Федеральное государственное бюджетное образовательное учреждение высшего образования

«Санкт-Петербургский государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича»

(СПбГУТ)

Лабораторная работа 6

Разработка Java-приложений управления телекоммуникациями

«Файлы»

Работу выполнил:

Студент 3-го курса

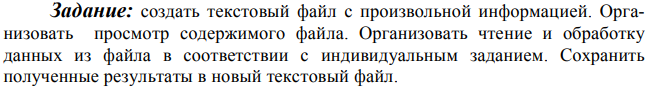
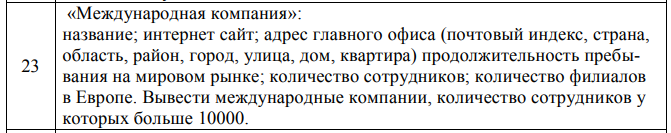
Группы ИКПИ-74

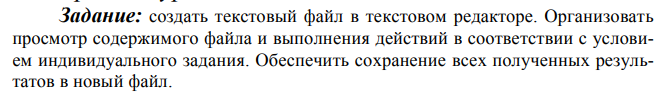
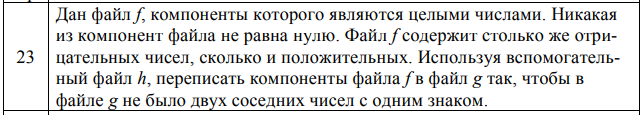
Чугуненко Александр Сергеевич

