Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение   
высшего образования

**Иркутский национальный исследовательский технический университет**

|  |
| --- |
| Институт информационных технологий и анализа данных |
| наименование института |
| Кафедра вычислительной техники |
| наименование кафедры |

|  |
| --- |
| **Отчет** |
| по лабораторной работе №1 по дисциплине «Объектно-ориентированное программирование»  «Разработка консольного приложения» |
| наименование темы  Вариант №29 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Выполнил студент |  | АСУб-19-1 |  | C:\Users\Kinzel\AppData\Local\Microsoft\Windows\INetCache\Content.Word\1.png |  | Янхаев Д.В. |
|  |  | шифр |  | подпись |  | И.О. Фамилия |
| Проверил |  |  |  |  |  | Маланова Т.В. |
|  |  |  |  | подпись |  | И.О. Фамилия |
| Работа защищена с оценкой | | | |  | | |

Иркутск 2020 г.

**Содержание**

[Содержание 2](#_Toc50599785)

[1 Постановка задачи 3](#_Toc50599786)

[2 Описание структуры программы 4](#_Toc50599787)

[3 Таблица спецификаций переменных 8](#_Toc50599788)

[4 Таблица тестов 9](#_Toc50599789)

[5 Результаты тестирования 10](#_Toc50599790)

[6 Исходный код 13](#_Toc50599791)

[Список использованных источников 15](#_Toc50599792)

1 Постановка задачи

1. Отсортируйте слова в массиве по третьей букве, если слово меньше трех букв, то по последней.
2. Определите, входит ли строка, указанная в командной строке, при запуске программы, в предложение.
3. Удалите в предложении каждое третье слово.

В первом задании разобьем строку на массив из слов, найдем третий символ сравниваемых элементов массива и применим пузырьковую сортировку.

Во втором задании применим метод, который проверяет, входит ли подстрока в строку.

В третьем задании разобьем строку на массив из слов, и создадим новую строку, пропуская каждый третий элемент массива.

2 Описание структуры программы

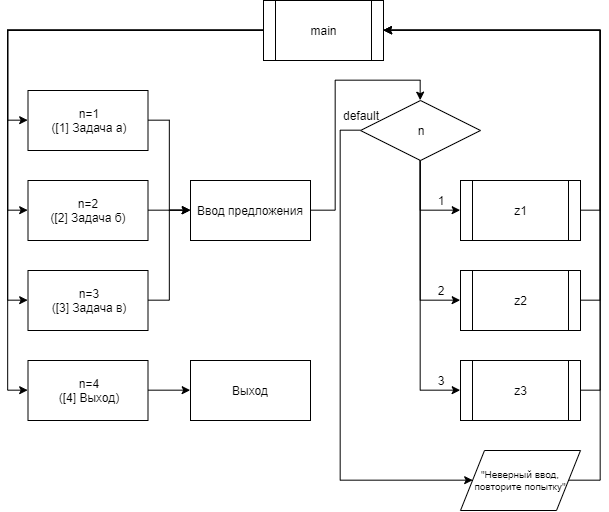


Рисунок 1 – Структура программы

Программа состоит из двух классов: Main и Tasks.

Класс Main реализует меню программы и содержит :

* Главный метод public static void main(String[] args) – выводит главное меню с вводом строки и вызывает методы z1 или z2 или z3 с аргументом mes;

Класс Tasks содержит:

* Метод public void z1(String mes) – выводит решение задачи а)
* Метод public void z2(String mes) – выводит решение задачи б)
* Метод public void z3(String mes) – выводит решение задачи в)

