

## 1. Последовательность

1 секунда

20 очков

Дана последовательность целых чисел  $A_1, A_2, \dots, A_N$ . Определим на основе этой последовательности новую последовательность  $B_1, B_2, \dots, B_N$ , где  $B_i = A_1 \cdot A_2 \cdot \dots \cdot A_{i-1} \cdot A_{i+1} \cdot \dots \cdot A_N$ . Написать программу для вычисления остатков членов новой последовательности при делении на  $10^8$ .

**Входные данные.** На первой строке текстового файла `jada.sis` дана длина последовательности  $N$  ( $1 \leq N \leq 50\,000$ ), а на следующих  $N$  строках даны числа  $A_i$  ( $0 \leq A_i < 100\,000\,000$ ), по одной на строке.

**Выходные данные.** В текстовый файл `jada.val` вывести в точности  $N$  строк. На строке  $i$  необходимо вывести одно целое число — остаток от деления  $B_i$  на  $100\,000\,000$ .

<b>Пример.</b>	<code>jada.sis</code>	<code>jada.val</code>
	4	24
	1	12
	2	8
	3	6
	4	

## 2. Резисторы

1 секунда

40 очков

Резистор — это электронный компонент, основной характеристикой которого является сопротивление. Сопротивление отмечается на резисторе цветовым кодом, в котором для обозначения цифр используются десять разных цветов.

На каждом резисторе нанесены три полосы. Первые две обозначают две первые значимые цифры сопротивления, а третья указывает степень десяти, на которую их нужно домножить. Т.е., если на резисторе нанесены полосы, соответствующие цифрам  $x$ ,  $y$  и  $z$ , то сопротивление данного резистора равно  $(10 \cdot x + y) \cdot 10^z$ .

У Электроника есть много резисторов. Он знает значения сопротивлений этих резисторов, но забыл, какое значение соответствует какому резистору. Написать программу, которая восстановит по имеющимся данным используемый на резисторах цветовой код.

**Входные данные.** На первой строке текстового файла `taki.sis` дано число резисторов  $N$  ( $1 \leq N \leq 1000$ ). На каждой из следующих  $N$  строк приведено описание одного резистора в виде трех заглавных латинских букв, где каждая из букв A...J обозначает один из цветов. На следующих  $N$  строках приведены сопротивления резисторов в виде целых чисел. Конечно же нельзя предполагать, что два списка даны в одном и том же порядке.

**Выходные данные.** В текстовый файл `taki.val` вывести числовые значения всех использованных во входных данных букв. Каждое значение необходимо вывести на отдельной строке, порядок строк значения не имеет. Если возможных решений несколько, можно вывести любое из них. Можно предполагать что хотя бы одно решение обязательно найдется.

<b>Пример.</b>	<code>taki.sis</code>	<code>taki.val</code>
	2	A=1
	ABC	B=0
	DDD	C=4
	100000	D=3
	33000	

Код ABC обозначает сопротивление  $10 \cdot 10^4 = 100\,000$ , а код DDD —  $33 \cdot 10^3 = 33\,000$ .

### 3. Треугольники

40 очков

Учитель дал Кате следующее задание:

Написать программу, которая проверяет, возможно ли поместить два треугольника с заданными длинами сторон один на другой таким образом, чтобы верхний нигде не выступал за края нижнего.

**Входные данные.** На первой строке входного файла даны длины сторон первого, а на второй — длины сторон второго треугольника. Все длины даны в виде целых чисел  $1 \dots 10\,000$ , числа на одной строке разделены пробелами.

**Выходные данные.** На единственной строке выходного файла вывести **EI SAA**, если треугольники невозможно поместить один на другой требуемым образом. Если же треугольники можно поместить один на другой требуемым образом, вывести **SAAB**. При этом не важно, какой из треугольников будет сверху, а какой — снизу.

Катя написала решение и хочет протестировать его перед тем как сдать учителю. Помогите ей создать для этого тестовые данные.

**Оценивание.** В этом задании в качестве решения надо предоставить **один текстовый файл**, в котором описано **вплоть до 20** тестов. Описание каждого теста должно состоять в точности из трех строк: на первых двух строках даны входные данные теста, а на третьей — соответствующий этим входным данным результат.

