

**МЕЖДУНАРОДНЫЙ ТУРНИР ПО ИНФОРМАТИКЕ**  
**22 Ноября, 2014, Шумен, Болгария**  
**Старшая Лига**

**ЗАДАЧА A1. Муравьи**

Муравьиная страна расширяется. Как и любая другая порядочная страна, она состоит из городов. Как только в одном из старых городов происходит перенаселение, недовольные этим муравьи образуют новый город и переселяются в него, таким образом стабилизируя ситуацию в старом городе. Изначально в муравьиной стране был всего один город. Издревле положено, чтобы каждый муравьиный город состоял из ровно  $M$  муравейников, пронумерованных целыми числами от 1 до  $M$ . Также мы знаем изначальные размеры муравейников в первом городе. Муравьи возводят новые города вместе с муравейниками мгновенно. Так как муравьи чтят традиции, то каждый новый город изначально является полной копией их прошлого. То есть размер первого муравейника нового города будет совпадать с размером первого муравейника старого города, аналогично со всеми остальными. Но после возведения нового города приходят муравьи-инноваторы и увеличивают размеры жилищ с номерами с  $L$  до  $R$  включительно, на некоторое число  $V$ .

После этого престижным объявляется квартал, состоящий из муравейников с номерами от  $i$  до  $j$  включительно, причем муравей-мэр нового города хочет немедленно узнавать, сколько муравьев можно поселить в этот престижный квартал.

Так как муравьи не сильны в подсчетах, то они попросили вас помочь им с этой нелегкой задачей.

**Формат входных данных**

В первой строке заданы два целых числа  $N$  и  $M$  – число городов в муравьиной стране после возведения всех городов и число муравейников в каждом из городов соответственно.

Во второй строке содержатся  $M$  целых чисел  $A_1, A_2, A_3, \dots, A_M$  – размеры муравейников в первом городе.

Следующие  $N - 1$  строк содержат историю основания муравьиных городов, каждый новый город описывается пятеркой целых чисел  $P, X, Y, Z, T$ .  $P$  соответствует номеру города из которого получился новый. Каждый новый город получает в качестве номера минимальное натуральное число, которое еще не было использовано в качестве номера города. Числа  $L, R, i, j$ , описанные выше, получаются следующим образом:

$$L = ((X + S) \bmod M) + 1; R = ((Y + S) \bmod M) + 1$$

$$i = ((Z + S) \bmod M) + 1; j = ((T + S) \bmod M) + 1$$

Изначально  $S$  равно 0, а после построения нового города оно становится равно суммарной вместительности престижного квартала в этом городе.

Как видно из формулы, это число будет использовано при генерации следующего запроса. Гарантируется, что верно вычисленные числа будут удовлетворять ограничениям:  $L \leq R$  и  $i \leq j$ .

**Формат выходных данных**

Для каждого из новых городов выводите по строчке, содержащей ровно одно целое число – вместимость престижного квартала.

**Ограничения**

$$1 \leq N, M \leq 100000$$

$$0 \leq A_i, V \leq 100000 \text{ для } 1 \leq i \leq M.$$

$$1 \leq L \leq R \leq M.$$

$$1 \leq i \leq j \leq M.$$

$$0 \leq X, Y, Z, T < M$$

**МЕЖДУНАРОДНЫЙ ТУРНИР ПО ИНФОРМАТИКЕ**  
**22 Ноября, 2014, Шумен, Болгария**  
**Старшая Лига**

**Пример**

<b>Ввод</b>	<b>Вывод</b>
4 4	
3 6 7 5	9
1 2 3 1 0 1	12
2 1 2 6 2 2	
1 0 2 8 0 3	45

***Пояснение к примеру***

В первом городе размеры зданий равны {3, 6, 7, 5}. При создании второго мы получаем  $S = 0$ ,  $L = 3$ ,  $R = 4$ ,  $V = 1$ ,  $i = 1$ ,  $j = 2$ . Таким образом второй город выглядит так: {3, 6, 8, 6}, а размер престижного квартала равен  $3 + 6 = 9$ . После этого шага  $S = 9$ .

Для третьего города вычисляем  $L = ((1 + 9) \bmod 4) + 1 = 3$ ,  $R = 4$ ,  
 $i = ((2 + 9) \bmod 4) + 1 = 4$ ,  $j = 4$ ;  $V = 6$ . Размеры муравейников третьего города будут равны {3, 6, 14, 12}. Размер престижного квартала 12, поэтому теперь  $S = 12$ .

Четвертый город получился из первого с помощью  $S = 12$ , мы вычисляем  $L = 1$ ,  $R = 3$ ,  $i = 1$ ,  $j = 4$ .  $V = 8$ . Для него размеры равны {11, 14, 15, 5}, а престижный квартал получился размера 45.

**Система оценки**

Вы получите не менее 10% очков, если ваша программа будет работать для ограничений  $N, M \leq 1000$ .

Вы получите не менее 30% очков, если ваша программа будет работать для случая, когда каждый следующий город будет образован от города, основанного на прошлом шаге (для первого запроса таким будет первый город).

