Міністерство освіти, науки, молоді та спорту України
Національний університет «Львівська політехніка»

Кафедра СШІ

Лабораторна робота №3

Виконав:
ст. групи КН-107
Куликович Д.С.
Прийняв:
Старший викладач кафедри
СШІ
Гасько Р.Т.

Львів-2018

}

Хід роботи

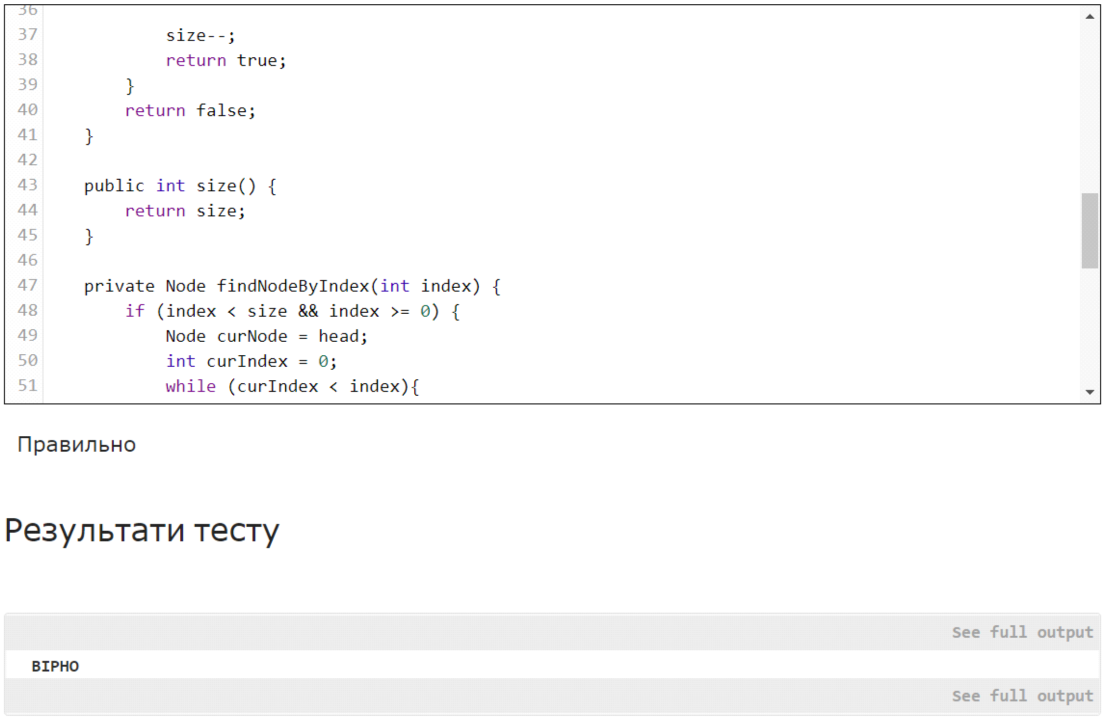
Код 1:

```
package com.tasks3.linkedlist;

public class LinkedList {
    private Node tail;
    private Node head;
    private int size = 0;

    public LinkedList() {

    }

    public void add(Integer data) {
        Node newNode = new Node();
        newNode.setData(data);

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

        tail = newNode;
        size++;
    }

    public Integer 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 findNodeByIndex(int index) {
    if (index < size && index >= 0) {
        Node 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();
    }
}
```

Результат на Prometheus:



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

    public int size() {
        return size;
    }

    private Node findNodeByIndex(int index) {
        if (index < size && index >= 0) {
            Node curNode = head;
            int curIndex = 0;
            while (curIndex < index){
```

Правильно

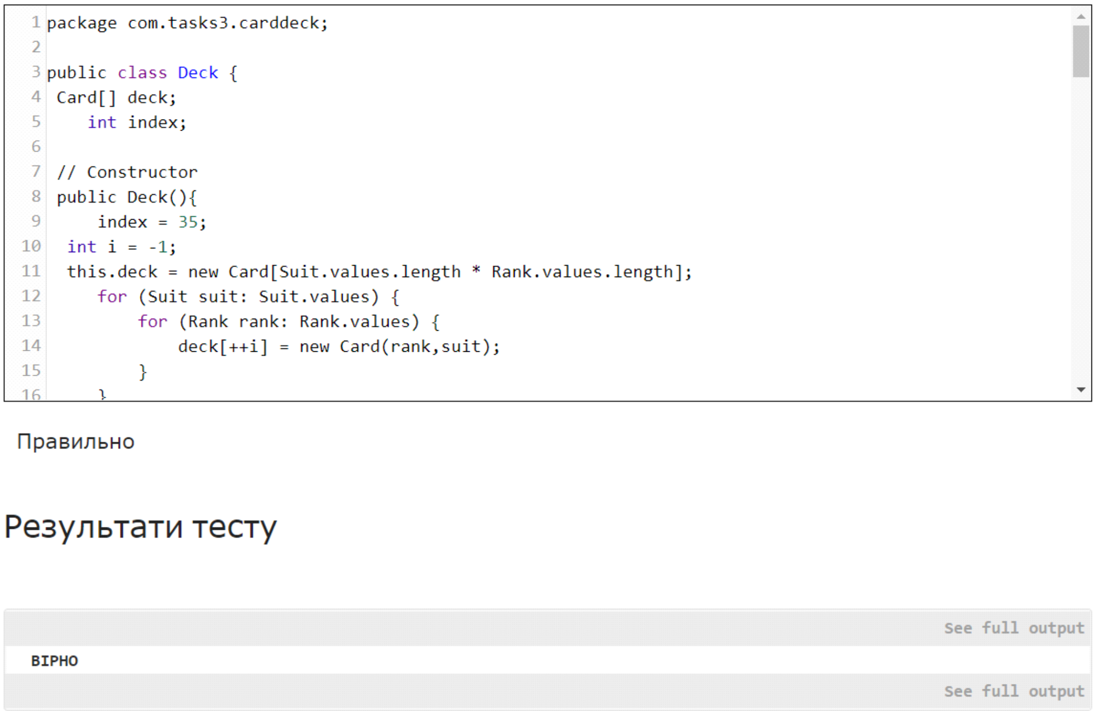
Результати тесту

ВІРНО

Код 2:

```
package com.tasks3.carddeck;

public class Deck {
 Card[] deck;
   int index;

 // Constructor
 public Deck(){
    index = 35;
 int i = -1;
 this.deck = new Card[Suit.values.length * Rank.values.length];
    for (Suit suit: Suit.values) {
       for (Rank rank: Rank.values) {
           deck[++i] = new Card(rank,suit);
       }
    }
}

   public void shuffle() {
    Card tmpCard;
    int q = (int) (Math.random()*300+700);
    for (int n=0; n<q; n++){
     int l = (int) (Math.random()*32);
     int m = (int) (Math.random()*(36-l)+l);
     tmpCard = this.deck[m];
     this.deck[m] = this.deck[m-l];
     this.deck[m-l] = tmpCard;
    }
   }
      public void order() {
 index = 35;
    int i = -1;
 this.deck = new Card[Suit.values.length * Rank.values.length];
    for (Suit suit: Suit.values) {
       for (Rank rank: Rank.values) {
           deck[++i] = new Card(rank,suit);
       }
    }
   }

      public boolean hasNext() {
    return index > -1;
   }
          public Card drawOne() {
    if (index >= 0)
       return this.deck[index--];
       else return null;
   }

   public void deckPrn(){
    System.out.println("---Cards in deck-"+(index+1)+"---");
    for (int j=0; j<index+1; j++){
  System.out.print("|");
  System.out.print(this.deck[j].getSuit().getName());
  System.out.print(" ");
  System.out.println(this.deck[j].getRank().getName());
    }
   }

   public void cardPrn(Card crd){
    if (crd != null){
     System.out.print(crd.getSuit().getName());
     System.out.print(" ");
     System.out.println(crd.getRank().getName());
    }
    else System.out.println("Card is NULL");
   }


public static void main(String[] args) {

 Deck dk1 = new Deck();

 dk1.shuffle();
 dk1.deckPrn();
 //dk1.order();
 //dk1.cardPrn(dk1.deck[10]);
 /*
 for (int t=0; t<36; t++){
  dk1.cardPrn(dk1.drawOne());
  System.out.println("----------------------");
 }
 System.out.println(dk1.hasNext());
 dk1.deckPrn();
 */
 }