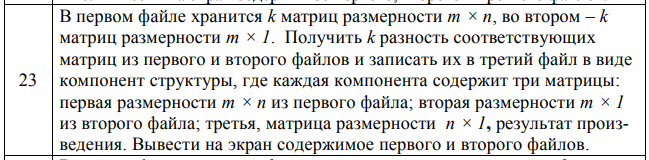
2020

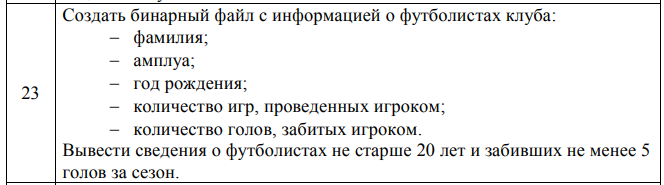
Санкт-Петербург

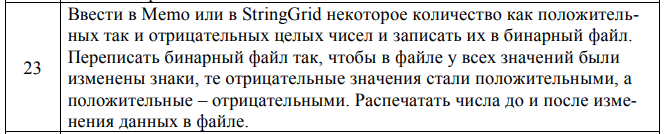
**Задание**

**1)** ****

**2)** ****

**3)** ****

**4)** 

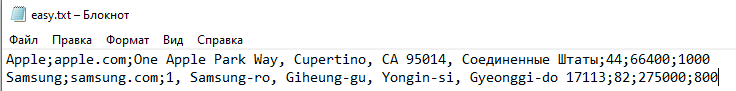
**5)** ****

**6) **

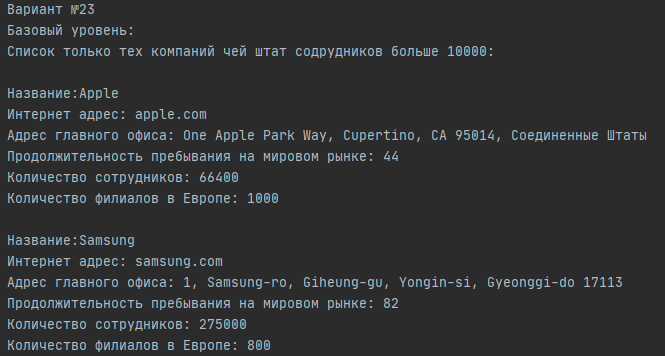
**Вывод результата в программе:**

**Задание 1:**

Файл easy.txt

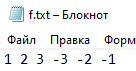
****

Вывод в программе

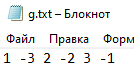


**Задание 2:**

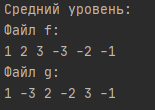
Файл f.txt



Файл g.txt

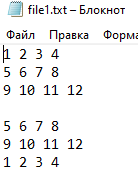


Вывод в программе

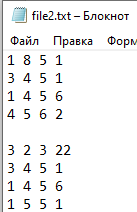


**Задание 3:**

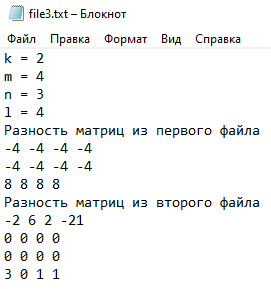
Файл file1.txt



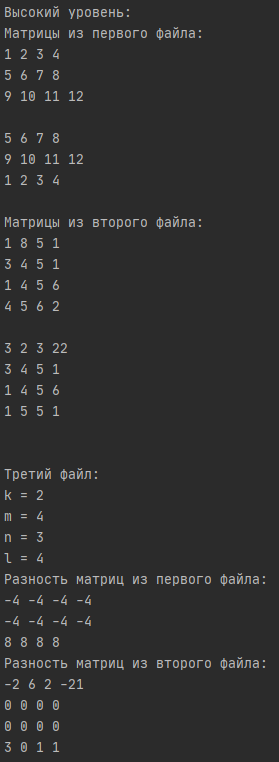
Файл file2.txt



Файл file3.txt

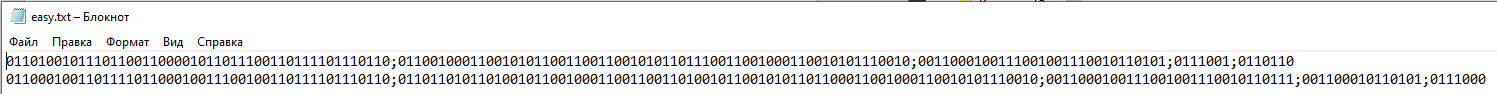


Вывод в программе

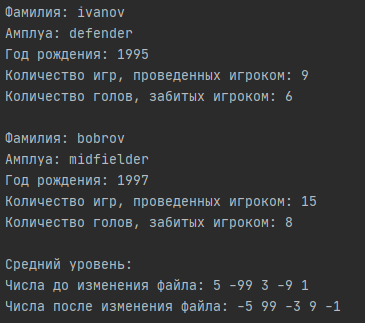


**Задание 4:**

Файл easy.txt

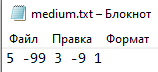


Вывод в программе

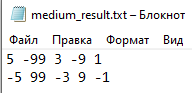


**Задание 5:**

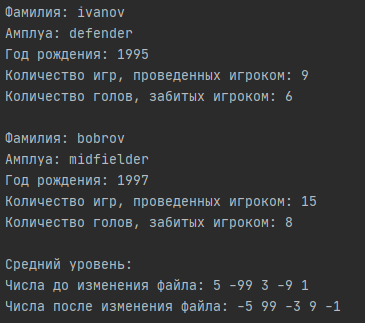
Файл medium.txt



Файл medium\_result.txt

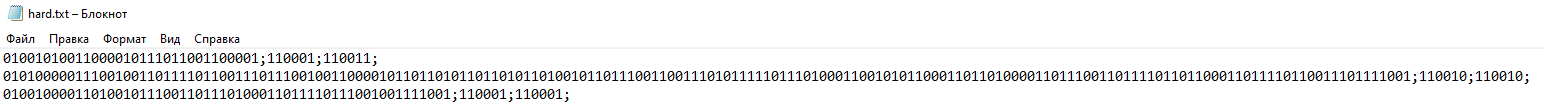


Вывод в программе

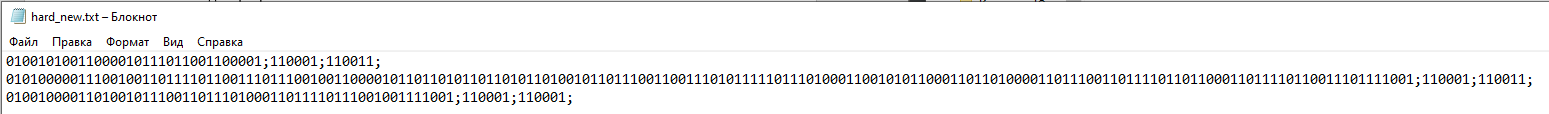


**Задание 6:**

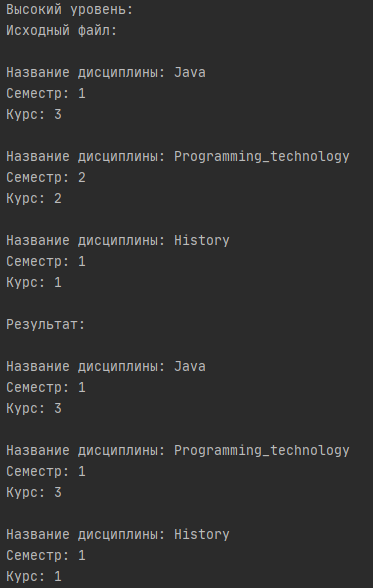
Файл hard.txt



Файл hard\_new.txt



Вывод в программе



**Код программы:**

8.1.

package com.company;  
  
import javax.swing.\*;  
import java.io.\*;  
import java.util.ArrayList;  
import java.util.Scanner;  
  
public class Main {  
  
 public static class Company {  
 private String name;  
 private String site;  
 private String adress;  
 private int year;  
 private int sum\_s;  
 private int sum\_f;  
  
 public String getName() {  
 return name;  
 }  
 public void setName(String name) {  
 this.name = name;  
 }  
 public String getSite() {  
 return site;  
 }  
 public void setSite(String site) {  
 this.site = site;  
 }  
 public String getAdress() {  
 return adress;  
 }  
 public void setAdress(String adress) {  