**МЕЖДУНАРОДНЫЙ ТУРНИР ПО ИНФОРМАТИКЕ**  
**22 Ноября, 2014, Шумен, Болгария**  
**Старшая Лига**

**ЗАДАЧА А2. АВ-СЛОВА**

Назовем АВ-словом строку, состоящую из букв А или В, длины хотя бы два, начинающуюся с буквы А. Также определим два преобразования для таких строк:

- **R<sub>1</sub>**: Все буквы кроме последней остаются без изменений, а последняя меняется на противоположную: А становится В, а В становится А.
- **R<sub>2</sub>**: Обозначим исходное слово за  $w$ . Тогда мы строим строку  $t$  из  $w$  следующим образом:
  - Первая буква  $t$  – это конечно же А.
  - Каждая следующая буква на позиции  $i$  ( $i > 1$ )  $t_i$  получается так: если  $w_{i-1} = w_i$ , то  $t_i = В$ , иначе  $t_i = А$ .
  - Результатом нашей операции будет строка  $t$ .

Назовем последовательность из  $N$  последовательных применений операций **R<sub>1</sub>** и **R<sub>2</sub>** к исходной строке  $w$  корректной, если:

- В конце мы получим тоже самое слово, что было у нас в начале ( $w$ );
- Все слова, полученные в процессе преобразований попарно различны.

Вам дано одно положительное число  $N$ . Вам требуется написать программу, которая находит кратчайшее АВ-слово для которого существует корректная последовательность из  $N$  операций, или установите, что такой строки не существует.

**Формат входных данных**

- Единственная строка содержит число  $2 \leq N \leq 100\,000$ .

**Формат выходных данных**

Ваша программа должна выводить:

- На первой строке кратчайшее АВ-слово;
- Вторая строка должна состоять из  $N$  символов 1 или 2 без разделителей, описывающих корректную последовательность преобразований.

Или

- Единственное слово NO, если ответа не существует.

**Ограничения**

- В 20% тестов  $N$  не превосходит 30;
- В 40% тестов  $N$  не превосходит 200;
- В 70% тестов  $N$  не превосходит 10 000;

**Пример**

Ввод	Вывод
6	AABV 221212

**Пояснение к примеру**

AABV  $\xrightarrow{\textcircled{2}}$  AVAV  $\xrightarrow{\textcircled{2}}$  AAAA  $\xrightarrow{\textcircled{1}}$  AAAB  $\xrightarrow{\textcircled{2}}$  AVVA  $\xrightarrow{\textcircled{1}}$  AVVV  
 $\xrightarrow{\textcircled{2}}$  AABV

**Система оценки**

- Если верно определено несуществование ответа, то вы получите все баллы за этот тест; Иначе
- Если первая строка не содержит АВ-слова, или выданная вами последовательность преобразований не является корректной, то вы не получите баллов за этот тест; Иначе
- Вы получите такой процент баллов от этого теста, на сколько ваше решение близко к оптимальному.

**МЕЖДУНАРОДНЫЙ ТУРНИР ПО ИНФОРМАТИКЕ**  
**22 Ноября, 2014, Шумен, Болгария**  
**Старшая Лига**

**ЗАДАЧА А3. Окно**

Квадратная стена размером  $N \times N$  выложена из  $N^2 / 2$  кирпичей, плотно приложенных друг к другу ( $N$  четно). Каждый кирпич размером  $2 \times 1$ . Кирпичи пронумерованы целым числами от 1 до  $N^2 / 2$ . Некоторые кирпичи расположены горизонтально, а некоторые вертикально. В стене нет дыр. На рисунке ниже показан пример такой стены при  $N = 6$ , квадраты с одинаковыми номерами соответствуют кирпичу с таким же номером.

1	1	4	4	13	14
2	3	3	5	13	14
2	6	7	5	12	12
9	6	7	10	10	15
9	8	8	11	11	15
16	16	17	17	18	18

В стене-то дыр нет, но мы очень хотим сделать одну под окно, причем удовлетворяющую следующим условиям:

- 1) Дыра должна быть квадратной.
- 2) Ее стороны должны быть параллельны сторонам стены.
- 3) Она не должна касаться сторон стены, то есть лежать строго внутри.
- 4) Мы хотим, чтобы при создании дыры ни один кирпич не пострадал, то есть стороны дыры должны проходить только по сторонам кирпичей.

Вам требуется написать программу, которая находит максимальный размер дыры, которую мы можем сделать при данных ограничениях, ведь чем больше дыра – тем больше размер окна, а маленькие окна никому не нужны. Гарантируется, что всегда можно сделать хотя бы одно окно.

**Формат входных данных**

Первая строка содержит единственное целое число  $N$  – ширину стены. Следующие  $N$  строк содержат описание стены в таком же формате, как описано выше.

**Формат выходных данных**

В единственной строке выведите три числа – ширину максимальной дыры и координаты ее левого верхнего угла. Если правильных ответов несколько, то разрешается выводить любой из них.

**Пример** (соответствует картинке выше)

Ввод	Вывод
6 1 1 4 4 13 14 2 3 3 5 13 14 2 6 7 5 12 12 9 6 7 10 10 15 9 8 8 11 11 15 16 16 17 17 18 18	2 3 2

**Пояснение к примеру:** Существует два ответа: либо вырезать кирпичи с номерами 6 и 7, либо с номерами 10 и 11.

**Система оценки**

Подзадача 1 (15 баллов):  $4 \leq N \leq 100$ .

Подзадача 2 (35 баллов):  $100 < N \leq 1000$ .

Подзадача 3 (50 баллов):  $1000 < N \leq 2000$ .

*Вы получите баллы за подзадачу только в том случае, если вы пройдете все соответствующие ей тесты.*