При оценивании будет дан штраф в минус 2 очка за каждый частный случай, неправильная обработка которого не может быть обнаружена предложенным тестовым комплектом и также минус 4 очка за каждый некорректный тест в комплекте. Тест считается некорректным в случае если входные или выходные данные не соответствуют приведенным требованиям или если приведенный в нем ответ не соответствует правильному ответу. Если сумма штрафов превышает стоимость задачи, участник получает за задачу ноль очков.

**Пример.**            Выходной файл

```
1 2 3
4 5 6
SAAB
1 10 10
5 5 5
EI SAA
1 10 10
5 5 5
SAAB
```

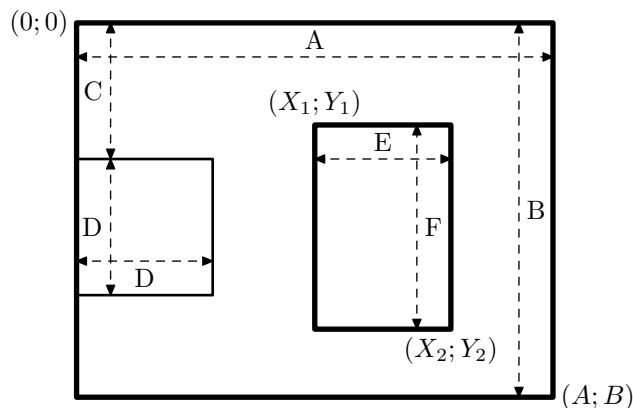
В приведенном в примере файле даны три теста. Из них два первых корректны, а третий — некорректен.

## 1. Письменный стол

1 секунда

20 очков

Митя хочет обставить свою комнату. Пол комнаты покрыт паркетом, состоящим из квадратных плиток. Комната имеет форму прямоугольника длиной  $A$  плиток и шириной  $B$  плиток, как на приведенном рисунке. В западной стене комнаты, на расстоянии  $C$  от северного угла, находится дверь шириной  $D$ . Перед дверью необходимо оставить свободное пространство размером  $D \times D$  плиток.



Необходимо узнать, возможно ли поместить в комнату стол размером  $E \times F$  плитки так, чтобы он не закрывал проход у двери. Стол необходимо расположить так, чтобы он занимал в точности  $E \times F$  целых плитки, это значит что его можно поворачивать на угол кратный  $90^\circ$  и двигать на расстояние, кратное ширине плитки.

**Входные данные.** На первой строке текстового файла `laud.sis` даны разделенные пробелом целые числа  $A$ ,  $B$ ,  $C$  и  $D$  ( $1 \leq A \leq 10\,000$ ,  $1 \leq B \leq 10\,000$ ,  $0 \leq D \leq A$ ,  $0 \leq C \leq C + D \leq B$ ), где  $A$  и  $B$  описывают размеры комнаты, а  $C$  и  $D$  — расположение и размер двери. На второй строке файла даны разделенные пробелом целые числа  $E$  и  $F$  ( $1 \leq E \leq 10\,000$ ,  $1 \leq F \leq 10\,000$ ), которые описывают размер стола.

**Выходные данные.** Если расположить стол требуемым образом возможно, вывести на первой строке файла `laud.val` слово `JAH`, а на второй и третьей строках вывести по два разделенных пробелом числа,  $X_1$ ,  $Y_1$ , и  $X_2$ ,  $Y_2$ . Числа на второй строке указывают координаты северо-западного угла стола, а на третьей строке — координаты юго-восточного угла. В используемой системе координат северо-западному углу комнаты соответствуют координаты  $(0; 0)$ , а юго-восточному —  $(A; B)$ . Если существует несколько способов разместить стол, достаточно вывести любой из них. Если стол разместить невозможно, на единственной строке выходного файла вывести слово `EI`.

<b>Пример.</b>	<code>laud.sis</code>	<code>laud.val</code>
	14 11 4 4	JAH
	4 6	7 3
		11 9

Пример соответствует приведенному в тексте задачи рисунку.

