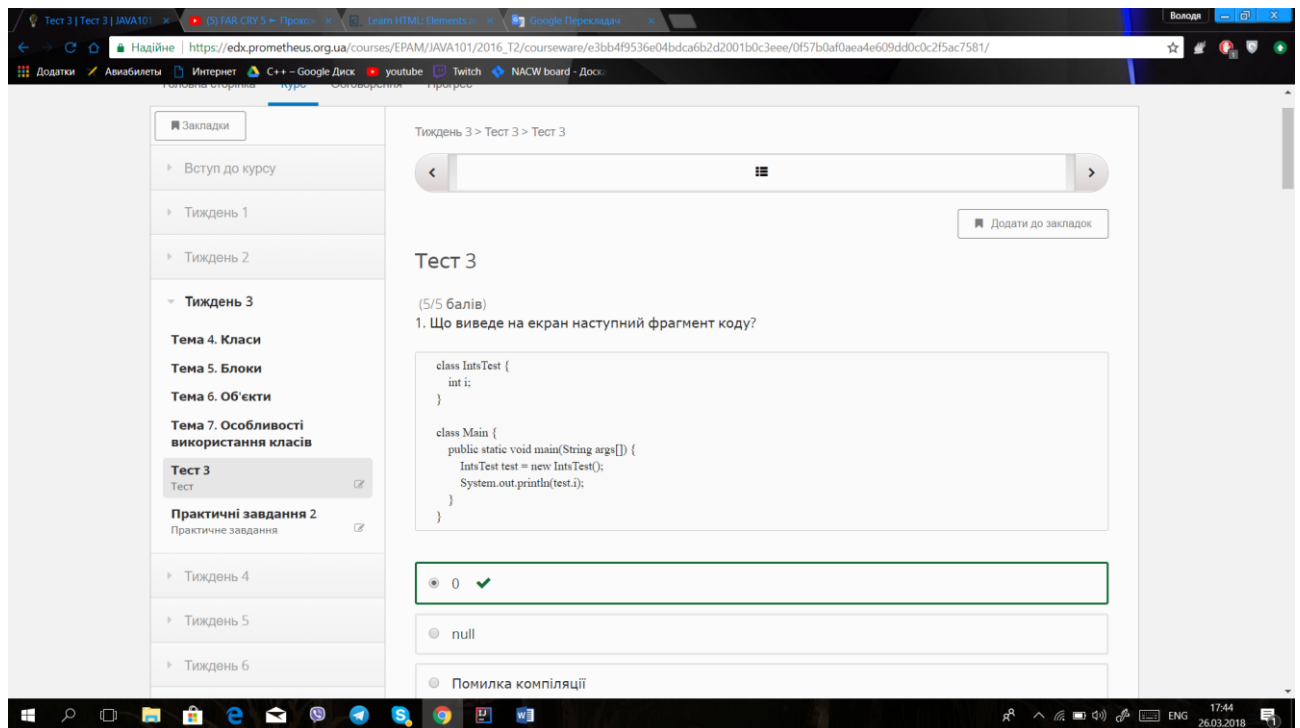
Шалавило Володимир

Викладач:

Гасько Р.Т.

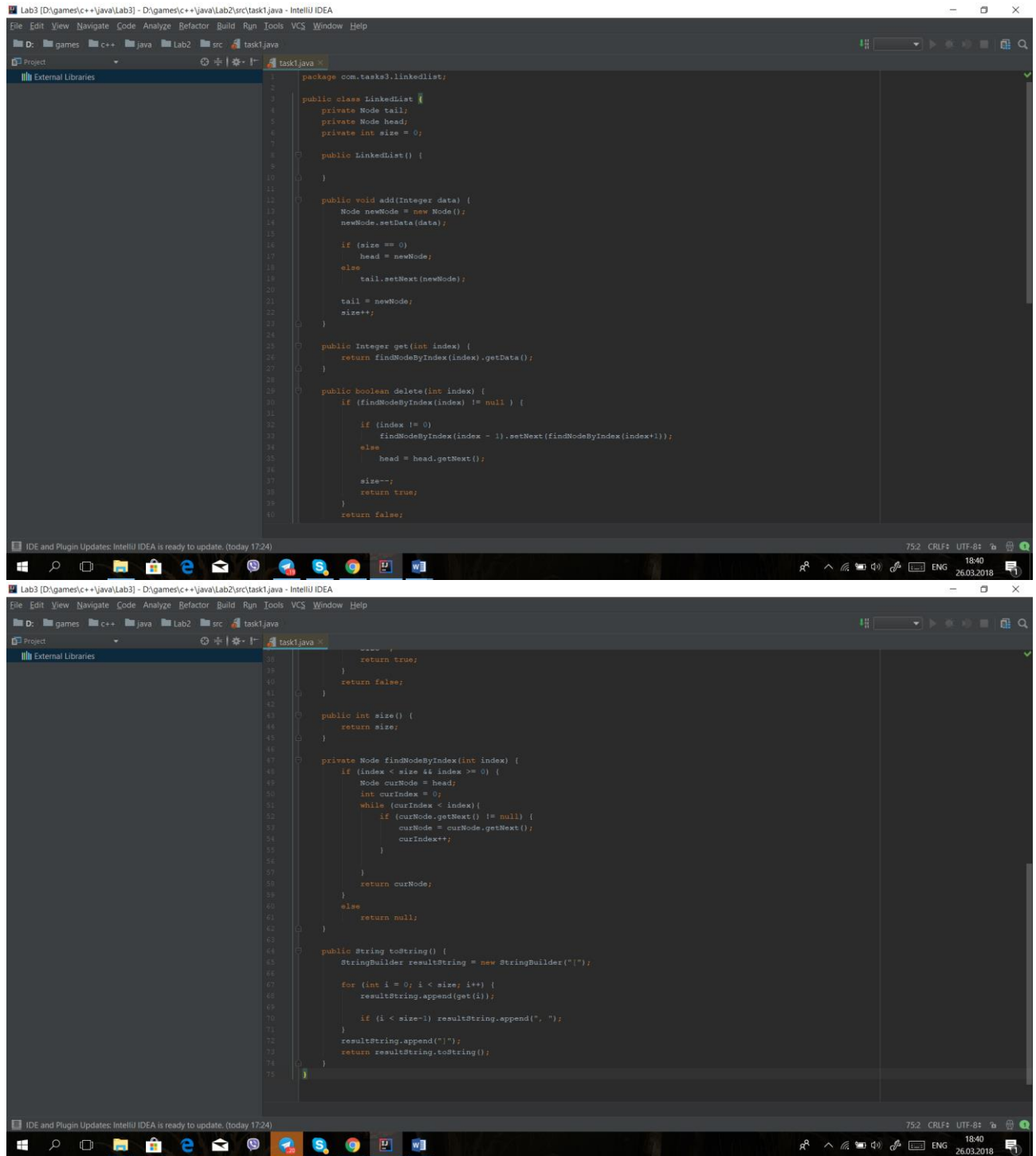
Львів – 2018 р.

