

## Задача 1. Районная олимпиада по информатике

Имя входного файла:	input.txt
Имя выходного файла:	output.txt
Ограничение по времени:	1 секунда
Ограничение по памяти:	64 мегабайта

В районной олимпиаде по информатике (втором этапе республиканской олимпиады) участвуют команды из  $N$  районов. Команда  $i$ -го района состоит из  $P_i$  участников, а квота для участия в третьем этапе, выделенная этому району до начала соревнований, составляет  $Q_i$  человек. Однако эта квота может быть увеличена за счёт участников, набравших такое же количество баллов, что и участник с рангом  $Q_i$ . Так, если квота для какого-нибудь района составляет три участника, а пять лучших участников этого района набрали соответственно 286, 285.5, 285.5, 285.5, и 274 балла, то на третий этап олимпиады проходят четверо лучших участников второго этапа из этого района.

Кроме того, участник, не набравший ни одного балла на втором этапе олимпиады, не проходит на третий этап, даже если в квоте района остались свободные места.

Каждому участнику второго этапа предложено для решения четыре задачи. Максимальная оценка по каждой задаче составляет 100 баллов, причём баллы по задачам могут считаться с точностью до сотых долей.

Вам необходимо на основании информации об официальных результатах второго этапа олимпиады составить таблицу участников третьего этапа.

### Формат входных данных

Первая строка содержит величину  $N$  ( $2 \leq N \leq 200$ ). Во второй строке записаны  $N$  чисел — величины  $P_i$  ( $2 \leq P_i \leq 1000$ ,  $P = \sum_{i=1}^N P_i \leq 100000$ , для 50 % тестов  $P \leq 10000$ ). Третья строка содержит величины  $Q_i$  в таком же формате ( $0 < Q_i < P_i$ ).

Далее следуют  $P$  строк с результатами каждого участника олимпиады. Каждая из этих строк содержит шесть чисел: уникальный код участника (целое положительное число, не превосходящее  $10^6$ ), номер команды (целое число в интервале от 1 до  $N$ ), баллы, набранные за решения каждой задачи (неотрицательные числа).

Никакой сортировки строк с результатами соревнований нет. Тесты подобраны таким образом, что на третий этап олимпиады должен проходить хотя бы один школьник. Кроме того, в 40 % тестов баллы по задачам считаются с точностью до целых.

### Формат выходных данных

В первой строке выведите общее количество участников третьего этапа  $Q$ .

Каждая из последующих  $Q$  строк содержит информацию об участниках третьего этапа и включает два числа: код участника и сумму баллов, набранных на втором этапе соревнований. Строки файла должны быть отсортированы по убыванию суммы баллов, а при равенстве этого показателя — по возрастанию кода участника.

## Примеры

input.txt	output.txt
5 4 2 3 3 3 2 1 1 2 1 101 1 12 86 100 100 102 1 13.48 15.97 100 38 403 4 11 86 98 0 201 2 0 0 5 0 104 1 57 35 91 95 202 2 48 11.95 99.99 30 103 1 37 28 93 47 401 4 91 41 100 100 301 3 0 5 0 0 303 3 2 0 100 10 302 3 47 71 100 78 402 4 18 32 100 74 503 5 55.16 95 100 65.18 502 5 100 100 100 15.34 501 5 0 18 0 0	8 401 332.00 502 315.34 503 315.34 101 298.00 302 296.00 104 278.00 402 224.00 202 189.94
5 4 2 3 3 3 2 1 1 2 1 101 1 12 86 100 100 102 1 13.48 15.97 100 38 403 4 0 0 0 0 201 2 0 0 5 0 104 1 57 35 91 95 202 2 48 11.95 99.99 30 103 1 37 28 93 47 401 4 0 0 0 0 301 3 0 5 0 0 303 3 2 0 100 10 302 3 47 71 100 78 402 4 18 32 100 74 503 5 55.16 95 100 65.18 502 5 45 98 100 11.07 501 5 0 0 0 0	6 503 315.34 101 298.00 302 296.00 104 278.00 402 224.00 202 189.94

## Замечание

В первом примере квота района с номером 5 увеличивается на одного участника, а во втором — район с номером 4 лишается одного участника.

## Задача 2. Чекер для sudoku

Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Судoku — головоломка, получившая невероятную популярность, начиная с 1980-х — 1990-х годов. Игровое поле для этой головоломки представляет собой квадрат размером  $9 \times 9$ , разделённый на меньшие квадраты со стороной в 3 клетки. Таким образом, всё игровое поле состоит из 81 клетки. В них уже в начале игры стоят некоторые числа (от 1 до 9), называемые *подсказками*. От игрока требуется заполнить свободные клетки цифрами от 1 до 9 так, чтобы в каждой строке, в каждом столбце и в каждом малом квадрате  $3 \times 3$  каждая цифра встречалась бы только один раз.

4	1	3				2	5	
		2	3					
			4	9			8	
7	8	9			3			
5			9	6				1
			2			9	7	5
	4			3	7			
					5	4		
	5	1				3	6	7

4	1	3	7	6	8	2	5	9
8	9	2	3	5	1	7	4	6
6	7	5	4	9	2	1	8	3
7	8	9	5	1	3	6	2	4
5	2	4	9	7	6	8	3	1
1	3	6	2	8	4	9	7	5
9	4	8	6	3	7	5	1	2
3	6	7	1	2	5	4	9	8
2	5	1	8	4	9	3	6	7

На приведенном рисунке показан пример исходного заполнения игрового поля подсказками и решённая головоломка.