 this.adress = adress;  
 }  
 public int getYear() {  
 return year;  
 }  
 public void setYear(int year) {  
 this.year = year;  
 }  
 public int getSum\_s() {  
 return sum\_s;  
 }  
 public void setSum\_s(int sum\_s) {  
 this.sum\_s = sum\_s;  
 }  
 public int getSum\_f() {  
 return sum\_f;  
 }  
 public void setSum\_f(int sum\_f) {  
 this.sum\_f = sum\_f;  
 }  
  
 public String toString(){  
 return "\nНазвание:" + getName() + "\nИнтернет адрес: " + getSite() +  
 "\nАдрес главного офиса: " + getAdress() + "\nПродолжительность пребывания на мировом рынке: " +  
 getYear() + "\nКоличество сотрудников: " + getSum\_s() + "\nКоличество филиалов в Европе: " + getSum\_f();  
 }  
 }  
  
 public static void easy() throws IOException {  
 System.*out*.println("Базовый уровень:");  
 BufferedReader reader = new BufferedReader(new FileReader("easy.txt"));  
  
 String line = null;  
 Scanner scanner = null;  
 int index = 0;  
 int j = 0;  
  
 Company[] comp = new Company[3];  
  
 while ((line = reader.readLine()) != null) {  
 comp[j] = new Company();  
 scanner = new Scanner(line);  
 scanner.useDelimiter(";");  
 while (scanner.hasNext()) {  
 String data = scanner.next();  
 if (index == 0)  
 comp[j].setName(data);  
 else if (index == 1)  
 comp[j].setSite(data);  
 else if (index == 2)  
 comp[j].setAdress(data);  
 else if (index == 3)  
 comp[j].setYear(Integer.*parseInt*(data));  
 else if (index == 4)  
 comp[j].setSum\_s(Integer.*parseInt*(data));  
 else if (index == 5)  
 comp[j].setSum\_f(Integer.*parseInt*(data));  
 else  
 System.*out*.println("\nНекорректные данные " + data);  
 index++;  
 }  
 j++;  
 index = 0;  
 }  
  
 System.*out*.println("Список только тех компаний чей штат содрудников больше 10000:");  
 for (int i = 0; i<j; i++) {  
 if (comp[i].getSum\_s() > 10000)  
 System.*out*.println(comp[i].toString());  
 }  
  
 }  
  
  
  
 public static void medium()throws IOException {  
 System.*out*.println("\n\nСредний уровень:");  
 BufferedReader reader = new BufferedReader(new FileReader("f.txt"));  
  