<b>Пример.</b>	<code>laud.sis</code>	<code>laud.val</code>
	14 11 4 4	EI
	15 15	

**Оценивание.** В этой задаче за тесты с ответом `EI` получают очки лишь те решения, которые пройдут хотя бы один тест с ответом `JAH`.

## 2. Табличные вычисления

1 секунда

40 очков

В программе для табличных вычислений первые 26 столбцов рабочей таблицы обозначаются латинскими буквами  $A \dots Z$ , следующие столбцы — двухбуквенными комбинациями  $AA \dots ZZ$ , следующие — трехбуквенными, и так далее. Написать программу для преобразования порядкового номера столбца в его буквенное обозначение и наоборот.

**Входные данные.** На единственной строке текстового файла `tab2.sis` дан либо порядковый номер искомого столбца,  $1 \dots 1\,000\,000\,000$ , либо его буквенное обозначение, состоящее из заглавных латинских букв. Если в файле дан порядковый номер, необходимо преобразовать его в соответствующее буквенное обозначение и наоборот.

**Выходные данные.** На единственной строке текстового файла `tab2.val` вывести требуемый порядковый номер или буквенное обозначение. При нахождении порядкового номера можно предполагать, что результат не превышает  $10^9$ .

<b>Пример.</b>	<code>tab2.sis</code>	<code>tab2.val</code>
	10	J

<b>Пример.</b>	<code>tab2.sis</code>	<code>tab2.val</code>
	AA	27

### 3. Телефонные номера

1 секунда

40 очков

Телефонные компании должны выдавать телефонные номера своим клиентам таким образом, чтобы ни один телефонный номер не был бы префиксом (т.е. началом) какого-либо другого номера.

Например, если бы у трех клиентов оказались телефонные номера 123, 1234, 1235, то при наборе цифр 1, 2, 3, у телефонного центра не было бы возможности определить, нужно ли соединять звонящего с первым клиентом или ждать пока он не наберет четвертую цифру.

В то же время, если короткий номер совпадает с серединой какого-либо уже имеющегося номера, это не мешает работе телефонного центра. Например, если три клиента используют номера 1234, 1235, 23, то после набора цифр 2, 3, телефонный центр сразу знает, что это звонок третьему клиенту.

Написать программу, которая проверяет заданный список телефонных номеров на то, чтобы ни один из номеров не совпадал с префиксом ни одного другого.

**Входные данные.** На первой строке текстового файла `tel2.sis` дано число интересующих нас телефонных номеров  $N$  ( $0 \leq N \leq 50\,000$ ), а на каждой из следующих  $N$  строк приведено по одному телефонному номеру. Каждый телефонный номер состоит из последовательности цифр  $0 \dots 9$  длиной от 1 до 10, и может начинаться с нулей.

**Выходные данные.** В первой строке текстового файла `tel2.val` вывести слово `EI`, если ни один из приведенных во входном файле номеров не совпадает с префиксом какого-либо другого номера. Если какой-либо из номеров совпадает с началом другого номера, на первой строке файла вывести слово `JAH`, а на второй и третьей строках — два телефонных номера, таких что номер, выведенный на второй строке совпадает с префиксом номера, выведенного на третьей строке. Если подходящих пар несколько, можно вывести любую из них.

<b>Пример.</b>	<code>tel2.sis</code>	<code>tel2.val</code>
	3	JAH
	123	123
	1234	1234
	1235	

<b>Пример.</b>	<code>tel2.sis</code>	<code>tel2.val</code>
	3	EI
	1234	
	1235	
	23	

**Оценивание.** В этой задаче за тесты с ответом `EI` получают очки лишь те решения, которые пройдут хотя бы один тест с ответом `JAH`.

## 1. Гипсовые платы

1 секунда 20 очков

Написать программу, которая для двух прямоугольных гипсовых плат определит, возможно ли положить одну из них на другую так, чтобы верхняя плата не выступала за края нижней. Платы можно поворачивать относительно друг друга только на угол, кратный  $90^\circ$ .

**Входные данные.** На первой строке текстового файла `kips.sis` дана длина и ширина первой платы, на второй строке — длина и ширина второй платы. Все размеры даны в виде целых чисел  $1 \dots 10\,000$ . Числа, находящиеся на одной строке разделены пробелом.