Правильно составленная головоломка имеет только одно решение. Тем не менее, иногда предлагаются варианты судoku с несколькими вариантами решения.

Соревнования по решению судoku предполагают, что каждый участник получает в качестве задания информацию о подсказках и сдаёт в качестве результата полностью заполненное цифрами игровое поле. Проверка правильности решения выполняется автоматически.

К сожалению, в случае наличия нескольких вариантов решения проверка путём сравнения с эталонным решением является недостаточной. Для проверки правильности необходимо написать специальную программу — *чекер*. Вам требуется написать такую программу...

Чекер должен находить следующие ошибки в решении головоломки:

- наличие в решении символов, не совпадающих с цифрами от 1 до 9;
- наличие одинаковых цифр в столбцах, строках, малых квадратах;
- наличие цифр, стоящих на месте подсказок и не совпадающих с подсказками.

### Формат входных данных

В первой строке задаётся величина  $Q$  ( $2 \leq Q \leq 10$ ) — количество решений одной головоломки, каждое из которых должен проверить чекер. Следующие девять строк входного файла содержит 9 символов каждая и содержат описание исходной головоломки. В этом описании допустимы символы '1', ..., '9', соответствующие цифрам-подсказкам, и символ '\*', означающий незаполненную клетку.

После этого следуют описания каждого решения (9 строк по 9 символов для одного решения). В описании решений могут встретиться только символы с кодами от 32 до 127.

### Формат выходных данных

Выведите строку из  $Q$  символов 'Y' или 'N'. Символ 'Y' означает, что очередное решение — правильное, в противном случае чекер обнаружил хотя бы одну ошибку.

## Примеры

input.txt	output.txt
3	YNN
413***25*	
**23*****	
***49**8*	
789**3***	
5**9*6**1	
***2**975	
*4**37***	
*****54**	
*51***367	
413768259	
892351746	
675492183	
789513624	
524976831	
136284975	
948637512	
367125498	
251849367	
213768254	
892351746	
675492183	
789513624	
524976831	
136284975	
948637512	
467125398	
251849367	
4137a8259	
89235b746	
6754921c3	
789*13624	
52*976831	
136254*75	
48637512	
3 7125498	
25 849367	

### Задача 3. Железнодорожная изгородь

Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 16 мегабайт

Метод «железнодорожной изгороди»<sup>1</sup> — один из простейших методов шифрования текстовых сообщений, разработанный в начале XX века. Конечно, сейчас говорить о его надёжности и стойкости смешно, но, тем не менее, в анналы криптографии он занесен...

Суть метода «железнодорожной изгороди» заключается в следующем: текст длиной  $N$  символов посимвольно записывается «змейкой» в таблицу из  $N$  столбцов и  $k$  строк (здесь  $k$  — ключ шифрования). Пример заполнения таблицы показан на рисунке для  $k = 5$  и исходной строки `This_is_a_secret_message`, для наглядности пробелы заменены символом нижнего подчёркивания.

T							a															
	h					_		_						t		m						e
		i				s				s				e				e				g
			s		i						e		r						s		a	
					_						c								s			

После заполнения таблицы символы исходного текста читаются из этой таблицы построчно. Таким образом, зашифрованная строка имеет вид `Ta_h_tmeisseegsiersa_cs`. Заметьте, что длина зашифрованного сообщения всегда совпадает с длиной исходного.

Вам требуется зашифровать сообщение методом «железнодорожной изгороди».

#### Формат входных данных

Первая строка входного файла содержит величину ключа  $k$  ( $2 \leq k \leq 1000$ ). Во второй строке записано шифруемое сообщение, длина которого не меньше 1 символа и не превосходит 30000 символов (в 50 % тестов эта длина не превосходит 255). В сообщение могут входить символы с кодами от 32 до 127, пробелов в начале и конце строки нет.

#### Формат выходных данных

Выведите единственную строку с зашифрованным сообщением.

#### Примеры

<code>input.txt</code>	<code>output.txt</code>
5 This is a secret message	Ta h tmeisseegsiersa cs
5 Cat	Cat

<sup>1</sup>По правде говоря, авторы задачи не понимают смысл этого термина :)

## Задача 4. Игра в 9999

Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 16 мегабайт

Антон и Борис увлечены новой игрой, суть которой заключается в следующем. Один из игроков, по жребию, становится *ведущим*. Ведущий придумывает четырёхзначное слово (возможно, с нулями в старших разрядах) и передаёт ход другому игроку. В дальнейшем игроки делают ходы по очереди.

Выполняя ход, игрок может увеличить одну из цифр текущего числа на 1, 2 или 3. Естественно, новое значение этой цифры не должно превосходить девяти. Игрок, получивший в результате хода число 9999, проигрывает.

Определите, выиграет ли ведущий игрок при заданном начальном числе, если в процессе игры ни Антон, ни Борис не будут совершать ошибок.

### Формат входных данных

В первой строке записано количество подтестов  $Q$  ( $3 \leq Q \leq 10$ ). В каждой из последующих строк записано одно число в диапазоне от 1 до 9998 — очередное придуманное ведущим игроком число. Незначащие нули в этих числах не записываются.

Для 10 % тестов во всех их подтестах десятичная запись придуманных чисел содержит три девятки.

### Формат выходных данных

Выведите строку из  $Q$  символов. В очередной позиции этой строки записывается символ 'Y', если ведущий игрок выигрывает, и 'N' — в противном случае.

### Примеры

<code>input.txt</code>	<code>output.txt</code>
5 9989 9995 9696 8999 8899	YNYYN
4 9989 999 9899 5999	YYYN