

## Лабораторна робота № 6 ООП

Виконав:  
ст. групи КН-107  
Тимків А.І.  
Прийняв:  
Старший викладач  
СШІ  
Гасько Р.Т.

**Завдання 1** Створити застосування, що упакує рядки за допомогою алгоритму RLE. У випадку Вашого застосування потрібно замінити послідовність однакових букв на букву за якою буде слідувати цифра від 2 до 9, що вкаже на кількість повтору символів. Якщо символ зустрівся лише 1 раз то заміну робити не потрібно. Регістр символів при заміні враховується. Код:

```
package com.tasks6.rle;
```

```
public class Application
```

```
{
```

```
    public static void main( String[] args ) {
```

```
        int count = 0;
```

```
        String rle = "";
```

```
        int numb;
```

```
        if (args.length>0) {
```

```
            String s = args[0];
```

```
            while (count<s.length()) {
```

```
                numb =1;
```

```
                char cur = s.charAt(count);
```

```
                char next = 0;
```

```
                if (count+1<s.length()) {
```

```
                    next = s.charAt(count + 1);
```

```
                }
```

```
                else {rle+=cur; break;}
```

```
                if (cur ==next) {
```

```
                    rle+=cur;
```

```
                    while (cur==next&&Character.isLetter(cur)&&Character.isLetter(next)) {
```

```
                        numb++;
```

```
                        count++;
```

```
                        cur=next;
```

```
                        if (count+1<s.length())
```

```
                            next = s.charAt(count + 1);
```

```
                        else break;
```

```
                    }
```

```
                    if (numb>=2&&numb<=9) {
```

```
                        rle+=numb;
```

```
                    }else if (numb>9) {
```

```
                        rle+=9;
```

```
                        rle+=rle.charAt(rle.length()-2);
```

```
                        rle+=numb-9;
```

```
                    }
```

```
                    count++;
```

```
                }else{
```

```
                    rle+=cur;
```

```
                    count++;
```

```
                }
```

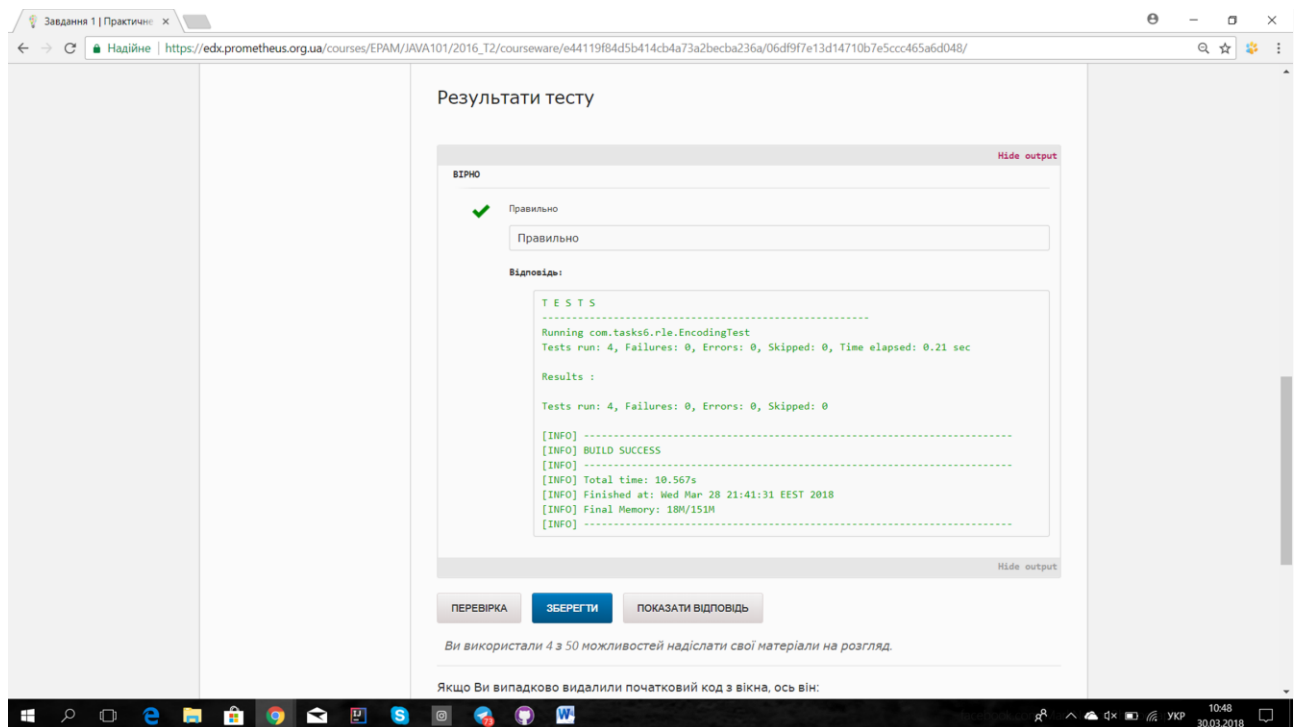
```
            }
```

```
            System.out.println(rle);
```

```
        }
```

```
    }
```

```
}        Р е з у л ь т а т :
```