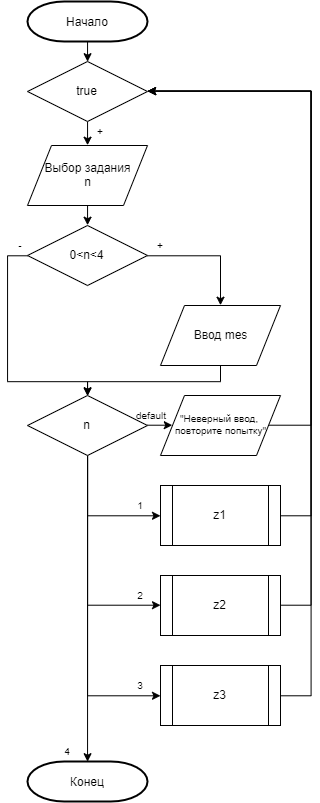


Рисунок 2 – Блок схема класса Main

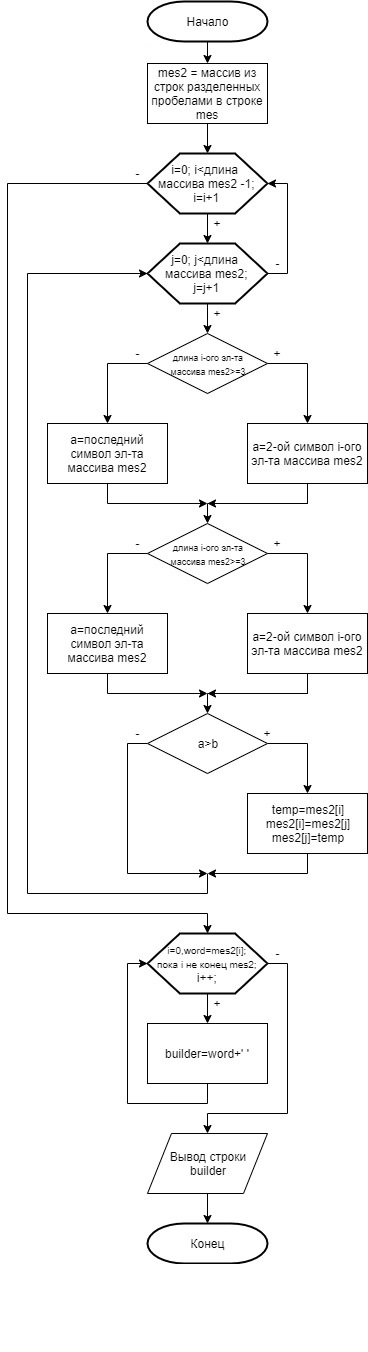


Рисунок 3 – Блок схема задачи а

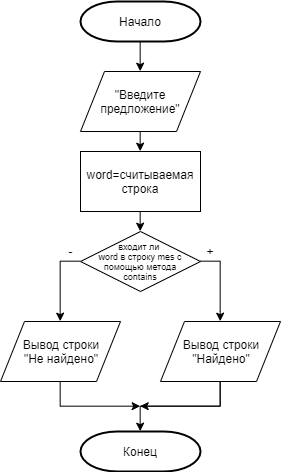


Рисунок 4 – Блок схема задачи б

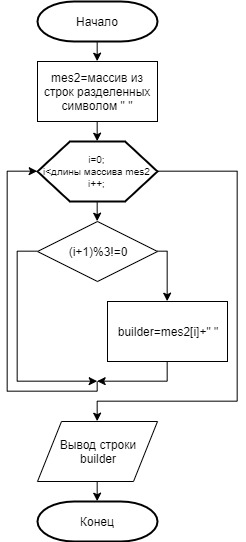


Рисунок 5 – Блок схема задачи в

3 Таблица спецификаций переменных

Таблица 1 – Спецификация переменных

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Назначение | Тип | Диапазон |
| mes | Ввод предложения для задания | String | - |
| n | Выбор пункта меню | Int | -2^32 … 2^32 - 1 |
| mes2 | Массив из строк разделенных пробелом в строке mes | String[] | - |
| a | Хранит третий или последний символ первого из сравниваемых эл-ов массива mes2 | Char | 0 .. 2^16 |
| b | Хранит третий или последний символ второго из сравниваемых эл-ов массива mes2 | Char | 0 .. 2^16 |
| builder | Хранит редактируемую строку | StringBuilder | - |
| word | Хранит слово разделенное пробелом | String | - |
| temp | Дополнительная переменная для перестановки двух элементов массива местами | String | - |

