МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

ФАКУЛЬТЕТ ІНФОРМАТИКИ ТА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

КАФЕДРА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

Лабораторна робота №3

з дисципліни «Об’єктно-орієнтоване програмування»

на тему: *«Рядки в мові програмування Java»*

Виконав:

студент гр.ІО-31

Долинний О.В.

Перевірив:

Невдащенко М.В.

Київ 2014

**Лабораторна робота №3**

**Тема:** Рядки в мові програмування Java.

**Мета:** Ознайомлення з рядками та використання основних методів їх обробки в мові програмування Java. Здобуття навичок у використанні рядків в мові програмування Java..

1. ***Варіант завдання***

Номер залікової книжки  *3110*

Тип текстових змінних: *C3 = 3110 mod 3 = 0* String

Дія з рядком: *C17 = 3110 mod 17 = 16*

В кожному слові заданого тексту, видалити всі попередні входження останньої літери цього слова.

Створити клас, який складається з виконавчого методу, що виконує дію текстовим рядком, тип якого визначено варіантом. Необхідно обробити всі виключні ситуації, що можуть виникнути під час виконання програмного коду. Всі змінні повинні бути описані та значення їх задані у виконавчому методі.

1. ***Опис алгоритму програми***

Клас *lab3* містить виконавчий метод, що в кожному слові заданого рядка з видаляє попередні входження останньої літери цього слова.

Спочатку задається рядок та за допомогою метода *trim* з нього видаляються пробіли на початку та в кінці рядка (створено рядок text). Потім текст перевіряється на ситуацію пустого рядка.

Якщо рядок не пустий, створюється нова строка-результат *result*, до якої в коді програми за допомогою конкатенації додаються символи.

Створюється маркер кінця слів *b* (перше входження пробілу в рядок), та маркер *a* початку слова (позиція початку рядка), що обчислюються за допомогою методу *indexOf*. Відбувається перевірка на те, що рядок складається з одного слова (пробілів немає і *b=-1)*. Якщо *b=-1*, то значенню *b* присвоюється позиція кінця рядка.

Кожне отримане слово передається у масив символів *buf*, у якому потім виконується перевірка на наявність розділового знаку на кінці, якщо його немає, то всі попередні літери порівнюються с останньою, якщо є – з передостанньою. При умові проходження перевірки символи масива *buf* заносяться до строки result.

При переході до іншого слова маркери початку *a* та кінця *b* слів здвигаються в бік кінця рядка (використовуючи пошук наступного пробілу), тобто кожне слово визначається положенням маркерів.

Операції повторюються, доки не пройдено останній пробіл. У цьому разі маркер *b* приймається за останню позицію рядка, операція зі словом проводиться в останній раз, формування строки-результату *result* завершується.

Після завершення циклу виводяться результати програми.

1. ***Програмний код***

**public** **class** lab3 {

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

String s = " ivnvvbcv! "; //Введення рядка

String text = s.trim(); // Видаляємо пробіли на початку та в кінці рядка

**if** ( text.length() == 0 ) // Виключна ситуація на введення пустого рядка

System.*out*.println("the text is empty");

**else** // Якщо рядок не пустий

{ String result = **new** String ();// Рядок-результат

**int** a=0; // Маркер-початок слова

**int** b = text.indexOf(" ");// Маркер-кінець слова

**if** (b==-1) // Виключна ситуація з рядком з одного слова

b=text.length();

**while** (b!=-1) // Працюємо, поки маркер b не вийде за границі рядка

{ **char** buf[] = **new** **char** [b-a] ;// Перевід слова до масива символів

text.getChars(a, b, buf, 0) ;

// Ситуація з розділовими знаками на кінці

**if** (buf[buf.length-1]==','||buf[buf.length-1]=='.' || buf[buf.length-1]=='!' || buf[buf.length-1]=='?' ||buf[buf.length-1]==')' || buf[buf.length-1]=='-' || buf[buf.length-1]=='"')

{ **for** (**int** i=0; i<(buf.length-2); i++)

**if** (buf[i]!=buf[buf.length-2])

result+=buf[i];

result=result+buf[buf.length-2]+buf[buf.length-1];

}

**else** // Коли чергове слово без розділових знаків { **for** (**int** i=0; i<(buf.length-1); i++)

**if** (buf[i]!=buf[buf.length-1])

result+=buf[i];

result+=buf[buf.length-1];

}

// Переходимо до наступного слова

a=b;

b=text.indexOf(" ",b+1);

// Ситуація з останнім словом

**if** (a== text.lastIndexOf(" "))

b=text.length();

};

System.*out*.print(result); // Вивід результату

}

}

}

1. ***Висновок***

Під час лабораторної роботи я закріпив навички роботи з рядками в мові Java, використанням логічних умов, перевірки алгоритму на наявність виключних ситуацій та покращив навички коментування коду.

Для полегшення роботи зі словами було використано розбиття слів на масив символів.

Ускладнило роботу програми неможливість зміни рядків String.