

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»**

**Інститут прикладної математики і фундаментальних наук**

**Кафедра прикладної математики**



**Лабораторна робота №7**  
**з курсу “Програмування настільних**  
**застосунків”**  
**Тема: “Робота з символьними рядками в Java.”**

Виконав студент групи ПМ-33  
Венгринюк Олег  
Прийняла  
Терендій О. В.

### **Завдання для лабораторної роботи:**

1. Ознайомитись з теоретичним матеріалом.
2. Дати відповідь на поставлені питання.
3. Розробити алгоритм розв'язання індивідуального завдання.
4. Скласти програму на мові Java у відповідності з розробленим алгоритмом.
5. Виконати обчислення по програмі.

### **Перелік контрольних запитань:**

1. Які класи існують в Java для роботи зі стрічками, і чим вони відрізняються один від одного?
2. Якими способами можна створити стрічку класу String в Java?
3. Які особливості реалізації стрічок в Java необхідно враховувати при порівнянні стрічок?
4. Які методи порівняння стрічок визначені в Java?
5. Які методи пошуку символів і підстрічок в стрічках визначені в Java?
6. Як можна створити нову стрічку класу String на основі існуючої стрічки?
7. Як можна передати параметри програмі Java?
8. В якому вигляді передаються ці параметри?

Відповіді на контрольні запитання:

1. String, StringBuilder, StringBuffer. Оскільки String є незмінними (immutable) типом даних, то після кожної маніпуляції над змінною старе значення залишається збирачу сміття і виділяється пам'ять під нове значення. В процесі маніпуляції над стрічками виділяється багато пам'яті і багато пам'яті звільняється. Тому для такого роду завдань є класи StringBuffer і StringBuilder, що є змінними типами даних. Вони забезпечують наступні методи: `append()`, `insert()`, `delete()`, `substring()`. StringBuffer – був розроблений раніше і всі його публічні методи працюють синхронізовано в потоці, за що приходится платити продуктивністю. StringBuilder – напрочуд, клас що появився відносно недавно і працює асинхронно і продуктивніше. Загальна рекомендація така: якщо програма працює зі стрічкою в одному потоці або не важлива безпечність виконання потоку – використовувати StringBuilder, оскільки він продуктивніший. Якщо важливе безпека потоку – StringBuffer.
2. Якими способами можна створити стрічку класу String в Java?
  - Створення літералу – стрічка потрапляє до пулу стрічок.
  - Створення явно стрічки за допомогою оператора `new`, приклад `new String(„Some String“);`
  - Створення об'єкта типу String за допомогою оператора `new` з параметром для конструктора – списку символів. `String char[] = „Some String“; new String(string);`
3. оператор рівності `==` порівнює чи однакові посилання у змінних. `String.equals` – поверне `true` у випадку позиційної рівності символів стрічки, інакше результат порівняння `false`. Порівняння відбувається лексикографічно та з урахуванням велика літера чи мала.
4. `compareTo`, `equals`, `equalsIgnoreCase`
5. `indexOf`, `contains`
6. За допомогою конструктора і в якості параметра передати стару стрічку.
7. Стрічкового типу
8. Як масив стрічок.

### Програмний код:

```
import java.util.Arrays;
public class Main {
    public static void main(String [] args){
        String s = "Some words in some order";
        char c = 'A';
        int k = 2;
        String words[] = s.split(" ");
        String result[] = new String[words.length];
        for (int i=0; i<words.length; ++i) {
            result[i] = replace(words[i], c, k);
        }
        String resultString = String.join(" ", result);
        System.out.println(resultString);
    }
    public static void testString(){
    }
    public static String replace(String s, char c, int k){
        StringBuilder sb = new StringBuilder(s);
        for(int i=0; i< s.length(); ++i){
            if((i+1)%k != 0)
                continue;
            sb.setCharAt(i, c);
        }
        return sb.toString();
    }
}
```

### Результат виконання роботи:

```
SAmA wArAs iA sAmA oAdAr
```

```
Process finished with exit code 0
```

**Висновок:** в ході виконання лабораторної роботи було опановано наступні типи даних для роботи зі стрічками: String, StringBuffer, StringBuilder. Було надано відповіді на контрольні питання та виконано завдання згідно з варіантом.