НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

ЛАБОРАТОРНА РОБОТА №3

з дисципліни «ООП»

Виконала студентка групи КІТ - 320

Бельчинська Катерина Юріївна

Перевірив викладач Жилін

Володимир Анатолійович

1 ВИМОГИ

* 1. **Розробник**
* Бельчинська Катерина Юріївна
* студентка групи КІТ-320
* номер варіанту - 3
  1. **Загальне завдання**
* Розробка власних утилітарних класів.
* Набуття навичок вирішення прикладних задач з використанням масивів і рядків.
  1. **Задача**

Ввести декілька рядків. Розбити на дві групи: рядки, довжина яких менша за середню; рядки, довжина яких не менше середньої. Вивести рядки та їх довжину по групах.

1. ОПИС ПРОГРАМИ
   1. **Засоби ООП**

Завдання виконувалися в головному класі Main.

* 1. **Ієрархія та структура класів**

.

├── Main.java

└── package-info.java

* 1. **Важливі фрагменти програми**

**package** ua.khpi.oop.belchynska03;

/\*\*

\* Ввести декілька рядків. Розбити на дві групи: рядки, довжина яких менша за середню; рядки, довжина яких не менше середньої. Вивести рядки та їх довжину по групах.

\* **@author** Kate Belchynska

\*

\*/

**public** **final** **class** Main {

/\*\*

\* Строки для викоання завдання

\*/

**private** **static** **final** String ***STR1*** = "Hello";

**private** **static** **final** String ***STR2*** = "my";

**private** **static** **final** String ***STR3*** = "dear";

**private** **static** **final** String ***STR4*** = "friend";

**private** Main() {

}

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

Find\_string m = **new** Find\_string();

StringBuilder str1 = **new** StringBuilder(***STR1***);

StringBuilder str2 = **new** StringBuilder(***STR2***);

StringBuilder str3 = **new** StringBuilder(***STR3***);

StringBuilder str4 = **new** StringBuilder(***STR4***);

**int** quantity = 4;

/\*\*

\* Середня довжина усіх строк

\*/

**int** avarage\_length = m.average\_length(str1.length(), str2.length(), str3.length(), str4.length(), quantity);

/\*\*

\* Масиви для групування строк за довжиною

\*/

StringBuilder max\_str[] = **new** StringBuilder[10];

StringBuilder min\_str[] = **new** StringBuilder[10];

m.find\_group(str1, avarage\_length, 0, 0, min\_str, max\_str);

m.find\_group(str2, avarage\_length, 1, 1, min\_str, max\_str);

m.find\_group(str3, avarage\_length, 2, 2, min\_str, max\_str);

m.find\_group(str4, avarage\_length, 3, 3, min\_str, max\_str);

/\*\*

\* Вивід на екран строк та іх довжину

\*/

Times New Roman System.***out***.println("Strings whith length bigger than average\n");

**for**(**int** i = 0; i < 4; i++) {

**if**(max\_str[i] != **null**) {

System.***out***.println(max\_str[i] + "; string length:" + max\_str[i].length() + "\n");

}

}

/\*\*

\* Вивід на екран строк та іх довжину

\*/

System.***out***.println("Strings whith length smaller than average\n");

**for**(**int** i = 0; i < 4; i++) {

**if**(min\_str[i] != **null**) {

System.***out***.println(min\_str[i] + "; string length:" + min\_str[i].length() + "\n");

}

}

}

}

/\*\*

\* Клас для визначення середньої довжини та розподілу строк у необхідний масив

\* **@author** kate

\*

\*/

**class** Find\_string {

**public** **void** find\_group(StringBuilder str, **int** average\_length, **int** pos\_min, **int** pos\_max, StringBuilder min\_str[], StringBuilder max\_str[]) {

**if**(str.length() < average\_length) {

get\_string( str, pos\_min, min\_str);

}

**else** {

get\_string( str, pos\_max, max\_str);

}

}

**public** StringBuilder[] get\_string( StringBuilder str, **int** max\_string\_value, StringBuilder max\_min\_str[]) {

max\_min\_str[max\_string\_value] = str;

**return** max\_min\_str;

}

**public** **int** average\_length(**int** l1, **int** l2, **int** l3, **int** l4, **int** quantity) {

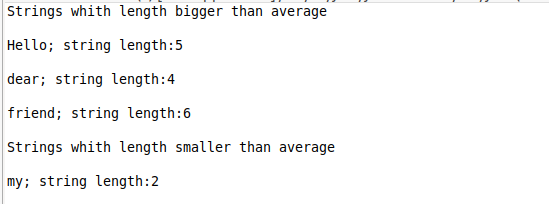
**int** average = (l1 + l2 + l3 + l4) / quantity;

**return** average;

}

}

3 ВАРІАНТИ ВИКОРИСТАННЯ

Рисунок 1. Результат виконання програми

ВИСНОВКИ

В ході виконання даної роботи були набуті навички вирішення прикладних задач з використанням масивів і рядків та розроблені власні утилітарні класи.