## МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

### Кафедра систем штучного інтелекту



# Лабораторна робота №3

з дисципліни

«Алгоритмізація та програмування»

#### Виконав:

студент групи КН-109

Питель А.

Викладач:

Гасько. Р.Т.

## КОД

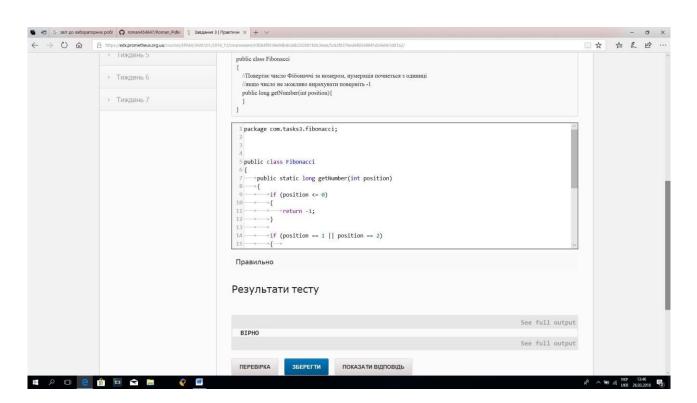
1. package com.tasks3.linkedlist;

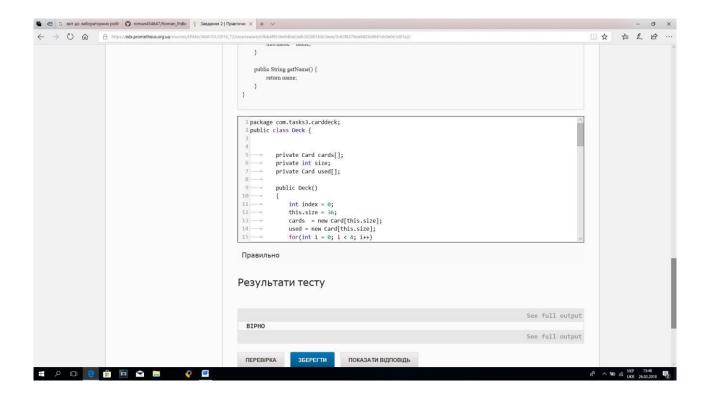
```
public class LinkedList { private
Node begin; private Node end;
private int num_ofEllem;
//конструктор без аргументів
public LinkedList() { begin = new
Node(); end = begin;
this.num_ofEllem = 0;
}
/* Додати елемент в кінець списку */
public void add(Integer data) {
end.setNext(new Node()); end =
end.getNext(); end.setData(data);
this.num_ofEllem++;
}
/* Отримати елемент по індексу, повертає null якщо такий елемент недоступний
public Integer get(int index) { Node
tbegin = begin;
for(int i = 0; i < index + 1; i++)
{
tbegin = tbegin.getNext();
}
return tbegin.getData();
}
/* Вилучення елементу за індексом, повертає true у разі успіху або false в
   іншому випадку */
public boolean delete(int index) { Node
tbegin = begin; if ((index < 0)||(index >=
this.num_ofEllem)) return false; for (int i = 0;
i < index; i++) { if (tbegin != null) tbegin =
tbegin.getNext(); else
return false;
}
if(tbegin.getNext().getNext() != null)
{
tbegin.setNext(tbegin.getNext().getNext());
}
```

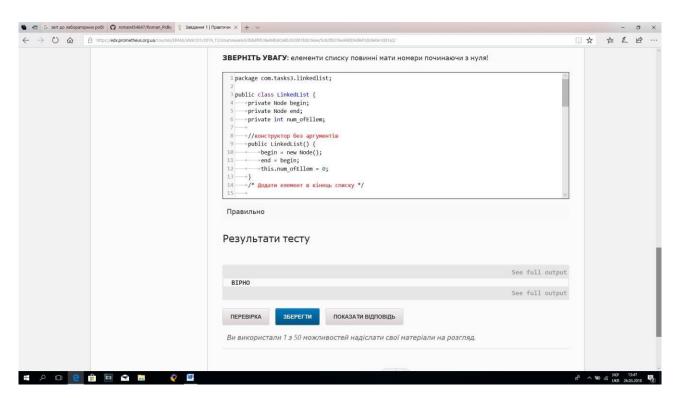
```
else
 tbegin.setNext(null); end
= tbegin;
 }
 num_ofEllem--; return
true;
 }
 /*Поверта розмір списку: якщо елементів в списку нема то повертає 0
   (нуль)*/ public int
size() { return
this.num_ofEllem;
 }
 }
 3. card package
com.tasks3.carddeck; public
class Deck { private Card cards[];
private int size; private Card
used[]; public Deck()
 {
 int index = 0; this.size = 36;
cards = new Card[this.size];
used = new Card[this.size];
for(int i = 0; i < 4; i++)
 {
 for(int j = 0; j < 9; j++)
 {
 3 this.cards[index++] = new Card(Rank.values[j], Suit.values[i]);
 }
 }
 }
 public void shuffle() { for(int i =
0; i < this.size; i++) {
 int random_num = (int)(Math.random()*this.size);
Card tmp = this.cards[i]; this.cards[i] =
this.cards[random_num]; this.cards[random_num]
= tmp;
 }
 }
```

```
public void order() { int
index = 0; for(int i = 0; i
< 4; i++)
 {
 for(int j = 0; j < 9; j++)
 {
 boolean create = true; for(int
k = size; k < 36; k++)
 if((used[k].getRank() == Rank.values[i])&&(used[k].getSuit() == Suit.values[j]))
create = false;
 if(create) { this.cards[index].setRank(Rank.values[j]);
this.cards[index].setSuit(Suit.values[i]);
 index++;
 }
 }
 }
 }
 public boolean hasNext() {
if(size != 0) return true;
else
 return false;
 public Card drawOne() { size--; if(size >= 0) { used[size] = new
Card(cards[size].getRank(), cards[size].getSuit()); return cards[size];
 } else
 return null;
 }
 3.
       fibonacci
                          package
com.tasks3.fibonacci;
                            public
class Fibonacci
 {
 public static long getNumber(int position)
 {
 if (position <= 0)
 {
```

```
return -1;
}
if (position == 1 | | position == 2)
{    return
1;
}
long[] nums = new long[position];    nums[0] =
nums[1] = 1;
for (int i = 2; i < position; i++)
{
    nums[i] = nums[i - 1] + nums[i - 2];
}
return nums[position - 1];
}
</pre>
```







Висновок я вивчив основи ооп на мові програмування JAVA