

## 1. Такси

1 секунда 10 баллов

Пассажир спешит на поезд, который отходит через  $T$  минут со станции, находящейся на расстоянии  $S$  км. Он может поехать на станцию на такси со скоростью  $V_1$  км/ч по цене  $R_1$  евро/км, пойти пешком со скоростью  $V_2$  км/ч совершенно бесплатно или комбинировать эти способы передвижения.

Написать программу, которая найдёт минимальную сумму денег, которую должен потратить пассажир, чтобы успеть на поезд.

**Входные данные.** В первой строке текстового файла `takso.sis` находятся расстояние до станции  $S$  и время до отправления поезда  $T$ . Во второй строке файла — скорость такси  $V_1$  и тариф  $R_1$ . В третьей — скорость пассажира пешим ходом  $V_2$ . Все числа целые, положительные и не превышают 10 000.

**Выходные данные.** В единственную строку текстового файла `takso.val` вывести минимальную сумму денег (с ровно двумя цифрами после запятой), которую нужно потратить, чтобы успеть на поезд, или текст `EI SAA`, если пассажир в любом случае не успеет на поезд.

<b>Пример.</b>	<code>takso.sis</code>	<code>takso.val</code>
	8 10	7.50
	90 1	
	6	

На такси со скоростью 90 км/ч пассажир проедет 7,5 км за 5 минут, а оставшиеся 0,5 км успеет пройти за 5 минут со скоростью 6 км/ч. Поездка на такси на 7,5 км стоит 7.50 евро.

<b>Пример.</b>	<code>takso.sis</code>	<code>takso.val</code>
	20 10	EI SAA
	90 1	
	6	

**Оценивание.** В этом задании за тесты с ответами `EI SAA` или `0.00` получают баллы только те программы, которые правильно решат хотя бы один тест, в котором ответ иной.

## 2. Таблица

1 секунда 20 баллов

Людам, которым часто нужно преобразовывать значения из одной единицы измерения в другую и обратно (например, переводить расстояния из километров в мили и обратно), удобно использовать для этого таблицу с двумя столбцами, где в порядке возрастания значений расположены все такие строки, где по крайней мере в одном столбце стоит целое число.

Написать программу, которая на основании отношения единиц измерения  $R$  (т.е. одна единица первого столбца равна  $R$  единиц второго столбца) возвращает  $N$  первых строк искомой (бесконечной) таблицы преобразований.

**Входные данные.** В единственной строке текстового файла `tabel.sis` находятся действительное число  $R$  ( $10^{-3} \leq R \leq 10^3$ ) — отношение единиц измерения, и целое число  $N$  ( $1 \leq N \leq 10\,000$ ) — количество выводимых строк.

**Выходные данные.** В текстовый файл `tabel.val` вывести точно  $N$  строк, в каждую — одну строку таблицы с двумя действительными числами, разделёнными пробелом. Выведенные значения не должны отличаться от точных больше чем на  $10^{-4}$ . Числа можно вывести в любых стандартных форматах вещественных чисел.

<b>Пример.</b>	<code>tabel.sis</code>	<code>tabel.val</code>
	1.6093 5	0.6214 1.0000 1.0000 1.6093 1.2428 2.0000 1.8642 3.0000 2.0000 3.2186

Преобразование расстояния из миль в километры (исходя из предположения, что 1 миля равна 1,6093 км).

### 3. Электрическая схема

1 секунда

30 баллов

В цифровой электронике для представления данных используются наличие или отсутствие напряжения в различных точках схемы. При этом часть точек являются входами в схему, часть выходами, а часть — внутренние точки схемы.

Данные обрабатываются находящимися между точками схемы логическими элементами, у которых в свою очередь есть входы и выходы. Наличие или отсутствие напряжения на выходе каждого элемента зависит от его наличия или отсутствия на входе или входах.

В этом задании рассмотрим схемы, которые образуются из элементов AND, OR и NOT. У элемента AND есть два входа и один выход. На его выходе есть напряжение тогда и только тогда, когда оно есть на обоих входах. У элемента OR также два входа и один выход, но у него на выходе есть напряжение тогда и только тогда, когда оно есть по крайней мере на одном входе. У элемента NOT один вход и один выход и у него на выходе есть напряжение тогда и только тогда, когда оно отсутствует на входе.

Написать программу, которая получает описание схемы и состояния её входов и рассчитывает на основании этих данных состояния её выходов.