4 Таблица тестов

Таблица 2 - Тесты

|  |  |  |  |
| --- | --- | --- | --- |
| № | Вход | Выход | Примечания |
| 1 | 1  “Прости меня мама” | мама меня Прости | - |
| 2 | 1  “Я хочу кушать” | Я хочу кушать | Значение символов верхнего регистра больше нижнего |
| 3 | 1  “Это не нужно” | не нужно Это | - |
| 4 | 1  “Не виноватая я” | Не виноватая я | - |
| 5 | 1  “Смотри это я!” | я! это Смотри | Значение служ. симв. больше |
| 6 | 2  “Смотри под ноги”  “три ” | Найдено | - |
| 7 | 2  “Hello world”  “Hell” | Найдено | - |
| 8 | 2  “Твои глаза”  “глаза ” | Не найдено | - |
| 9 | 3  “Ого” | Ого | - |
| 10 | 3  “Вижу насквозь” | Вижу насквозь |  |
| 11 | 3  “Мама, а я купил сок в магазине” | Мама, а купил сок магазине | - |
| 12 | 3  “Я смотрю в чужое небо из чужого окна” | Я смотрю чужое небо чужого окна | - |
| 13 | 10 | - | Будет выведено предупреждение и ввод повторится |
| 14 | 4 | - | Выход из программы |

