

Міністерство освіти і науки України
Національний університет «Львівська політехніка»



Лабораторна робота №8
з курсу:
«Об'єктно-орієнтоване програмування»

Виконала:
ст. гр. КН-110
Славчаник Олеся
Прийняв:
Гасько Р. Т.

Лабораторна робота № 8

Мета роботи: • Розробка власних утилітарних класів.

• Набуття навичок вирішення прикладних задач з використанням масивів і рядків.

• Реалізація діалогового режиму роботи з користувачем в консольних програмах мовою Java.

Розробник: дивитися вище, хто виконав роботу. *Варіант 1*

Вимоги:

1. Розробити та продемонструвати консольну програму мовою Java в середовищі Eclipse для вирішення прикладної задачі за номером, що відповідає збільшеному на одиницю залишку від ділення на 15 зменшеного на одиницю номера студента в журналі групи.

2. Використовуючи програму рішення завдання відповідно до прикладної задачі забезпечити обробку команд користувача у вигляді текстового меню :

a. введення даних;

b. перегляд даних;

c. виконання обчислень;

d. відображення результату;

e. завершення програми і т.д.

3. Забезпечити обробку параметрів командного рядка для визначення режиму роботи програми:

a. параметр “-h” чи “-help”: відображається інформація про

*автора програми, призначення (індивідуальне завдання),
детальний опис режимів роботи (пунктів меню та параметрів
командного рядка);*

*б. параметр “-d” чи “-debug”: в процесі роботи програми
відображаються додаткові дані, що полегшують
налагодження та перевірку працездатності програми:
діагностичні повідомлення, проміжні значення змінних,
значення тимчасових змінних та ін.*

4. При вирішенні прикладних задач використовувати латинку .

*5. Продемонструвати використання об'єктів класу `StringBuilder` або
`StringBuffer` .*

*6. Застосувати функціональну (процедурну) декомпозицію - розробити
власні утилітарні класи (особливий випадок допоміжного класу, див.
`Helper Class`) та для обробки даних використовувати відповідні
статичні методи.*

*7. Забороняється використовувати засоби обробки регулярних
виразів: класи пакету `java.util.regex` (`Pattern` , `Matcher` та ін.), а також
відповідні методи класу `String` (`matches` , `replace` , `replaceFirst` , `replaceAll` ,
`split`).*

Задача: Ввести текст. Текст розбити на речення. Для кожного речення знайти та надрукувати всі слова максимальної та всі слова мінімальної довжини. Результат вивести у вигляді таблиці.

```
public class Test {  
  
    public static void main(String[] args) {  
        if (args != null && args.length != 0 && args[0] != null){  
            //StringBuilder max = new StringBuilder("");  
        }  
    }  
}
```

```

String res = new StringBuilder("");
String max1 = args[0];
String min1 = args[0];
int n = 0, k = 0, q = 0;
if (args[0].contentEquals("-h") || args[0].contentEquals("-help"))
    System.out.println("Olesia Slavchanyk, CS-110\nThis program returns the shortest and longest word from each
sentence\n");

else if (args[0].contentEquals("-d") || args[0].contentEquals("-debug"))
    System.out.println(res + "Debug - done");
else {
    if (args[0].endsWith(".")) {
        res.append(args[0].substring(0, args[0].length() - 1));
        res.append("\t");
        res.append(args[0].substring(0, args[0].length() - 1));
        res.append("\n");
    } else {
        for (int i = 0; i < args.length; i++) {
            if (args[i].endsWith(".") || args[i].endsWith(",") || args[i].endsWith("\n") || args[i].endsWith(")")) {
                if (args[i].length() - 1 < min1.length()) {
                    min1 = args[i].substring(0, args[i].length() - 1);
                    k = i;
                }
            } else if (args[i].length() < min1.length()) {
                min1 = args[i];
                k = i;
            }
        }

        if (args[i].endsWith(".") || args[i].endsWith(",") || args[i].endsWith("\n") || args[i].endsWith(")")) {
            if (args[i].length() - 1 > max1.length()) {
                max1 = args[i].substring(0, args[i].length() - 1);
                q = i;
            }
        } else if (args[i].length() > max1.length()) {
            max1 = args[i];
            q = i;
        }
    }

    if (args[i].endsWith(".") || args[i].endsWith("!") || args[i].endsWith("?")) {
        for (int j = n; j <= i; j++) {
            if (args[j].endsWith(".") || args[j].endsWith(",") || args[j].endsWith("\n") ||
args[j].endsWith(")")) {
                if (args[j].length() - 1 == args[k].length() && j != k) {
                    min1 += "\n" + args[j].substring(0, args[j].length() -
1);
                }
            } else if (args[j].length() == args[k].length() && j != k) {
                min1 += "\n" + args[j];
            }
        }

        if (args[j].endsWith(".") || args[j].endsWith(",") || args[j].endsWith("\n") ||
args[j].endsWith(")")) {
            if (args[j].length() - 1 == args[q].length() && j != q) {
                max1 += "\n" + args[j].substring(0, args[j].length() -
1);
            }
        } else if (args[j].length() == args[q].length() && j != q) {
            max1 += "\n" + args[j];
        }
    }
}
res.append(min1);
res.append("\t");
res.append(max1);
res.append("\n");
n = i + 1;
if (i != args.length - 1) {
    min1 = args[i + 1];
    max1 = args[i + 1];
}
}
}
System.out.println(res);
}
}
}

```

}

Висновки: я навчилася розробляти власні утилітарні класи, набула навичок вирішення прикладних задач з використанням масивів та рядків та реалізувала діалоговий режим роботи програми