 String line = null;  
 Scanner scanner = null;  
 ArrayList<Integer> mas = new ArrayList<>();  
 ArrayList<Integer> plus = new ArrayList<>();  
 ArrayList<Integer> minus = new ArrayList<>();  
 while ((line = reader.readLine()) != null) {  
 scanner = new Scanner(line);  
 scanner.useDelimiter(" ");  
 while (scanner.hasNext()) {  
 String data = scanner.next();  
 mas.add(Integer.*parseInt*(data));  
 }  
 }  
  
 System.*out*.println("Файл f:");  
 for (int i = 0; i < mas.size(); i++){  
 System.*out*.print(mas.get(i) + " ");  
 if (mas.get(i) > 0) plus.add(mas.get(i));  
 if (mas.get(i) < 0) minus.add(mas.get(i));  
 }  
  
  
 try(FileWriter g = new FileWriter("g.txt", false)) {  
 System.*out*.println("\nФайл g:");  
 for (int i = 0; i < mas.size()/2; i++){  
 System.*out*.print(plus.get(i) + " ");  
 System.*out*.print(minus.get(i) + " ");  
 g.write(plus.get(i) + " ");  
 g.write(minus.get(i) + " ");  
 }  
 }  
  
 }  
  
  
 public static void hard()throws IOException{  
 System.*out*.println("\n\nВысокий уровень:");  
 String line;  
 String chislo;  
 int m = 0;  
 int n = 0;  
 int l = 0;  
 int k = 0;  
 ArrayList<ArrayList<ArrayList<String>>> file1 = new ArrayList<ArrayList<ArrayList<String>>>();  
 ArrayList<ArrayList<ArrayList<String>>> file2 = new ArrayList<ArrayList<ArrayList<String>>>();  
 ArrayList<Integer> mas1 = new ArrayList<>();  
 ArrayList<Integer> mas2 = new ArrayList<>();  
 ArrayList<Integer> result1 = new ArrayList<>();  
 ArrayList<Integer> result2 = new ArrayList<>();  
  
  
 try (BufferedReader dano\_file = new BufferedReader(new FileReader("file1.txt"))) {  
 //чтение матриц  
 file1.add(new ArrayList<ArrayList<String>>());  
 while ((line = dano\_file.readLine()) != null) {  
 if (line.length() == 0) {  
 file1.add(new ArrayList<ArrayList<String>>());  
 continue;  
 }  
 file1.get(file1.size() - 1).add(new ArrayList<String>());  
 chislo = "";  
 for (int i = 0; i < line.length(); i++) {  
 if (line.charAt(i) == ' ') {  
 file1.get(file1.size() - 1).get((file1.get(file1.size() - 1)).size() - 1).add(chislo);  
 chislo = "";  
 continue;  
 }  
 chislo += line.charAt(i);  
 }  
 file1.get(file1.size() - 1).get((file1.get(file1.size() - 1)).size() - 1).add(chislo);  
 }  
 }  
 try (BufferedReader dano1\_file = new BufferedReader(new FileReader("file2.txt"))) {  
 //чтение матриц  
 file2.add(new ArrayList<ArrayList<String>>());  
 while ((line = dano1\_file.readLine()) != null) {  
 if (line.length() == 0) {  
 file2.add(new ArrayList<ArrayList<String>>());  
 continue;  
 }  
 file2.get(file2.size() - 1).add(new ArrayList<String>());  
 chislo = "";  
 for (int i = 0; i < line.length(); i++) {  
 if (line.charAt(i) == ' ') {  
 file2.get(file2.size() - 1).get((file2.get(file2.size() - 1)).size() - 1).add(chislo);  
 chislo = "";  
 continue;  
 }  
 chislo += line.charAt(i);  
 }  
 file2.get(file2.size() - 1).get((file2.get(file2.size() - 1)).size() - 1).add(chislo);  
 }  
 }  
  
 System.*out*.println("Матрицы из первого файла:");  
 for (int a = 0; a < file1.size(); a++) {  
 for (int i = 0; i < file1.get(a).size(); i++) {  
 for (int j = 0; j < file1.get(a).get(i).size(); j++) {  
 System.*out*.print(file1.get(a).get(i).get(j) + " ");  
 }  
 System.*out*.println();  
 }  
 System.*out*.println();  
 }  
  