5 Результаты тестирования

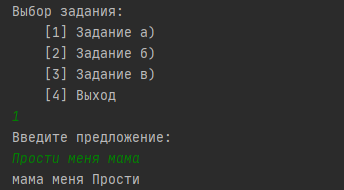


Рисунок 6 – Результат теста 1

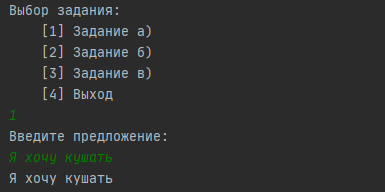


Рисунок 7 – Результат теста 2

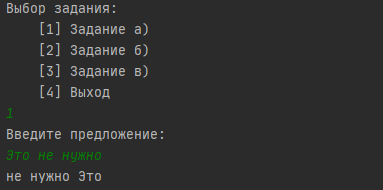


Рисунок 8 – Результат теста 3

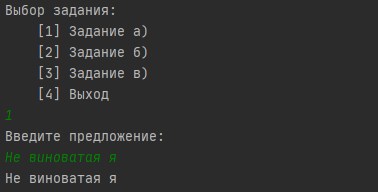


Рисунок 9 – Результат теста 4

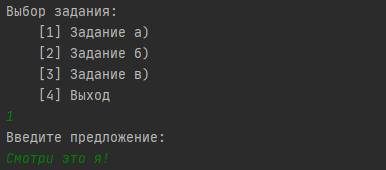


Рисунок 10 – Результат теста 5

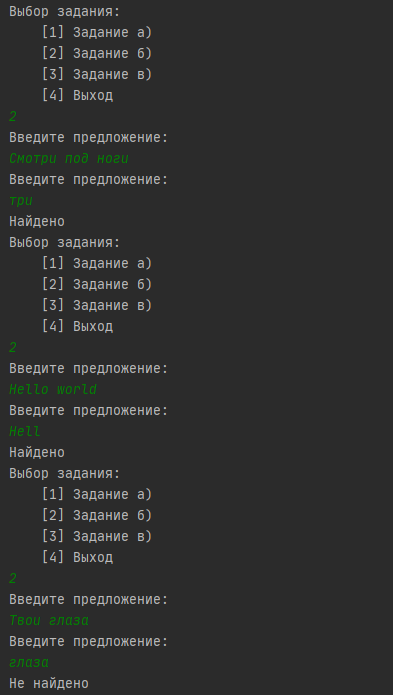


Рисунок 11 – Результат тестов 6,7,8

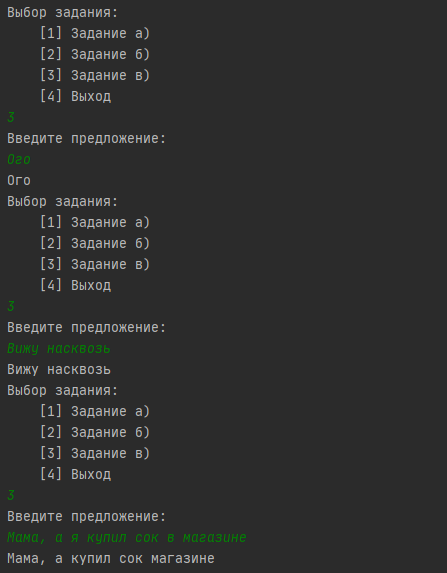


Рисунок 12 – Результат тестов 9,10,11

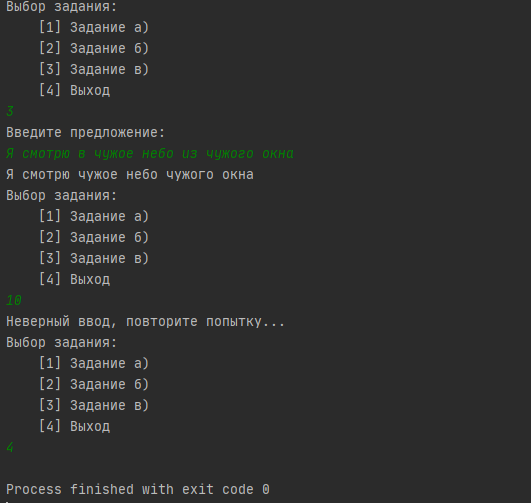


Рисунок 13 – Результат тестов 12,13,14

6 Исходный код

Класс Main

package lab1;  
import java.util.Scanner;  
public class Main {  
  
 public static void main(String[] args) {  
 Tasks t = new Tasks();  
 Scanner in = new Scanner(System.*in*);  
 String mes = null;  
  
 while(true) {  
 System.*out*.println("Выбор задания:\n \t[1] Задание а)\n \t[2] Задание б)\n \t[3] Задание в)\n \t[4] Выход");  
 int n = Integer.*parseInt*(in.nextLine());  
 if(n<4 && n>0) {  
 System.*out*.println("Введите предложение: ");  
 mes = in.nextLine();  
 }  
 switch (n) {  
 case 1:  
 t.z1(mes);  
 break;  
 case 2:  
 t.z2(mes);  
 break;  
 case 3:  
 t.z3(mes);  
 break;  
 case 4:  
 return;  
 default:  
 System.*out*.println("Неверный ввод, повторите попытку...");  
 break;  
 }  
 }  
 }  
}

Класс Tasks

package lab1;  
import java.util.Scanner;  
  
public class Tasks {  
  
 Scanner in = new Scanner(System.*in*);  
 public void z1(String mes){  
  
 String[] mes2;  
 mes2 = mes.split(" ");  
  
 for(int i=0; i<mes2.length-1; i++){  
 for(int j=i+1;j<mes2.length;j++) {  
  
 char a = mes2[i].length() >= 3 ?  
 mes2[i].charAt(2) :  
 mes2[i].charAt(mes2[i].length() - 1);  
  
 char b = mes2[j].length() >= 3 ?  
 mes2[j].charAt(2) :  
 mes2[j].charAt(mes2[j].length() - 1);  
  
 if (a > b){  
 String temp = mes2[i];  
 mes2[i]=mes2[j];  
 mes2[j]=temp;  
 }  
 }  
 }  
 StringBuilder builder = new StringBuilder();  
  
 for (String word : mes2) {  
 builder.append(word + ' ');  
 }  
 System.*out*.println(builder.toString());  
 }  
 public void z2(String mes){  
  
 System.*out*.println("Введите предложение: ");  
 String word = in.nextLine();  
 if (mes.contains(word))  
 System.*out*.println("Найдено");  
 else  
 System.*out*.println("Не найдено");  
 }  
 public void z3(String mes){  
 StringBuilder builder = new StringBuilder();  
 String[] mes2;  
 mes2 = mes.split(" ");  
 for (int i=0; i< mes2.length; i++){  
 if ((i+1)%3!=0){  
 builder.append(mes2[i] + " ");  
 }  
 }  
 System.*out*.println(builder.toString());  
 }  
}

Список использованных источников

1. API Specification for the Java 7 SE. [официальный сайт]

URL: <http://docs.oracle.com/javase/7/docs/api/> (дата обращения:08.09.2020)

1. The Java Tutorials. SE [электронный ресурс],

URL:<http://docs.oracle.com/javase/tutorial/index.html>

(дата обращения:08.09.2020)

1. StringBuffer, StringBuilder в Java [электронный ресурс],

URL: https://javarush.ru/groups/posts/2351-znakomstvo-so-string-stringbuffer-i-stringbuilder-v-java

(дата обращения:08.09.2020)