**Входные данные.** В первой строке текстового файла `skeem.sis` находятся количество входов схемы  $S$  ( $1 \leq S \leq 10$ ), количество внутренних точек  $P$  ( $0 \leq P \leq 1000$ ) и количество выходов  $V$  ( $1 \leq V \leq 10$ ). Входы схемы пронумерованы  $1 \dots S$ , внутренние точки  $S + 1 \dots S + P$  и выходы  $S + P + 1 \dots S + P + V$ .

Во второй строке — количество логических элементов в схеме  $N$  ( $1 \leq N \leq 1000$ ) и в каждой из следующих  $N$  строк описание одного элемента:

- строка вида AND  $s_1 s_2 v$  обозначает элемент AND, входы которого в точках  $s_1$  и  $s_2$ , а выход в точке  $v$ ;
- строка вида OR  $s_1 s_2 v$  обозначает элемент OR, входы которого в точках  $s_1$  и  $s_2$ , а выход в точке  $v$ ;
- строка вида NOT  $s v$  обозначает элемент NOT, вход которого в точке  $s$ , а выход в точке  $v$ .

В последней строке файла находятся  $S$  целых чисел 0 или 1, где число 1 обозначает наличие, а 0 — отсутствие напряжения на соответствующем входе схемы.

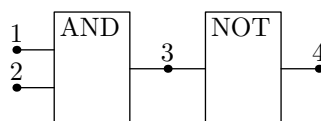
Можно считать, что в каждой внутренней точке и в каждом выходе схемы находится выход ровно одного элемента, а также вход любого элемента не зависит напрямую или косвенно от выхода этого же элемента (в схеме нет циклов).

**Выходные данные.** В единственную строку текстового файла `skeem.val` вывести  $V$  целых чисел 0 или 1, где 1 обозначает наличие, а 0 — отсутствие напряжения на соответствующем выходе схемы.

**Пример.**

	<code>skeem.sis</code>	<code>skeem.val</code>
	2 1 1	0
	2	
	AND 1 2 3	
	NOT 3 4	
	1 1	

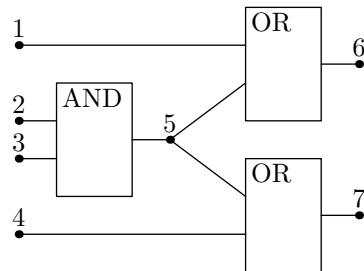
Схему иллюстрирует следующий рисунок. Так как на обоих входах элемента AND (входы 1 и 2) есть напряжение, оно есть и на его выходе (внутренняя точка 3). Так как на входе элемента NOT (внутренняя точка 3) есть напряжение, то на его выходе (выход 4) нет напряжения.



**Пример.**

skeem.sis	skeem.val
4 1 2	1 0
3	
AND 2 3 5	
OR 1 5 6	
OR 4 5 7	
1 0 1 0	

Схему иллюстрирует следующий рисунок. Так как на одном входе (вход 1) верхнего элемента OR есть напряжение, оно есть и на его выходе (выход 6). Так как на одном входе (вход 2) элемента AND напряжения нет, его нет и на выходе этого элемента (внутренняя точка 5). Так как на обоих входах (внутренняя точка 5 и вход 4) нижнего элемента OR нет напряжения, то его нет и на выходе этого элемента (выход 7).



## 4. Грабители и полиция

1 секунда

40 баллов

У банды грабителей есть сайт, где они обсуждают планы следующих ограблений. Для входа на сайт у каждого члена банды есть карточка паролей с  $N$  уникальными числами. Для входа на сайт нужно каждый раз вводить  $M$  из этих чисел. Точнее сервер указывает последовательность чисел  $a_1, a_2, \dots, a_M$ , где  $1 \leq a_1 < a_2 < \dots < a_M \leq N$ , и пользователь должен ввести со своей карточки числа с соответствующими порядковыми номерами.

Полиция следит за одним подозреваемым и им  $K$  раз удалось сохранить последовательность, которую вводил грабитель при входе на сайт. Конечно же, полицейские желают сами попасть на сайт, а для этого они должны уметь ввести правильную последовательность по требованию сервера.

Написать программу, которая на основании собранных данных угадывает содержание карточки паролей преступника.