## Завдання 2

Створіть застосування, що буде декодувати рядки отримані в завданні з кодуванням рядків.

Код:

```
package com.tasks6.rle_decoder;
```

```
public class Application
{
    public static void main( String[] args )
    {
        int count = 1;
        int size = args[0].length();
        String rle = "";
        char next;
        boolean pass= true;
        if (args.length>0) {
            if (size==0)
                System.out.println(rle);
            else {
                for (int i = 0; i < size-1; i++)
                    if (args[0].charAt(i)==args[0].charAt(i+1)) {
                        pass = !pass;
                        break;
                    }
                if (pass==true) {
                    if (!Character.isDigit(args[0].charAt(0))) {
                        args[0].charAt(0);
                        rle += args[0].charAt(0);
                        for (count = 1; count < size; count++) { //перебіркожного елемента
```

стрічки

The screenshot shows a web browser window with the address bar displaying the URL: [https://edx.prometheus.org.ua/courses/EPAM/JAVA101/2016\\_T2/courseware/e44119f84d5b414cb4a73a2becba236a/06df9f7e13d14710b7e5ccc465a6d048/](https://edx.prometheus.org.ua/courses/EPAM/JAVA101/2016_T2/courseware/e44119f84d5b414cb4a73a2becba236a/06df9f7e13d14710b7e5ccc465a6d048/). The page content is in Ukrainian and indicates a successful test result.

Правильно

## Результати тесту

**ВІРНО** Hide output

✓ Правильно

**Відповідь:**

```
T E S T S
-----
Running com.tasks6.rle_decoder.DecodingTest
Tests run: 9, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.215 sec

Results :

Tests run: 9, Failures: 0, Errors: 0, Skipped: 0

[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 11.297s
[INFO] Finished at: Thu Mar 29 17:50:00 EEST 2018
[INFO] Final Memory: 16M/154M
[INFO] -----
```

Hide output

Ви використали 5 з 50 можливостей надіслати свої матеріали на розгляд

The bottom of the image shows a Windows taskbar with various application icons and a system clock indicating 10:50 on 30.03.2018.

```

        for(int j = 0; j < Rank.values.length; j++) {
            deck[k] = new Card(Rank.values[j], Suit.values[i]);
            k++;
        }
    }
}

```

```

public void shuffle() {
    if(currentCardNumber >= 0){
        Card[] tempDeck = new Card[1];
        for(int k = 0; k < currentCardNumber + 1; k++){
            int a = random(currentCardNumber + 1);
            tempDeck[0] = deck[k];
            deck[k] = deck[a];
            deck[a] = tempDeck[0];
        }
    }
}

```

```

public void order() {
    Card[] tempDeck = new Card[deck.length];
    int n = 0;
    for(int i = 0; i < Suit.values.length; i++) {
        for(int j = 0; j < Rank.values.length; j++) {
            for(int k = 0; k <= currentCardNumber; k++){

```

```

        if(deck[k].getSuit().getName().equals(Suit.values[i].getName()) &&
deck[k].getRank().getName().equals(Rank.values[j].getName())) {

            tempDeck[n] = deck[k];

            n++;

        }

    }

}

}

deck = tempDeck;

}

