Министерство образования и науки Российской Федерации.

Федеральное государственное бюджетное образовательное учреждение высшего образования.

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ИРКУТСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ.

ОТЧЕТ

к лабораторной работе по дисциплине:

|  |
| --- |
| **Объектно-ориентированное программирование** |
| Разработка консольного приложения |

наименование темы

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Выполнила студентка группы: | ИСТб-16-1 |  |  |  | Винзовский М.Е. |
|  | шифр группы | подпись | Фамилия ИО |
| Проверил: | доцент |  | Аршинский В.Л. |
|  | должность | подпись | Фамилия ИО |

**г. Иркутск 2018 г.**

1. **Постановка задачи**

Разработать консольное приложение на языке java для решения поставленной задачи:

* 1. Найдите символ в строке, который встречается наибольшее число раз.
  2. Найдите самое длинное слово в предложении.
  3. Удалите в предложении каждое третье слово.

1. **Таблица спецификации переменных**

Таблица 1 – спецификации переменных

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Имя переменной | Назначение  переменной | Тип переменной | ОДЗ |
| Функция main(точка входа) | | | | |
| 1 | text | Строка для вывода ответа на 3-е задание | String | - |
| Процедура numberOfCharacters(String text)  Поиск символа во входящей строке, который встречается наибольшее число раз | | | | |
| 2 | text | Входящая строка (Строка 1) | String | - |
| 3 | max | Максимальное кол-во встречаемого символа | int | [0.. 2147483647] |
| 4 | sim | Часто встречаемый символ | char | - |
| 5 | count | Текущее кол-во частоты встречаемого символа | int | [0.. 2147483647] |
| 6 | length | Кол-во символов в строке | int | [0.. 2147483647] |
| 7 | i,j | Вспомогательные переменные для цикла | byte | [0..127] |
| Процедура longWord(String text)  Поиск самого длинного слова во входящей строке | | | | |
| 8 | text | Входящая строка (Строка 2) | String | - |
| 9 | words | Массив строк (Строка 2 разделенная на слова) | String[] | - |
| 10 | str | Текущее слово | String | - |
| 11 | length | Кол-во элементов в массиве words | int | [0.. 2147483647] |
| 12 | i | Вспомогательная переменная для цикла | byte | [0..127] |

Продолжение таблицы 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Функция deleteThreeWords(String text)  Удаление из входящей строки каждого третьего слова | | | | |
| 13 | text | Входящая и выходящая строка (Строка 3) | String | - |
| 14 | words | Массив строк (Строка 3 разделенная на слова) | String[] | - |
| 15 | lenght | Кол-во элементов в массиве words | int | [0.. 2147483647] |
| 16 | i | Вспомогательная переменная для цикла | byte | [0..127] |