}
```

Результат на Prometheus:



```
package com.tasks3.carddeck;

public class Deck {
 Card[] deck;
    int index;

 // Constructor
 public Deck(){
     index = 35;
  int i = -1;
  this.deck = new Card[Suit.values.length * Rank.values.length];
     for (Suit suit: Suit.values) {
         for (Rank rank: Rank.values) {
             deck[++i] = new Card(rank,suit);
         }
     }
```

Правильно

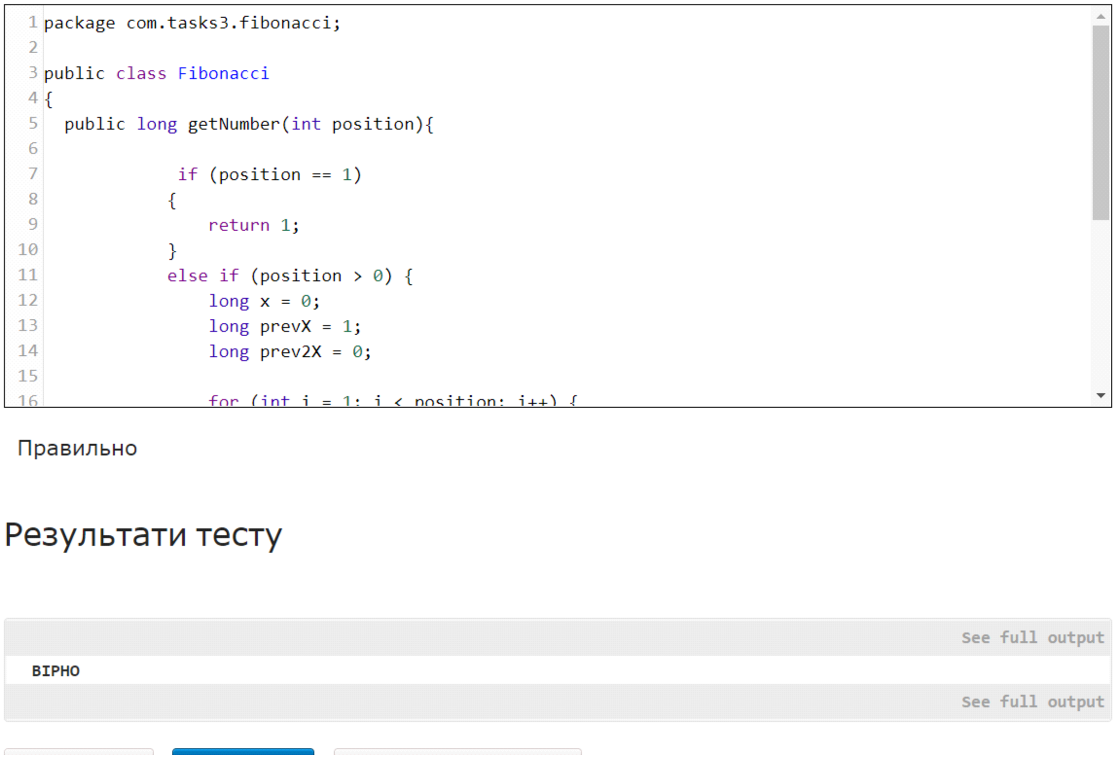
Результати тесту

ВІРНО

Код 3:

```
package com.tasks3.fibonacci;

public class Fibonacci
{
 public long getNumber(int position){

         if (position == 1)
        {
           return 1;
        }
        else if (position > 0) {
           long x = 0;
           long prevX = 1;
        long prev2X = 0;

        for (int i = 1; i < position; i++) {
          x = prevX + prev2X;
          prev2X = prevX;
          prevX = x;
        }

        return x;
      }
      else
      {
        return -1;
      }
 }
}
```

Результат на Prometheus:



```
package com.tasks3.fibonacci;

public class Fibonacci
{
  public long getNumber(int position){

             if (position == 1)
            {
                return 1;
            }
            else if (position > 0) {
                long x = 0;
                long prevX = 1;
                long prev2X = 0;

                for (int i = 1; i < position; i++) {
```

Правильно

Результати тесту

ВІРНО