НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

ФАКУЛЬТЕТ ІНФОРМАТИКИ ТА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

КАФЕДРА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

**Лабораторна робота №3**

з дисципліни **«**Основи технологій програмування**»**

Варіант: «6412»

Виконав:

студент 2 курсу

група ІП-64

Лейзьо Сергій Іванович

Перевірив:

Подрубайло Олександр Олександрович

Київ – 2018

**Завдання**

1. Визначити C3 як остачу від ділення номера залікової книжки студента на 3

6412%3=1

**StringBuffer**

1. C13 як остачу від ділення номера залікової книжки студента на 17.

С13=6412%13= 3

**В усіх питальних реченнях заданого тексту знайти та надрукувати без повторень слова заданої довжини.**

**Код програми**

package com.company;  
  
import java.util.Scanner;  
  
public class Main {  
  
 public static void main(String[] args) {  
 /\*  
 С3=1 StringBuffer  
 С13=3 В усіх питальних реченнях заданого тексту знайти та надрукувати без  
 повторень слова заданої довжини.  
 \*/  
 StringBuffer text = new StringBuffer();  
 Scanner scan = new Scanner(System.*in*);  
 System.*out*.println(" Введите текст: ");  
 text.append(scan.nextLine());  
 int counter = 0;//счетчик  
 if (text.length() != 0) {  
 System.*out*.println(" Вы ввели: \n" + text);  
 System.*out*.println(" Задайте количество букв в слове: ");  
 int wordlenght = scan.nextInt();  
 System.*out*.println(" Слова которые находятся в предложениях со знаком '?': ");  
 for (int i = 0; i < text.length(); i++) {  
 //Удаление ненужных символов  
 if(text.charAt(i) == ','||text.charAt(i) == '-'||text.charAt(i) == ':'||text.charAt(i) == ';'||text.charAt(i) == '\''||text.charAt(i) == '"'||text.charAt(i) == '%'||text.charAt(i) == '$'||text.charAt(i) == ')'||text.charAt(i) == '('||text.charAt(i) == '`') {  
 text.deleteCharAt(i);  
 }  
 if ((text.charAt(i) == '?') || (text.charAt(i) == '!') || (text.charAt(i) == '.')) {  
 if ((text.charAt(i) == '?'))  
 counter++; //+1 ячейка для предложения со знаком '?'  
 }  
 }  
 StringBuffer moderntext[] = new StringBuffer[counter]; //столько ячеек, столько и предложений  
 int currPos = 0;//ячейка  
 int lastPos = 0;//начало предложения  
 for (int i = 0; i < text.length(); ++i) {  
 if (text.charAt(i) == '?' || text.charAt(i) == '!' || text.charAt(i) == '.') {  
 if (text.charAt(i) == '?') {  
 moderntext[currPos] = new StringBuffer(text.subSequence(lastPos, i));  
 ++currPos;  
 }  
 lastPos = i + 1;  
 }  
 }  
 /////////////////////////Вывод/////////////////////////////  
 for (int index = 0; index < moderntext.length; index++) {  
 System.*out*.println(moderntext[index]);  
 }  
 System.*out*.println();  
 ////////Подсчёт ячеек для вывода слов в результате////////  
 int prevPos = 0;//предыдущая позиция  
 int res=0;//ячейка результата  
 System.*out*.print("Слова с количеством букв "+wordlenght+" :");  
 int count = 0;//счётчик  
 for (int i = 0; i < counter; i++) {  
 for (int j = 0; j < moderntext[i].length(); j++) {  
 if (moderntext[i].charAt(j) == ' ' || j == moderntext[i].length() - 1) {  
 if (j == moderntext[i].length() - 1) {  
 if (j - prevPos + 1 == wordlenght) {  
 ++count;  
 }  
 } else {  
 if (j-prevPos == wordlenght) {  
 ++count;  
 }  
 prevPos=j+1;  
 }  
 }  
 }  
 }  
 /\*Вывод слов с указаным количеством букв без повторов.  
 В StringBuffer resulttext[], записаны все слова с указаным количеством букв без повторов  
 На местах "null" были слова которые повторялись, это указано для проверки вывода слов без их повтора \*/  
 StringBuffer resulttext[] = new StringBuffer[count];  
 for (int i = 0; i < counter; i++) {  
 System.*out*.println();  
 for (int j = 0; j < moderntext[i].length(); j++) {  
 if (moderntext[i].charAt(j) == ' ' || j == moderntext[i].length() - 1) {  
 if (j == moderntext[i].length() - 1) {  
 if (j - prevPos + 1 == wordlenght) {  
 StringBuffer temp = new StringBuffer(moderntext[i].subSequence(prevPos, j + 1));  
 boolean b = false;  
 for (int k = 0; k < res; k++) {  
 if (resulttext[k].toString().equals(temp.toString())) {  
 b = true;  
 }  
 }  
 if (!b) {  
 resulttext[res++] = temp;  
 }  
 }  
 } else {  
 if (j-prevPos == wordlenght) {  
 StringBuffer temp = new StringBuffer(moderntext[i].subSequence(prevPos, j));  
 boolean b = false;  
 for(int k = 0; k < res; k++) {  
 if (resulttext[k].toString().equals(temp.toString())) {  
 b = true;  
 }  
 }  
 if(!b) {  
 resulttext[res++] = temp;  
 }  
 }  
 prevPos=j+1;  
 }  
 }  
 }  
 }  
 /////////////////////////Вывод/////////////////////////////  
 for (int index = 0; index < resulttext.length; index++) {  
 System.*out*.println(resulttext[index]);  
 }  
 }  
 else{  
 System.*out*.println("Вы ничего не ввели");  
 }  
 }  
}

**Висновок**

У готовій лабораторній роботі програма знаходить питальні речення, оброблює їх таким чином: Виводяться слова без повторень, кількість букв яких, вказується на початку. Програма працює коректно, всі слова виводяться правильно (не повторюються).

**Для перевірки використовувався текст:**

Яблуко апельсин мандарин кокос, банан, банан? вишня! арбуз. кокос, батон, флако? "a `n, ;б: в- с? a a?

**Примітка!!!** Якщо вводити *мульти-фрукт?*, то програма підрахує кількість букв, окрім знаку «-», що є правильним з моєї точки зору на поставлене завдання. Між словами повинні бути пробіли, а також між знаками.