**Входные данные.** В первой строке текстового файла `kripo.sis` находятся целые числа  $N$  ( $1 \leq N \leq 10^3$ ),  $M$  ( $1 \leq M \leq N$ ) и  $K$  ( $0 \leq K \cdot M \leq 10^6$ ). В каждой из следующих  $K$  строк находятся  $M$  положительных целых чисел (ни одно из которых не превышает  $10^9$ ) — одна из введённых подозреваемым последовательностей.

**Выходные данные.** В единственную строку текстового файла `kripo.val` вывести точно  $N$  целых чисел — все числа с карточки паролей в верном порядке. Если собранных полицией данных недостаточно для выяснения содержания карточки, вывести текст `EI SAA`.

<b>Пример.</b>	<code>kripo.sis</code>	<code>kripo.val</code>
	5 3 2	14 15 92 65 35
	14 15 92	
	92 65 35	

<b>Пример.</b>	<code>kripo.sis</code>	<code>kripo.val</code>
	5 3 2	EI SAA
	14 15 92	
	15 92 65	

**Оценивание.** В этом задании за тесты с ответами `EI SAA` получают баллы только те программы, которые правильно решат хотя бы один тест, в котором выяснение содержания карточки паролей возможно.

## 5. Мозаика

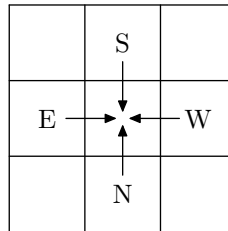
1 секунда

50 баллов

Рассмотрим головоломку, состоящую из квадратной игровой доски  $N \times N$  и  $N^2 - 1$  кубиков, которые обозначены  $1 \dots 9$  и  $A \dots Z$  (используется ровно столько обозначений сколько есть кубиков, в соответствии с размером доски).

В начале игры кубики расположены на доске так, что на каждом поле стоит один кубик, и только правый нижний угол пуст (см. рисунок слева). Цель: разместить кубики так, чтобы они, считая слева направо и сверху вниз, были упорядочены  $1 \dots 9$  и  $A \dots Z$ , а пустое место осталось бы по прежнему справа снизу (см. рисунок справа).

1	2	3
4	6	8
7	5	



1	2	3
4	5	6
7	8	

На каждом шагу игрок выбирает один кубик, у которого есть общая сторона с пустым полем и передвигает его на пустое поле. Таким образом, в любой ситуации есть максимально четыре возможных хода, которые обозначаются буквами S, E, W и N в зависимости от направления движения кубика (см. рисунок в центре).

Написать программу для решения этой головоломки.

**Входные данные.** В первой строке текстового файла `pusle.sis` находится целое число  $N$  ( $2 \leq N \leq 6$ ) — размер игровой доски. В каждой из следующих  $N$  строк находятся точно  $N$  знаков, которые описывают начальное расположение кубиков по строчкам (пустое поле обозначается точкой).

**Выходные данные.** В единственную строку текстового файла `pusle.val` вывести последовательность ходов решения или текст `EI SAA`, если решения нет. Не требуется, чтобы решение было кратчайшим.

**Пример.**

<code>pusle.sis</code>	<code>pusle.val</code>
2	SEW
23	
1.	

**Пример.**

<code>pusle.sis</code>	<code>pusle.val</code>
2	EI SAA
21	
3.	

**Оценивание.** В этом задании за тесты с ответами `EI SAA` получают баллы только те программы, которые правильно решат хотя бы один тест, где решение найдётся.

## 6. Дешифровка

50 баллов

Криптография или наука о шифрах — учение о способах кодирования информации с целью, чтобы третьи лица не могли её получить. За историю человечества для этого были придуманы различные алгоритмы, некоторые из которых оказались весьма слабыми — после некоторого исследования возможно угадать используемый алгоритм и заполучить секретную информацию. Такой взлом секретного шифра называется криптоанализом.

**Оценивание.** В этом задании даны три комплекта входных файлов `dekodtest.?.sis`. В каждом комплекте есть несколько схожим способом зашифрованных секретных послания, из которых для первого также дан и открытый текст (оригинальный текст в незашифрованном виде). Нужно раскодировать оставшиеся файлы. В качестве решения предъявить только расшифрованные файлы. Программу предъявлять не требуется, программа не оценивается.

Расшифровка файлов `dekodtest.1?.sis` даёт 15, расшифровка файлов `dekodtest.2?.sis` также 15, а расшифровка файлов `dekodtest.3?.sis` — 20 баллов.