 System.*out*.println("Матрицы из второго файла:");  
 for (int a = 0; a < file2.size(); a++) {  
 for (int i = 0; i < file2.get(a).size(); i++) {  
 for (int j = 0; j < file2.get(a).get(i).size(); j++) {  
 System.*out*.print(file2.get(a).get(i).get(j) + " ");  
 }  
 System.*out*.println();  
 }  
 System.*out*.println();  
 }  
  
 for (int a = 0; a < file1.size(); a++) {  
 for (int i = 0; i < file1.get(a).size(); i++) {  
 for (int j = 0; j < file1.get(a).get(i).size(); j++) {  
 mas1.add(Integer.*parseInt*(file1.get(a).get(i).get(j)));  
 }  
 }  
 }  
  
 for (int i = 0; i < mas1.size()/2; i++){  
 result1.add(mas1.get(i)-mas1.get((mas1.size()/2)+i));  
 }  
  
  
 for (int a = 0; a < file2.size(); a++) {  
 for (int i = 0; i < file2.get(a).size(); i++) {  
 for (int j = 0; j < file2.get(a).get(i).size(); j++) {  
 mas2.add(Integer.*parseInt*(file2.get(a).get(i).get(j)));  
 }  
 }  
 }  
  
 for (int i = 0; i < mas2.size()/2; i++){  
 result2.add(mas2.get(i)-mas2.get((mas2.size()/2)+i));  
 }  
  
 try(FileWriter file3 = new FileWriter("file3.txt", false)) {  
 System.*out*.println("Третий файл:");  
 k = file1.size();  
 System.*out*.println("k = " + k);  
 file3.write("k = " + k +"\n");  
 m = file1.get(0).get(0).size();  
 System.*out*.println("m = " + m);  
 file3.write("m = " + m +"\n");  
 n = file1.get(0).size();  
 System.*out*.println("n = " + n);  
 file3.write("n = " + n +"\n");  
 l = file2.get(0).size();  
 System.*out*.println("l = " + l);  
 file3.write("l = " + l +"\n");  
  
 System.*out*.println("Разность матриц из первого файла:");  
 file3.write("Разность матриц из первого файла\n");  
 int check = 1;  
 for (int i = 0; i < result1.size(); i++) {  
 System.*out*.print(result1.get(i) + " ");  
 file3.write(result1.get(i) + " ");  
 if ((check + i) % m == 0){  
 System.*out*.println();  
 file3.write("\n");  
 }  
 }  
 System.*out*.println("Разность матриц из второго файла:");  
 file3.write("Разность матриц из второго файла\n");  
 check = 1;  
 for (int i = 0; i < result2.size(); i++) {  
 System.*out*.print(result2.get(i) + " ");  
 file3.write(result2.get(i) + " ");  
 if ((check + i) % m == 0){  
 System.*out*.println();  
 file3.write("\n");  
 }  
 }  
 }  
  
 }  
  
 public static void main(String[] args) throws IOException {  
 System.*out*.println("Вариант №23");  
 *easy*();  
 *medium*();  
 *hard*();  
 }  
 }

8.2.

package com.company;  
  
import java.io.\*;  
import java.util.ArrayList;  
import java.util.Scanner;  
  
public class Main {  
  
 public static class Football {  
 private String surname;  
 private String role;  
 private String year;  
 private String sum\_game;  
 private String sum\_goal;  
  
 public String getSurname() {  
 return surname;  
 }  
 public void setSurname(String surname) {  
 this.surname = surname;  
 }  
 public String getRole() {  
 return role;  
 }  
 public void setRole(String role) {  
 this.role = role;  
 }  
 public String getYear() {  
 return year;  
 }  
 public void setYear(String year) {  
 this.year = year;  
 }  
 public String getSum\_game() {  
 return sum\_game;  
 }  
 public void setSum\_game(String sum\_game) {  
 this.sum\_game = sum\_game;  
 }  
 public String getSum\_goal() {  
 return sum\_goal;  
 }  
 public void setSum\_goal(String sum\_goal) {  
 this.sum\_goal = sum\_goal;  
 }  
  
 public void player(){  
 String[] surname = getSurname().split("(?<=\\G.{8})");  
 String[] role = getRole().split("(?<=\\G.{8})");  
 String[] year = getYear().split("(?<=\\G.{8})");  
 String[] sum\_game = getSum\_game().split("(?<=\\G.{8})");  
 String[] sum\_goal = getSum\_goal().split("(?<=\\G.{8})");  
  