Уважно передивившись відео другого тижня на платформі Prometheus я пройшов Тест 3.



Після чого я виконав Практичне завдання 3. Написавши коди у IntelliJ , я закинув їх на перевірку на платформі Prometheus .

Код 1



```
1 package com.tasks3.linkedlist;
2
3 public class LinkedList {
4     private Node tail;
5     private Node head;
6     private int size = 0;
7
8     public LinkedList() {
9     }
10
11
12     public void add(Integer data) {
13         Node newNode = new Node();
14         newNode.setData(data);
15
16         if (size == 0)
17             head = newNode;
18         else
19             tail.setNext(newNode);
20
21         tail = newNode;
22         size++;
23     }
24
25     public Integer get(int index) {
26         return findNodeByIndex(index).getData();
27     }
28
29     public boolean delete(int index) {
30         if (findNodeByIndex(index) != null) {
31
32             if (index != 0)
33                 findNodeByIndex(index - 1).setNext(findNodeByIndex(index + 1));
34             else
35                 head = head.getNext();
36
37             size--;
38             return true;
39         }
40         return false;
41     }
42
43     public int size() {
44         return size;
45     }
46
47     private Node findNodeByIndex(int index) {
48         if (index < size && index >= 0) {
49             Node curNode = head;
50             int curIndex = 0;
51             while (curIndex < index) {
52                 if (curNode.getNext() != null) {
53                     curNode = curNode.getNext();
54                     curIndex++;
55                 }
56             }
57             return curNode;
58         }
59         else
60             return null;
61     }
62
63     public String toString() {
64         StringBuilder resultString = new StringBuilder("");
65
66         for (int i = 0; i < size; i++) {
67             resultString.append(get(i));
68
69             if (i < size - 1) resultString.append(", ");
70         }
71         resultString.append("\n");
72         return resultString.toString();
73     }
74 }
75 }
```

Код 2

```
Lab3 [D:\games\c++\java\Lab3] - D:\games\c++\java\Lab2\src\task1.java - IntelliJ IDEA
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
D: games c++ java Lab2 src task1.java
Project task1.java
External Libraries
package com.tasks3.carddeck;

public class Deck {

    private static final int DECKSIZE = 36;
    private int decksize = 36;

    Card[] deck = new Card[DECKSIZE];

    public Deck() {
        for (int i = 0; i < 4; i++) {
            for (int j = 0; j < 9; j++) {
                deck[i*9+j] = new Card(Rank.values[j], Suit.values[i]);
            }
        }
    }

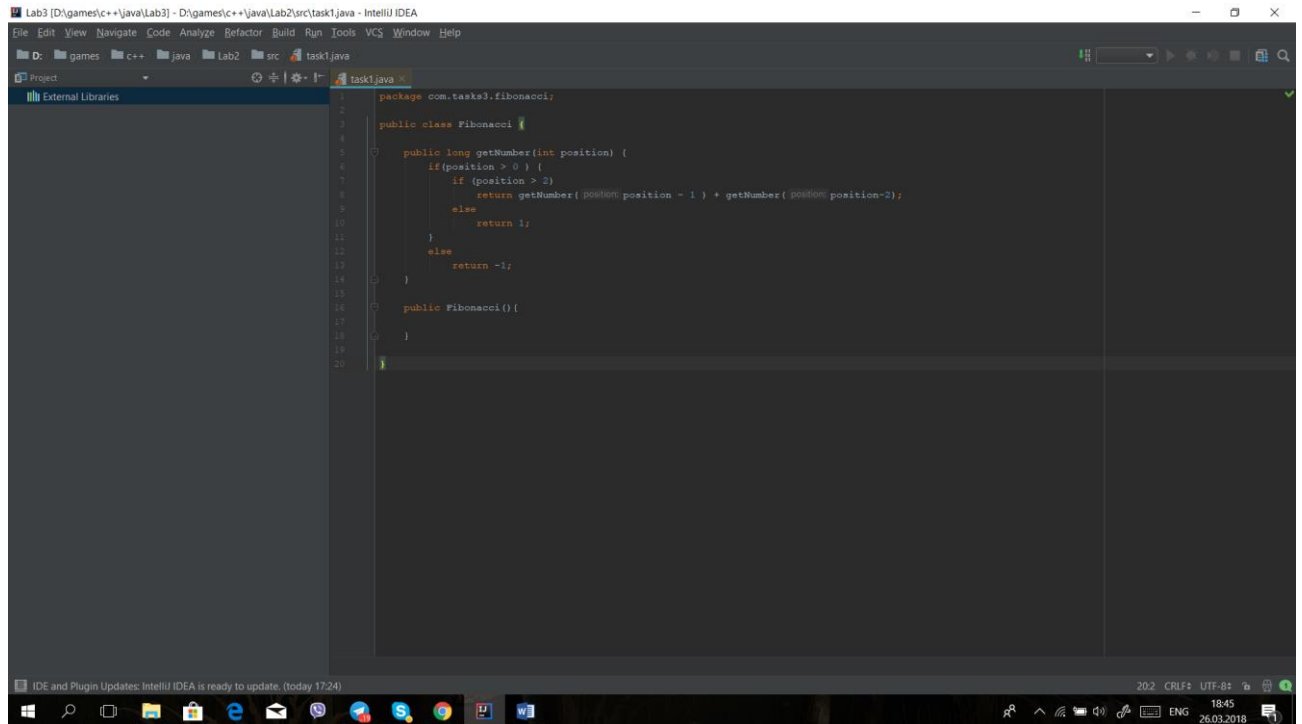
    public void shuffle() {
        Card temp = new Card(Rank.ACE, Suit.CLUBS);
        for (int i = 0; i < Math.random()*100; i++) {
            int randomNumberA = (int) (Math.random() * DECKSIZE);
            int randomNumberB = (int) (Math.random() * DECKSIZE);
            temp = deck[randomNumberA];
            deck[randomNumberA] = deck[randomNumberB];
            deck[randomNumberB] = temp;
        }
    }

    /* * Пропажування колоди за мастями та значенням
     * Порядок сортування:
     * Спочатку всі карти в мастях HEARTS, потім DIAMONDS, CLUBS, SPADES
     * Для кожної масті порядок наступний: Ace, King, Queen, Jack, 10, 9, 8, 7, 6
     * Наприклад:
     * HEARTS Ace
     * HEARTS King
     * HEARTS Queen
     * HEARTS Jack
     * HEARTS 10
     * HEARTS 9
     * HEARTS 8
     */
    // Пропажування колоди за мастями та значенням
    public void order() {
        for (int i = 0; i < 4; i++) {
            for (int j = 0; j < 9; j++) {
                deck[i*9+j] = new Card(Rank.values[j], Suit.values[i]);
            }
        }
    }

    // Повертає true у випадку коли в колоді ще доступні карти
    public boolean hasNext() {
        if (decksize > -1) {
            return true;
        }
        else return false;
    }

    // Повертає одну карту з колоди, коли буде видано всі 36 карт повертає null
    // Карти виділяються з "вершини" колоди. Наприклад першою вийде карта SPADES 6 потім
    // SPADES 7, ..., CLUBS 6, ..., CLUBS Ace і так далі до HEARTS Ace
    public Card drawOne() {
        decksize--;
        if (decksize > -1) {
            return deck[decksize];
        }
        else return null;
    }
}
```

Код 3



The screenshot shows the IntelliJ IDEA IDE interface. The top menu bar includes File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, and Help. The project structure on the left shows a project named 'task1.java' with a package 'com.tasks3.fibonacci'. The main editor displays the following Java code:

```
1 package com.tasks3.fibonacci;
2
3 public class Fibonacci {
4
5     public long getNumber(int position) {
6         if(position > 0 ) {
7             if (position > 2)
8                 return getNumber( position-1 ) + getNumber( position-2);
9             else
10                return 1;
11         }
12         else
13             return -1;
14     }
15
16     public Fibonacci() {
17
18     }
19
20 }
```

The status bar at the bottom indicates 'IDE and Plugin Updates: IntelliJ IDEA is ready to update. (today 17:24)', '202 CRLF UTF-8', and the date '18:45 26.03.2018'.

Коди доступні на GitHub