**Выходные данные.** На единственной строке текстового файла `kips.val` вывести `EI SAA`, если платы невозможно положить одну на другую требуемым образом. Если первую плату можно положить на вторую, вывести `2 1`. Если вторую плату можно положить на первую, вывести `1 2`. Если платы можно положить одну на другую в любом порядке, вывести любой ответ (либо `2 1`, либо `1 2`).

**Пример.**

<code>kips.sis</code>	<code>kips.val</code>
1 4	EI SAA
2 3	

**Пример.**

<code>kips.sis</code>	<code>kips.val</code>
1 4	2 1
2 5	

**Пример.**

<code>kips.sis</code>	<code>kips.val</code>
2 5	1 2
1 4	

## 2. Табличные вычисления

1 секунда 40 очков

В программе для табличных вычислений первые 26 столбцов рабочей таблицы обозначаются латинскими буквами `A..Z`, а последующие столбцы — двухбуквенными комбинациями `AA..ZZ`, в порядке, соответствующем латинскому алфавиту. Написать программу для преобразования порядкового номера столбца в его буквенное обозначение и наоборот.

**Входные данные.** На первой строке текстового файла `tab1.sis` даны две буквы, либо `NT`, либо `TN`. Если на первой строке файла даны буквы `NT`, то на второй строке дан порядковый номер искомого столбца,  $1 \dots 702$ , и требуется найти буквенное обозначение этого столбца. Если же на первой строке файла даны буквы `TN`, то на второй строке дано обозначение столбца, состоящее из одной или двух заглавных латинских букв, и необходимо найти порядковый номер этого столбца.

**Выходные данные.** На единственной строке текстового файла `tab1.val` вывести требуемый порядковый номер или буквенное обозначение.

**Пример.**

<code>tab1.sis</code>	<code>tab1.val</code>
NT	J
10	

**Пример.**

<code>tab1.sis</code>	<code>tab1.val</code>
TN	27
AA	

### 3. Телефонные номера

1 секунда

40 очков

Телефонные компании должны выдавать телефонные номера своим клиентам таким образом, чтобы ни один телефонный номер не был бы префиксом (т.е. началом) какого-либо другого номера.

Например, если бы у трех клиентов оказались телефонные номера 123, 1234, 1235, то при наборе цифр 1, 2, 3, у телефонного центра не было бы возможности определить, нужно ли соединять звонящего с первым клиентом или ждать пока он не наберет четвертую цифру.

В то же время, если короткий номер совпадает с серединой какого-либо уже имеющегося номера, это не мешает работе телефонного центра. Например, если три клиента используют номера 1234, 1235, 23, то после набора цифр 2, 3, телефонный центр сразу знает, что это звонок третьему клиенту.

Написать программу, которая проверяет заданный список телефонных номеров на то, чтобы ни один из номеров не совпадал с префиксом ни одного другого.

**Входные данные.** На первой строке текстового файла `tel1.sis` дано число интересующих нас телефонных номеров  $N$  ( $0 \leq N \leq 1000$ ), а на каждой из следующих  $N$  строк приведено по одному телефонному номеру. Каждый телефонный номер состоит из последовательности цифр  $0 \dots 9$  длиной от 1 до 10, и может начинаться с нулей.

**Выходные данные.** В первой строке текстового файла `tel1.val` вывести слово `EI`, если ни один из приведенных во входном файле номеров не совпадает с префиксом какого-либо другого номера. Если какой-либо из номеров совпадает с началом другого номера, на первой строке файла вывести слово `JAH`, а на второй и третьей строках — два телефонных номера, таких что номер, выведенный на второй строке совпадает с префиксом номера, выведенного на третьей строке. Если подходящих пар несколько, можно вывести любую из них.

<b>Пример.</b>	<code>tel1.sis</code>	<code>tel1.val</code>
	3	JAH
	123	123
	1234	1234
	1235	

<b>Пример.</b>	<code>tel1.sis</code>	<code>tel1.val</code>
	3	EI
	1234	
	1235	
	23	

**Оценивание.** В этой задаче за тесты с ответом `EI` получают очки лишь те решения, которые пройдут хотя бы один тест с ответом `JAH`.