 String s = null;  
 String r = null;  
 String y = null;  
 String sgm = null;  
 String sgl = null;  
  
 System.*out*.print("\n\nФамилия: ");  
 for (int i = 0; i < surname.length; i++) {  
 int count = Integer.*parseInt*(surname[i], 2);  
 s = new Character((char)count).toString();  
 System.*out*.print(s);  
 }  
 System.*out*.print("\nАмплуа: ");  
 for (int i = 0; i < role.length; i++) {  
 int count = Integer.*parseInt*(role[i], 2);  
 r = new Character((char)count).toString();  
 System.*out*.print(r);  
 }  
 System.*out*.print("\nГод рождения: ");  
 for (int i = 0; i < year.length; i++) {  
 int count = Integer.*parseInt*(year[i], 2);  
 y = new Character((char)count).toString();  
 System.*out*.print(y);  
 }  
 System.*out*.print("\nКоличество игр, проведенных игроком: ");  
 for (int i = 0; i < sum\_game.length; i++) {  
 int count = Integer.*parseInt*(sum\_game[i], 2);  
 sgm = new Character((char)count).toString();  
 System.*out*.print(sgm);  
 }  
 System.*out*.print("\nКоличество голов, забитых игроком: ");  
 for (int i = 0; i < sum\_goal.length; i++) {  
 int count = Integer.*parseInt*(sum\_goal[i], 2);  
 sgl = new Character((char)count).toString();  
 System.*out*.print(sgl);  
 }  
  
 }  
 }  
  
 public static void easy() throws IOException {  
 System.*out*.println("Базовый уровень:");  
 BufferedReader reader = new BufferedReader(new FileReader("easy.txt"));  
  
 String line = null;  
 Scanner scanner = null;  
 int index = 0;  
 int j = 0;  
  
 Football[] player = new Football[3];  
  
 while ((line = reader.readLine()) != null) {  
 player[j] = new Football();  
 scanner = new Scanner(line);  
 scanner.useDelimiter(";");  
 while (scanner.hasNext()) {  
 String data = scanner.next();  
 if (index == 0)  
 player[j].setSurname(data);  
 else if (index == 1)  
 player[j].setRole(data);  
 else if (index == 2)  
 player[j].setYear(data);  
 else if (index == 3)  
 player[j].setSum\_game(data);  
 else if (index == 4)  
 player[j].setSum\_goal(data);  
 else  
 System.*out*.println("null" + data);  
 index++;  
 }  
 j++;  
 index = 0;  
 }  
  
 for (int n = 0; n<j; n++){  
 int goal = 0;  
 String[] sum\_goal = player[0].sum\_goal.split("(?<=\\G.{8})");  
 for (int i = 0; i < sum\_goal.length; i++) {  
 int count = Integer.*parseInt*(sum\_goal[i], 2);  
 String str = new Character((char)count).toString();  
 goal = Integer.*parseInt*(str);  
 }  
 if(goal >= 5)player[n].player();  
 }  
 }  
  
  
  
 public static void medium()throws IOException {  
 System.*out*.println("\n\nСредний уровень:");  
 ArrayList <Integer> dano = new ArrayList<Integer>();  
 dano.add(5);  
 dano.add(-99);  
 dano.add(3);  
 dano.add(-9);  
 dano.add(1);  
  
  
 try {  
 FileOutputStream medium = new FileOutputStream("medium.txt");  
 System.*out*.print("Числа до изменения файла: ");  
 for(int i : dano) {  
 System.*out*.print(i +" ");  
 medium.write((i + " ").getBytes());  
 }  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
  
 try {  
 FileOutputStream medium = new FileOutputStream("medium\_result.txt");  
 for(int i : dano) {  
 medium.write((i + " ").getBytes());  
 }  
 medium.write(("\n").getBytes());  
  
 System.*out*.print("\nЧисла после изменения файла: ");  
 for (int i : dano){  
 i \*= -1;  
 System.*out*.print(i +" ");  
 medium.write((i + " ").getBytes());  
 }  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
  
  
  
