

# Тема роботи

Утилітарні класи Java SE. Обробка масивів і рядків. Інтерактивні консольні програми для платформи

## Мета

- 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.

## Загальне завдання

### 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);

b. 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`).

## Важливі фрагменти програми

```
public class Counter {
    public int[] how_much(StringBuffer s,int debug){
        int countV = 0, countC = 0;
        char[] vowels={'a','e','y','u','i','o'};
        char[] consonants={'q','w','r','t','p','s','d','f','g','h','j','k','l','z','x','c','v','b','n','m'};
        for (int i=0;i<s.length();i++){
            for (int j=0;j<vowels.length;j++){
                if (vowels[j]==s.charAt(i)){
                    countV++;
                }
            }
            for (int f=0; f<consonants.length;f++){
                if (consonants[f]==s.charAt(i)){
                    countC++;
                }
            }
        }

        int[] letters = {countV, countC};
        if (debug == 1) {

            for (int k = 0; k < vowels.length; k++) {
                System.out.print(vowels[k] + " ");
            }
            for (int k = 0; k < consonants.length; k++) {
                System.out.print(consonants[k] + " ");
            }
        }
        return letters;
    }
}

if (args.length > 0) {
    if (args[0].equals("-h") || args[0].equals("--help") || args[1].equals("-h") || args[1].equals("--help")) {
        help.help();
    }
    if (args[0].equals("-d") || args[0].equals("--debug") || args[1].equals("-d") || args[1].equals("--debug")) {
        debug++;
    }
}
```

## Приклад використання програми

```
**Commands:
    e: Enter text
    v: View text
    p: Process text
    r: Results
    x: Exit
Enter the commands: e
Enter data:
asdsad sa
Enter the commands: v
asdsad sa
Enter the commands: p
Enter the commands: r
Vowels      | 3
_____
Consonants   | 5
Enter the commands:
```