```

```

public boolean hasNext() {

    return currentCardNumber >= 0;

}

```

```

public Card drawOne() {

    if(hasNext()){

        return deck[currentCardNumber--];

    }

    return null;

}

```

```

public static int random(int deckLength){

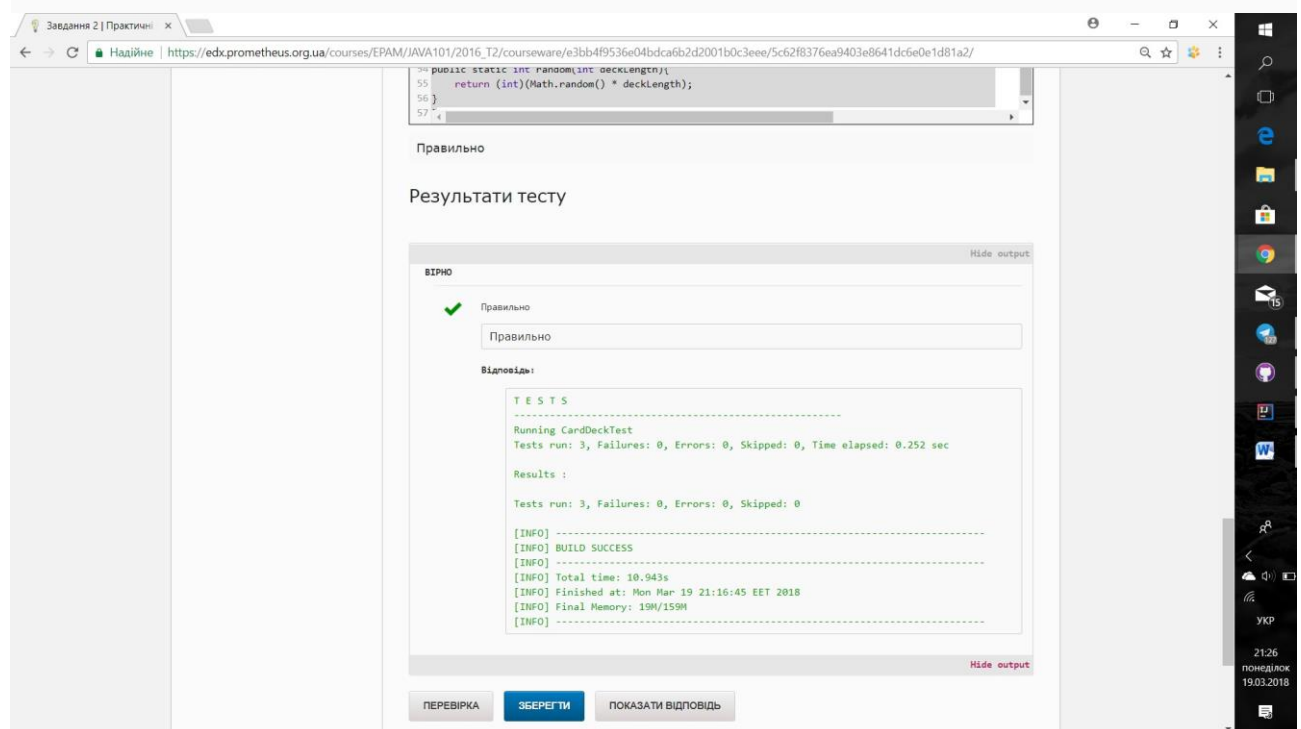
    return (int)(Math.random() * deckLength);

}

}

```

Результат:



**Завдання 3** Використовуючи рекурсію, виведіть на екран задане по порядковому номеру число Фібоначі.

```
package com.tasks3.fibonacci;
```

```
public class Fibonacci {
```

```
    public Fibonacci() {}
```

```
    public int getNumber(int position) { if(position
        > 0 && position < 93) {
        if (position > 2)
            return getNumber(position - 1 ) + getNumber(position-2);
        else
            return 1;
        }
        else
            return -1;
    }
}
```

Результат:

