Министерство науки и высшего образования РФ

ФГАОУ ВО Пермский национальный исследовательский

политехнический университет

Кафедра «Вычислительная математика, механика и биомеханика»

Отчет по лабораторной работе № 8

Вариант 12

тема «Строки»

по дисциплине «Информатика»

Выполнил: студент группу ИСТ-22-1б Ипанов М.В.

Проверил: доцент каф. ВММБ Каменских А.А.

Пермь, 2022

**Содержание**

[Задание 1 3](#_Toc137606987)

[1.1. Постановка задачи 3](#_Toc137606988)

[1.3. Тестирование работы программы 3](#_Toc137606989)

[Задание 2 4](#_Toc137606990)

[2.1. Постановка задачи 4](#_Toc137606991)

[2.3. Тестирование работы программы 4](#_Toc137606992)

[Задание 3 5](#_Toc137606993)

[3.1. Постановка задачи 5](#_Toc137606994)

[3.3. Проверка решения 6](#_Toc137606995)

[Задание 4 7](#_Toc137606996)

[4.1. Постановка задачи 7](#_Toc137606997)

[4.3. Проверка решения 7](#_Toc137606998)

[Задание 5 8](#_Toc137606999)

[5.1. Постановка задачи 8](#_Toc137607000)

[5.3. Проверка решения 8](#_Toc137607001)

[Задание 6 9](#_Toc137607002)

[6.1. Постановка задачи 9](#_Toc137607003)

[6.3. Проверка решения 10](#_Toc137607004)

[Задание 7 11](#_Toc137607005)

[7.1. Постановка задачи 11](#_Toc137607006)

[7.3. Проверка решения 11](#_Toc137607007)

[Задание 8 12](#_Toc137607008)

[8.1. Постановка задачи 12](#_Toc137607009)

[8.3. Проверка решения 12](#_Toc137607010)

[Задание 9 13](#_Toc137607011)

[9.1. Постановка задачи 13](#_Toc137607012)

[9.3. Проверка решения 13](#_Toc137607013)

[Задание 10 14](#_Toc137607014)

[10.1. Постановка задачи 14](#_Toc137607015)

[10.3. Проверка решения 14](#_Toc137607016)

[Задание 11 15](#_Toc137607017)

[11.1. Постановка задачи 15](#_Toc137607018)

[11.3. Проверка решения 15](#_Toc137607019)

[Задание 12 16](#_Toc137607020)

[12.1. Постановка задачи 16](#_Toc137607021)

[12.3. Проверка решения 16](#_Toc137607022)

[Приложение (весь код) 17](#_Toc137607023)

# Задание 1

## 1.1. Постановка задачи

Введите массив символов из N элементов. Переведите все строчные буквы-символы в верхний регистр, перевести все прописные буквы в строчные, заменить все цифры на знак «!».

## 1.3. Тестирование работы программы

Таблица 1

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Проверка |
| 1 |  | верно |
| 2 |  | верно |

Программа работала корректно со всеми введёнными данными.

# Задание 2

## 2.1. Постановка задачи

Дан массив символов, содержащий число в восьмеричной системе счисления. Проверить правильность ввода этого числа (в его записи должны быть только символы 0, 1, 2, … , 7). Если число введено неверно, сообщить об этом пользователю и повторить ввод, ввод должен повторяться до тех пор, пока не будет введено число. При правильном вводе перевести число в двоичную систему счисления, записать число в новый массив символов.

## 2.3. Тестирование работы программы

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Проверка |
| 1 |  | верно |
| 2 |  | верно |

При выполнении была использована часть кода из преведущей лаб. работы по конвертации чисел из разных систем исчисления.

# Задание 3

## 3.1. Постановка задачи

Дан массив символов. Проанализировать массив и ответить на вопросы:

а) Определить количество строчных латинских букв.

б) Выяснить, входит ли в последовательность хотя бы 1-н символ слова, введенного с клавиатуры, если входит, то посчитать сколько раз.

в) Выяснить, верно ли, что среди символов имеются точки, запятые и многоточия.

г) Выяснить, имеется ли среди символов пара соседствующих скобок «(, ), {, }, [, ]» если скобки присутствуют в последовательности, то выяснить есть ли закрытые пары разного сочетания и вывести ответ.

д) Выяснить, имеется ли среди символов пара соседствующих одинаковых символов, относящихся к знакам препинания.

## 3.3. Проверка решения

Таблица 1

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| .0№ п.п. | Решение Java | Проверка |
| 1 |  | верно |
| 2 |  | верно |

# Задание 4

## 4.1. Постановка задачи

Выяснить, верно ли. что существуют такие натуральные i и j, что i<k <j < п и что si, si+1, sk, sj, sj+1 цифры, si, si+1 больше sk, a sj, sj+1 меньше, чем sk.Задана матрица символов N×N. Определить построчно и по столбцам, есть ли в них выражения вида cos«число» - sin«число» = «число» или последовательность чисел. Вывести выражения данного вида по строкам и по столбцам. Проверить есть ли правильно вычисленные выражения, и подсчитать их количество. Создать ArrayList и записать в него выражения, которые были вычислены не правильно с корректировкой ответа. Вывести информацию на экран.

## 4.3. Проверка решения

Таблица 1

Тестирование работы программы и проверка результатов решения

|  |  |
| --- | --- |
| № п.п. | Решение Java |
| 1 |  |

«Вот и много букв, но нету паники, Ведь надо читать задание внимательно»

Форматирование не очень красивое через Arrays.ToString но… вполне читаемо.

# Задание 5

## 5.1. Постановка задачи

Найти слова в тексте, которые встречаются два и более раз. Вывести их на экран с указанием сколько раз встречается слово. При сравнении слов регистр не учитывать.

## 5.3. Проверка решения

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Проверка |
| 1 |  | Почти верно |
| 2 |  | верно |

При тестировании программы выяснилось, что точку она тоже считает за часть слова. Но проблема решается коррекцией входных данных (не правильно, конечно, но эффективно)

# Задание 6

## 6.1. Постановка задачи

Дан массив строк, который содержит информацию о семьях учеников одного класса, не менее 20 строк. Создать две программы для регламентированного и нерегламентированного ввода данных по аудиториям. Определить в каждой программе количество полных семей по районам с доходом на одного члена семьи превышающим прожиточный минимум в два и более раз, и отдельно количество полных семей, в которых есть хотя бы одно домашнее животное.

- Первая программа: Сделать регламентированный ввод данных по семьям. Каждая ячейка массива содержит информацию в определенном виде: Фамилия; Район, где живут; Полная/Неполная семья; Количество членов семьи; Количество детей; Количество комнат в жилье; Квадратные метры жилья; Доход на одного члена семьи; Наличие компьютеров; Количество компьютеров; Наличие домашних животных; Количество домашних животных.

- Вторая программа: Сделать нерегламентированный ввод данных по семьям. Т.е. пользователи вводят информацию в произвольном виде.

## 6.3. Проверка решения

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Проверка |
| 1 |  | верно |
| 2 |  | верно |
|  |  | верно |

Программа работает так, как задумано. (При проверке мы убедились, что программа составила таблицу данных.)

# Задание 7

## 7.1. Постановка задачи

В заданной строке расположить в обратном порядке слова, которые включают русские буквы. Разделителями слов считаются пробелы.

## 7.3. Проверка решения

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Проверка |
| 1 |  | верно |
| 2 |  | верно |

Программа работает корректно со всеми введёнными данными.

# Задание 8

## 8.1. Постановка задачи

8. Осуществить форматирование текста, где имена переменных записаны с прописных букв в формате MyText, такие слова нужно переделать к формату my\_text.

## 8.3. Проверка решения

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Проверка |
| 1 |  | верно |
| 2 |  | Всё правильно. Программа расчитана только на Латиницу |
| 3 |  | верно |

В последнем примере ввёл максимально плохие данные, но программа всё равно работала корректно.

# Задание 9

## 9.1. Постановка задачи

Дан текст с комментариями. Из текста удалить все комментарии (/…/, \....\,|...|), если комментарий нет, то сообщить об этом.

## 9.3. Проверка решения

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Проверка |
| 1 |  | верно |
| 2 |  | верно |
|  |  | верно |

Из за специфики работы Java мне не удалось запрограммировать поиск комментариев с обратной косой чертой. Это какой-то специальный символ, и он всё ломает.

# Задание 10

## 10.1. Постановка задачи

Дан текст, который содержит данные определенной структуры, разделителем данных об одном объекте является «;», разделитель между объектами «.». «Школьник»: фамилия; имя; отчество; пол; национальность; рост; вес; дата рождения (год, месяц число); номер телефона; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); школа; класс. Разбить данные по объектам на массив строк. Посчитать количество людей, у которых в номере телефона есть сочетание цифр 912 или 919 и последняя цифра номера 5. Вывести сведения про всех учеников пятых классов.

## 10.3. Проверка решения

Тестирование работы программы и проверка результатов решения

|  |  |
| --- | --- |
| № п.п. | Решение Java |
| 1 |  |
|  | Верно, потому что нарушен ввод данных – нет фамилии ученика |
| 2 |  |
|  | верно |

# Задание 11

## 11.1. Постановка задачи

Создать программу форматирования массива строк используя три разных регулярных выражения, т.е. регулярные выражения разного типа, созданные при использовании разных метасимволов и их сочетаний.

## 11.3. Проверка решения

Тестирование работы программы и проверка результатов решения

|  |  |
| --- | --- |
| № п.п. | Решение Java |
| 1 |  |
|  | Верно, это не один из форматов |
| 2 |  |
|  | верно |
|  |  |
|  | верно |

При тестировании программы на некоторых реальных и произвольных данных при правильном вводе данных ошибок выявлено не было. Сравнение проверки и ответа, который выдал код показывают правильность написанного кода.

# Задание 12

## 12.1. Постановка задачи

Дан массив строк. Создать массив регулярных выражений [ИСТ, бакалавр, отличник, пять и более пробелов, !, 2019] и массив замен [студент, ФПММ, …, выпускник, магистр, ???, 2024]. В цикле с использование регулярных выражений выполнить автозамену информации. Исходную и измененную информацию вывести построчно: сначала исходная строка, затем измененная.

## 12.3. Проверка решения

Тестирование работы программы и проверка результатов решения

|  |  |  |
| --- | --- | --- |
| № п.п. | Решение Java | Проверка |
| 1 |  |  |
| 2 |  |  |

Не понял формулировку задания. Слова в квадратных скобках не поддаются классификации и рационализации, также ввело в заблуждение понятие «Массив замен». В связи с чем задание было пропущено.

# Приложение (весь код)

import java.lang.String;  
import java.lang.reflect.Array;  
import java.nio.Buffer;  
import java.util.\*;  
import java.util.regex.Matcher;  
import java.util.regex.Pattern;  
  
import static java.lang.Math.\*;  
  
class func {  
  
 public static int[][] mass\_in(int i) { // инициализация массива  
 // 1 - прямоуг. матрица  
 // 2 - квадр. матрица  
 Scanner in = new Scanner(System.*in*);  
  
 if (i == 1) {  
 System.*out*.println("Введите размерность матрицы:");  
 System.*out*.print("m (строк) = ");  
 int m = in.nextInt();  
 System.*out*.print("n (столбцов) = ");  
 int n = in.nextInt();  
 int[][] mass = new int[m][n];  
 return mass;  
 } else {  
 System.*out*.println("Введите размерность квадратной матрицы:");  
 System.*out*.print("n = ");  
 int n = in.nextInt();  
 int[][] mass = new int[n][n];  
 return mass;  
 }  
 }  
  
 public static int[][] mass\_enter(int[][] mass) { // ручной ввод массива  
 // ввод - пустой массив нужной размерности  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.printf("(ручной ввод массива %d\*%d)\n", mass.length, mass[0].length);  
 for (int i = 0; i < mass.length; i++) {  
 System.*out*.printf("%d строка: \n", i + 1);  
 for (int j = 0; j < mass[0].length; j++) {  
 System.*out*.printf("x\_%d = ", j + 1);  
 mass[i][j] = in.nextInt();  
 }  
 }  
 return mass;  
 }  
  
 public static void mass\_out(int[][] mass) { // форматированный вывод массива (sp - промежуток между числами  
  
 int[][] mass\_spase = new int[mass.length][mass[0].length];  
 int k\_last = 0, k\_first = 1, k;  
 int x;  
 // определение позиций элемента в списке (для форматирования)  
 for (int m = 0; m < mass.length; m++) {  
 for (int n = 0; n < mass[0].length; n++) {  
 x = *abs*((int) (mass[m][n] % 1 \* 1000));  
 k = 3;  
 for (int i = 2; i >= 0; i--) {  
 if (x / *stepen*(10, i) > 0) {  
 k = i;  
 }  
 }  
 mass\_spase[m][n] = 3 - k;  
 k\_last = *max*(3 - k, k\_last);  
  
 x = *abs*((int) mass[m][n]);  
 k = 0;  
 if (x == 0) {  
 k++;  
 }  
 while (x > 0) {  
 k++;  
 x /= 10;  
 }  
 mass\_spase[m][n] += k \* 10;  
 k\_first = *max*(k, k\_first);  
 }  
 }  
 // форматированный вывод результатов  
 for (int m = 0; m < mass.length; m++) {  
 for (int n = 0; n < mass[0].length; n++) {  
 if (mass[m][n] >= 0) {  
 System.*out*.print(" ");  
 }  
 for (int i = 0; i < k\_first - mass\_spase[m][n] / 10; i++) {  
 System.*out*.print(" ");  
 }  
 if (mass\_spase[m][n] % 10 == 0) {  
 System.*out*.print((int) mass[m][n]);  
 System.*out*.print(" ");  
 } else {  
 System.*out*.printf("%." + mass\_spase[m][n] % 10 + "f", mass[m][n]);  
 }  
 for (int i = 0; i < k\_last - mass\_spase[m][n] % 10 + 1; i++) {  
 System.*out*.print(" ");  
 }  
 }  
 System.*out*.println();  
 }  
 }  
  
 public static int stepen(int x, int n) {  
 int sum = 1;  
 for (int i = 0; i < n; i++) {  
 sum = sum \* x;  
 }  
 return sum;  
 }  
 public static boolean num2\_1(int x, int n) {  
 // является ли число n-ричным?  
  
 for (int i = 1; i <= 10; i++) {  
 if (x % 10 >= n) {  
 return false;  
 }  
 x /= 10;  
 }  
 return true;  
 }  
  
 public static int[] num2\_2(int x) {  
 // перевод int в массив  
  
 int x\_i = 1, S = 1;  
 while (true) {  
 for (int i = 0; i < x\_i - 1; i++) {  
 S = S \* 10;  
 }  
 if (S >= x) {  
 break;  
 } else {  
 x\_i += 1;  
 }  
 }  
  
 int[] mass = new int[x\_i];  
 for (int i = 0; i < x\_i; i++) {  
 mass[x\_i - i - 1] = x % 10;  
 x /= 10;  
 }  
 return mass;  
 }  
  
 public static int[] num2\_3(int[] mass, int N\_1, int N\_2) {  
 // перевод массива из любой СС (массив) в любую СС (массив)  
  
 int S = 0;  
 for (int i = 0; i < mass.length; i++) {  
 S += mass[mass.length - 1 - i] \* *stepen*(N\_1, i);  
 }  
  
 int x\_i = 1, x = 1;  
 while (true) {  
 x \*= N\_2;  
 if (x >= S) {  
 break;  
 } else {  
 x\_i += 1;  
 }  
 }  
  
 int[] mass\_new = new int[x\_i];  
 for (int i = x\_i - 1; i >= 0; i--) {  
 mass\_new[i] = S % N\_2;  
 S /= N\_2;  
 }  
  
 return mass\_new;  
 }  
}  
  
class Work\_is\_start { // вариант 12  
  
 public static void main(String[] args) {  
 Scanner in = new Scanner(System.*in*);  
  
 String S, S1;  
 char[] array\_s;  
 int x\_i, i\_i, n\_i, y\_i, t\_i, k\_i;  
 double x\_d, y\_d, r\_d;  
 char l, s\_copy, ch;  
 String s, str;  
 boolean flag = false;  
  
  
 int num;  
 while (true) {  
 System.*out*.println("\_\_\_Введите номер задачи от 1 до 12\_\_\_");  
 num = in.nextInt();  
  
 flag = false;  
 switch (num) {  
 case 1:  
 System.*out*.println("Введите массив симоволов: ");  
 in.nextLine();  
 S = in.nextLine();  
 array\_s = S.toCharArray();  
  
 for (int i = 0; i < array\_s.length; i++) {  
 if (Character.*isLowerCase*(array\_s[i])) {  
 array\_s[i] = Character.*toUpperCase*(array\_s[i]);  
 } else if (Character.*isUpperCase*(array\_s[i])) {  
 array\_s[i] = Character.*toLowerCase*(array\_s[i]);  
 } else if (Character.*isDigit*(array\_s[i])) {  
 array\_s[i] = '!';  
 }  
 System.*out*.print(array\_s[i]);  
 }  
 break;  
 case 2:  
 in.nextLine();  
 while (true) {  
 System.*out*.println("Введите массив: ");  
 s = in.nextLine();  
 String s\_new = "";  
 array\_s = s.toCharArray();  
 for (int i = 0; i < array\_s.length; i++) {  
 if (Character.*isDigit*(array\_s[i])) {  
 s\_new += array\_s[i];  
 flag = true;  
 } else if (flag) {  
 flag = false;  
 break;  
 }  
 }  
 x\_i = Integer.*parseInt*(s\_new);  
 if (flag && func.*num2\_1*(x\_i, 8)) {  
 break;  
 } else {  
 System.*out*.println("Число введено неверно.");  
 }  
 }  
 System.*out*.println(Arrays.*toString*(func.*num2\_3*(func.*num2\_2*(x\_i), 10, 2)));  
 break;  
  
 case 3:  
 System.*out*.println("Сколько элементов в массиве?");  
 n\_i = in.nextInt();  
 char[] mass = new char[n\_i];  
 System.*out*.println("Введите элементы массива: ");  
 in.nextLine();  
 for (int i = 0; i < n\_i; i++) {  
 System.*out*.print(i+1 + ") ");  
 mass[i] = in.nextLine().charAt(0);  
 }  
 System.*out*.println("Ответ: ");  
 x\_i = 1;  
 for (int i = 0; i < n\_i; i++) {  
 if ((int) mass[i] >= 97 && (int) mass[i] >= 122) {  
 x\_i += 1;  
 }  
 }  
 System.*out*.println("a) " + x\_i);  
  
 System.*out*.print("Введите слово для пункта (б): ");  
 s = in.nextLine();  
 k\_i = 0;  
 ch = ' ';  
 for (int i = 0; i < s.length(); i++) {  
 for (int j = 0; j < mass.length; j++) {  
 if (s.charAt(i) == mass[j]) {  
 ch = s.charAt(i);  
 k\_i += 1;  
 }  
 }  
 if (ch != ' ') {  
 break;  
 }  
 }  
 if (ch != ' ') {  
 System.*out*.println("б) символ (" + ch + ") встречался " + k\_i + " раз");  
 } else {  
 System.*out*.println("б) Не встречано данных символов");  
 }  
  
 flag = false;  
 for (int j = 0; j < mass.length; j++) {  
 if (mass[j] == 44 || mass[j] == 46) {  
 flag = true;  
 }  
 }  
 if (flag) {  
 System.*out*.println("в) верно");  
 } else {  
 System.*out*.println("в) Не верно");  
 }  
  
 flag = false;  
 for (int j = 0; j < mass.length - 1; j++) {  
 if ((mass[j] == '(' || mass[j] == '{' || mass[j] == '[')  
 && (mass[j+1] == ')' || mass[j+1] == '}' || mass[j+1] == ']')) {  
 flag = true;  
 }  
 }  
 if (flag) {  
 System.*out*.println("г) такие пары имеются");  
 } else {  
 System.*out*.println("г) такие пары НЕ имеются");  
 }  
  
 str = ".?!:;,-()";  
 flag = false;  
 for (int j = 0; j < mass.length - 1; j++) {  
 for (int i = 0; i < str.length(); i++) {  
 if (mass[j] == str.charAt(i) && mass[j+1] == str.charAt(i)) {  
 flag = true;  
 }  
 }  
 }  
 if (flag) {  
 System.*out*.println("д) такие пары имеются");  
 } else {  
 System.*out*.println("д) такие пары НЕ имеются");  
 }  
 break;  
 case 4:  
 ArrayList<String> list = new ArrayList<>();  
 list.add("cos(");  
 list.add("x");  
 list.add(") - sin(");  
 list.add("y");  
 list.add(") = ");  
 list.add("z");  
 System.*out*.println(list.toString());  
 System.*out*.println("Скорроктируйте параметры x, y, z:");  
 in.nextLine();  
 System.*out*.print("x = ");  
 list.set(1, in.nextLine());  
 System.*out*.print("y = ");  
 list.set(3, in.nextLine());  
 System.*out*.print("z = ");  
 list.set(5, in.nextLine());  
 System.*out*.println(list.toString());  
 break;  
 case 5:  
 System.*out*.println("Введите текст: ");  
 in.nextLine();  
 s = in.nextLine();  
  
 str = ".?!:;,-()";  
 for (int i = 0; i < str.length(); i++) {  
 s.replaceAll("str.charAt(i)", " ");  
 }  
  
 List<String> list\_1 = Arrays.*asList*(s.split(" "));  
 for (int i = 0; i < list\_1.size(); i++) {  
 list\_1.set(i, list\_1.get(i).toLowerCase());  
 }  
  
 Set<String> uniqueWords = new HashSet<String>(list\_1);  
 for (String word : uniqueWords) {  
 if (Collections.*frequency*(list\_1, word) >= 2 && word.equals("") == false) {  
 System.*out*.println(word + ": " + Collections.*frequency*(list\_1, word));  
 }  
 }  
  
 break;  
 case 7:  
 System.*out*.println("Введите текст: ");  
 in.nextLine();  
 s = in.nextLine();  
 List<String> list\_2 = Arrays.*asList*(s.split(" "));  
 List<String> list\_2\_1 = new ArrayList<>();  
 for (int i = 0; i < list\_2.size(); i++) {  
 for (int j = 0; j < list\_2.get(i).length(); j++) {  
 if ( (list\_2.get(i).charAt(j) >= 'А' && list\_2.get(i).charAt(j) <= 'Я') ||  
 (list\_2.get(i).charAt(j) >= 'а' && list\_2.get(i).charAt(j) <= 'я') ) {  
 list\_2\_1.add(list\_2.get(i));  
 break;  
 }  
 }  
 }  
 Collections.*reverse*(list\_2\_1);  
 System.*out*.println(list\_2\_1);  
 break;  
 case 8:  
 System.*out*.println("Введите текст: ");  
 in.nextLine();  
 s = in.nextLine();  
 List<String> list\_3 = Arrays.*asList*(s.split(" "));  
 for (int i = 0; i < list\_3.size(); i++) {  
 flag = false;  
 for (int j = 1; j < list\_3.get(i).length(); j++) {  
 if (list\_3.get(i).charAt(j) >= 'A' && list\_3.get(i).charAt(j) <= 'Z' && list\_3.get(i).charAt(j - 1) != ' ' && list\_3.get(i).charAt(j - 1) != '\_') {  
 list\_3.set(i, list\_3.get(i).substring(0, j) + "\_" + list\_3.get(i).substring(j));  
 flag = true;  
 }  
 }  
 if (flag) {  
 list\_3.set(i, list\_3.get(i).toLowerCase());  
 }  
 }  
 s = "";  
 for (int i = 0; i < list\_3.size(); i++) {  
 s += list\_3.get(i) + " ";  
 }  
 System.*out*.println(s);  
 break;  
 case 9:  
 System.*out*.println("Введите текст: ");  
 in.nextLine();  
 s = in.nextLine();  
 flag = false;  
 Boolean flag\_m = false;  
 x\_i = 0;  
 for (int i = 0; i < s.length(); i++) {  
 if (s.charAt(i) == '/' && flag == false) {  
 flag = true;  
 x\_i = i;  
 } else if (s.charAt(i) == '/' && flag == true) {  
 s = s.substring(0, x\_i) + s.substring(i + 1, s.length());  
 flag = false;  
 i = x\_i - 1;  
 flag\_m = true;  
 }  
 }  
 for (int i = 0; i < s.length(); i++) {  
 if (s.charAt(i) == '|' && flag == false) {  
 flag = true;  
 x\_i = i;  
 } else if (s.charAt(i) == '|' && flag == true) {  
 s = s.substring(0, x\_i) + s.substring(i + 1, s.length());  
 flag = false;  
 i = x\_i - 1;  
 flag\_m = true;  
 System.*out*.println("\*");  
 }  
 }  
  
 if (flag\_m) {  
 System.*out*.println(s);  
 } else {  
 System.*out*.println("Текст без комментариев!");  
 }  
 break;  
 case 10:  
 System.*out*.println("Введите текст: ");  
 in.nextLine();  
 s = in.nextLine();  
 s = s.replaceAll(":", ";");  
 x\_i = 0;  
 for (int i = 0; i < s.length(); i++) {  
 if (s.charAt(i) == '.') {  
 x\_i += 1;  
 }  
 }  
 String[] s\_1 = s.split("\\.");  
 String[][] s\_2 = new String[x\_i][11];  
 for (int i = 0; i < x\_i; i++) {  
 s\_2[i] = s\_1[i].split(";");  
 }  
  
 System.*out*.println("Люди c номерами ...912... или ...919... и ...5:");  
 x\_i = 0;  
 for (int i = 0; i < s\_2.length; i++) {  
 s\_2[i][8] = s\_2[i][8].replaceAll(" ", "");  
 if ((s\_2[i][8].indexOf("912") > -1 || s\_2[i][8].indexOf("919") > -1)  
 && s\_2[i][8].charAt(s\_2[i][8].length() - 1) == '5') {  
 x\_i++;  
 }  
 }  
 System.*out*.println(x\_i);  
  
 System.*out*.println("Ученики 5-х классов: ");  
 for (int i = 0; i < s\_2.length; i++) {  
 if (s\_2[i][11].indexOf("5") > 0) {  
 System.*out*.println(Arrays.*toString*(s\_2[i]));  
 }  
 }  
 break;  
 case 11:  
 System.*out*.println("Введите email или десятичное число или абривеатуру факультета: ");  
 in.nextLine();  
 s = in.nextLine();  
  
 Pattern pt = Pattern.*compile*("\\w+@\\w+\\.(ru||com)");  
 Matcher m\_1 = pt.matcher(s);  
 if (m\_1.matches()) {  
 System.*out*.println("Вы ввели имейл");  
 }  
  
 pt = Pattern.*compile*("(\\d\\.\\d||\\d,\\d)");  
 m\_1 = pt.matcher(s);  
 if (m\_1.matches()) {  
 System.*out*.println("Вы ввели десятичное число");  
 }  
  
 pt = Pattern.*compile*("[А-Я]{2,}");  
 m\_1 = pt.matcher(s);  
 if (m\_1.matches()) {  
 System.*out*.println("Вы ввели абривеатуру факультета");  
 }  
 break;  
 case 12:  
 System.*out*.println("Не понял формулироваку задания");  
 break;  
 default:  
 System.*out*.println("Вы ввели неверный номер задания");  
 }  
 System.*out*.println();  
 }  
  
  
 }  
}