 }  
  
 public static class Subject {  
 private String name;  
 private String semester;  
 private String course;  
 private String name\_binary;  
 private String semester\_binary;  
 private String course\_binary;  
  
 public String getName() {  
 return name;  
 }  
 public String getName\_binary() {  
 return name\_binary;  
 }  
 public void setName(String name) {  
 this.name = name;  
 this.name\_binary = name;  
 }  
 public String getSemester() {  
 return semester;  
 }  
 public String getSemester\_binary() {  
 return semester\_binary;  
 }  
 public void setSemester(String semester) {  
 this.semester = semester;  
 this.semester\_binary = semester;  
 }  
 public void setSemester\_binary(String semester) {  
 this.semester\_binary = semester;  
 }  
 public String getCourse() {  
 return course;  
 }  
 public String getCourse\_binary() {  
 return course\_binary;  
 }  
 public void setCourse(String course) {  
 this.course = course;  
 this.course\_binary = course;  
 }  
 public void setCourse\_binary(String course) {  
 this.course\_binary = course;  
 }  
  
 public void translate(){  
 String[] name = getName().split("(?<=\\G.{8})");  
 String[] semester = getSemester().split("(?<=\\G.{8})");  
 String[] course = getCourse().split("(?<=\\G.{8})");  
 String [] Name = new String[name.length];  
 String [] Semester = new String[semester.length];  
 String [] Course = new String[course.length];  
 for (int i = 0; i < name.length; i++) {  
 int count = Integer.*parseInt*(name[i], 2);  
 Name[i] = new Character((char)count).toString();  
 }  
 for (int i = 0; i < semester.length; i++) {  
 int count = Integer.*parseInt*(semester[i], 2);  
 Semester[i] = new Character((char)count).toString();  
 }  
 for (int i = 0; i < course.length; i++) {  
 int count = Integer.*parseInt*(course[i], 2);  
 Course[i] = new Character((char)count).toString();  
 }  
  
 this.name = String.*join*("", Name);  
 this.semester = String.*join*("", Semester);  
 this.course = String.*join*("", Course);  
 }  
 public void subject(){  
 System.*out*.print("\n\nНазвание дисциплины: ");  
 System.*out*.print(this.name);  
 System.*out*.print("\nСеместр: ");  
 System.*out*.print(this.semester);  
 System.*out*.print("\nКурс: ");  
 System.*out*.print(this.course);  
 }  
 }  
  
 public static void hard()throws IOException{  
 System.*out*.println("\n\nВысокий уровень:");  
 BufferedReader reader = new BufferedReader(new FileReader("hard.txt"));  
 ArrayList<String> list = null;  
 String line = null;  
 Scanner scanner = null;  
 int index = 0;  
 int j = 0;  
  
 Subject[] subject = new Subject[3];  
  
 while ((line = reader.readLine()) != null) {  
 subject[j] = new Subject();  
 scanner = new Scanner(line);  
 scanner.useDelimiter(";");  
 while (scanner.hasNext()) {  
 String data = scanner.next();  
 if (index == 0)  
 subject[j].setName(data);  
 else if (index == 1)  
 subject[j].setSemester(data);  
 else if (index == 2)  
 subject[j].setCourse(data);  
 else  
 System.*out*.println("null" + data);  
 index++;  
 }  
 j++;  
 index = 0;  
 }  
 System.*out*.print("Исходный файл:");  
 for(int i = 0; i < j; i++ ){  
 subject[i].translate();  
 subject[i].subject();  
 }  
  
 System.*out*.print("\n\nРезультат:");  
 for(int i = 0; i < j; i++ ) {  
 if(subject[i].getName().equals("Programming\_technology")){  
 subject[i].setSemester("1");  
 subject[i].setCourse("3");  
 subject[i].setSemester\_binary("110001");  
 subject[i].setCourse\_binary("110011");  
 }  
 subject[i].subject();  
 }  
 try {  
 FileOutputStream hard\_new = new FileOutputStream("hard\_new.txt");  
 for(int i = 0; i < j; i++ ) {  
 hard\_new.write((subject[i].getName\_binary()+";"+subject[i].getSemester\_binary()+";"+subject[i].getCourse\_binary()+";\n").getBytes());  
 }  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
 }  
  
 public static void main(String[] args) throws IOException {  
 System.*out*.println("Вариант №23");  
 *easy*();  
 *medium*();  
 *hard*();  
 }  
}