1. **Блок-схема алгоритма**

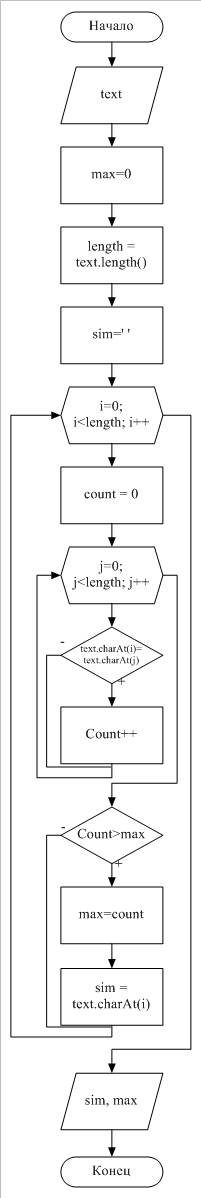
****

Рисунок 1 – алгоритм первого задания

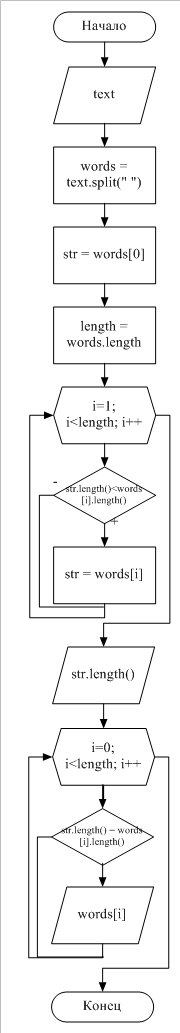


Рисунок 2 – алгоритм второго задания

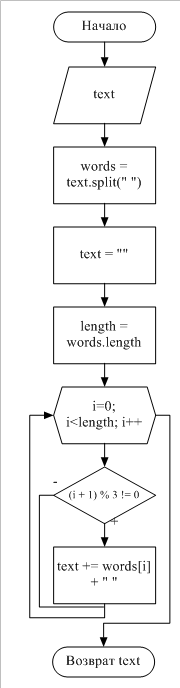


Рисунок 3 – алгоритм третьего задания

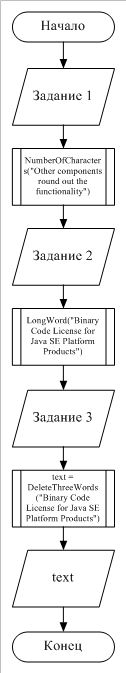


Рисунок 4 – алгоритм функции main

1. **Таблица тестов**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № Теста | Исходные данные | | | Результаты | | |
| Строка 1 | Строка 2 | Строка 3 | 1 | 2 | 3 |
| 1 | Other components round out the functionality | Binary Code License for Java SE Platform Products | Binary Code License for Java SE Platform Products | Символ: t  Количество символов: 6 | Наибольшая длина слова: 8  Слова:  Platform  Products | Удалено каждое 3-е слово:  Binary Code for Java Platform Products |

1. **Результаты тестирования**

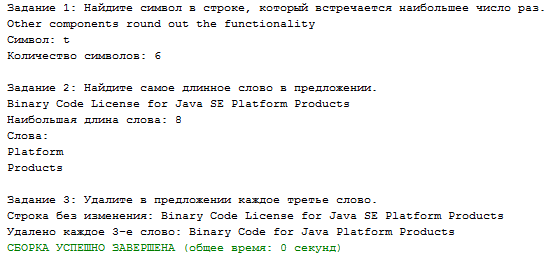


Рисунок 5 – результаты выполнения теста

1. **Исходный код**

public class Lab1 {

public static void numberOfCharacters(String text)//Поиск символа в строке встречаемого наибольшее число раз

{

System.out.println(text);

int max = 0, length = text.length();

char sim = ' ';

for (byte i = 0; i<length; i++){

int count = 0;

for (byte j = 0; j<length; j++)

{

if (text.charAt(i) == text.charAt(j)) count++;

}

if (count > max)

{

max = count;

sim = text.charAt(i); // Выбор символа из строки

}

}

System.out.println("Символ: " + sim);

System.out.println("Количество символов: " + max);

}

public static void longWord(String text)//Поиск самого длинного слова

{

System.out.println(text);

String[] words = text.split(" "); //Разбиение предложения на слова

String str = words[0];

int length = words.length;

for (byte i=1; i<length;i++)

{

if (str.length()<words[i].length())

str = words[i];

}

System.out.println("Наибольшая длина слова: " + str.length());

System.out.println("Словa:");

for (byte i=0; i<length;i++)

{

if (str.length() == words[i].length())

{

System.out.println(words[i]);

}

}

}

public static String deleteThreeWords(String text)//Удаление в предолжении каждоетретье слово

{

System.out.println("Строка без изменения: " + text);

String[] words = text.split(" ");

text = "";

int length = words.length;

for (byte i=0; i<length;i++)

{

if ( (i + 1) % 3 != 0) text += words[i] + " ";

}

return text;

}

public static void main(String [] args)

{

System.out.println("Задание 1: Найдите символ в строке, который встречается наибольшее число раз.");

numberOfCharacters("Other components round out the functionality");

System.out.println("\nЗадание 2: Найдите самое длинное слово в предложении.");

longWord("Binary Code License for Java SE Platform Products");

System.out.println("\nЗадание 3: Удалите в предложении каждое третье слово.");

String text = deleteThreeWords("Binary Code License for Java SE Platform Products");

System.out.println("Удалено каждое 3-е слово: " + text);

}

}

1. **Список использованной литературы**
2. Объектно-ориентированное программирование. Метод.указания по выполнению лабораторных работ / сост.: В.Л. Аршинский. – Иркутск : Изд-во ИРНИТУ, 2015. – 23 c.
3. Программирование на Java для детей, родителей, дедушек и бабушек/ Файн Я.