

# Морской бой

## Тип 1, задача 1

Все знают и любят замечательную игру «Морской бой». В этой игре принимают участие два игрока, которые по очереди называют координаты на неизвестной им карте соперника. Если у соперника по этим координатам располагается корабль, то корабль или его часть топится, а игрок получает право сделать ещё один ход. Цель игры – первым потопить все корабли соперника.

Чтобы карта игрока была корректной, для неё должны выполняться следующие условия:

- карта представляет собой квадрат размера  $10 \times 10$ ;
- горизонтали пронумерованы сверху вниз числами от 1 до 10;
- вертикали помечаются слева направо буквами латинского алфавита от «а» до «к» (пропускается буква «j»);
- на карте размещаются следующие корабли:
  - не более одного четырёхпалубного корабля – ряд или столбец из 4 клеток;
  - не более двух трёхпалубных кораблей – ряд или столбец из 3 клеток;
  - не более трех двухпалубных кораблей – ряд или столбец из 2 клеток;
  - не более четырех однопалубных кораблей из одной клетки;
- на карте должен быть расположен, по крайней мере, один корабль;
- никакие два корабля не должны касаться друг друга по углу или по стороне.

Ваш друг решил написать компьютерную версию игры и просит Вас помочь написать ему модуль, который по известной карте соперника и сделанным ходам будет выдавать один из ответов:

- «MISS» (без кавычек), если удар был нанесён по пустой клетке или клетке, по которой уже был нанесён удар ранее;
- «HIT» (без кавычек), если удар потопил часть корабля, но не потопил весь корабль;
- «DEAD» (без кавычек), если был нанесён удар по однопалубному кораблю или последней ещё не потопленной части многопалубного корабля, и после этого удара остался хотя бы один ещё не потопленный корабль на карте;
- «GAME OVER» (без кавычек), если был нанесён удар по однопалубному кораблю или последней ещё не потопленной части многопалубного корабля, и после этого удара были потоплены все корабли на карте;
- если уже было выведено сообщение «GAME OVER», то все последующие ходы не обрабатываются.



### Входные данные

Первые 10 строк входного файла содержат по 10 символов, задающих корректную карту, каждый из которых равен ' .' (ASCII 46), если координата не содержит палубы корабля, и '# ' (ASCII 35), если в координате находится палуба одного из кораблей.

11-ая строка содержит единственное натуральное число  $N$  ( $1 \leq N \leq 100$ ) – количество ударов, которые необходимо обработать.

Следующие **N** строк содержат число от 1 до 10 и букву латинского алфавита от «a» до «k» (кроме буквы «j»), разделённые одиночным пробелом, – координаты на карте, по которым был совершён удар.

Гарантируется, что на игровом поле размещен хотя бы один корабль.

### Выходные данные

Выходной файл должен содержать не более **N** строк – ответы модуля на совершённые удары.

| <i>input.txt</i> | <i>output.txt</i> | Пояснение                                  |
|------------------|-------------------|--|
| .....            | HIT               | Первым ударом игрок топил часть            |
| .###.....        | MISS              | трёхпалубного корабля.                     |
| .....            | DEAD              | Вторым ударом попадает в пустую клетку.    |
| .....            | HIT               | Третьим ударом игрок топил однопалубный    |
| .....            | GAME OVER         | корабль.                                   |
| .....            |                   | Четвёртым и пятым ударами топим оставшиеся |
| .....            |                   | две палубы трёхпалубного корабля.          |
| .....            |                   | После пятого удара игра заканчивается, и   |
| .....            |                   | шестой удар обрабатывать не следует.       |
| .....#           |                   |  |
| 6                |                   |  |
| 2 b              |                   |  |
| 1 a              |                   |  |
| 10 k             |                   |  |
| 2 d              |                   |  |
| 2 c              |                   |  |
| 10 k             |                   |  |

## Технологии

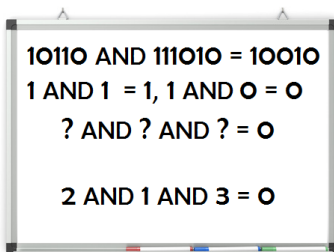
### Тип 1, задача 2

Благодаря Байтландской Организации по Исследованиям (БОИ), Байтландия достигла высочайшего уровня технологического развития. В настоящее время БОИ разрабатывает секретный проект, который (планируется) «перевернёт» к лучшему жизнь Байтландии и всего мира. Учёные из БОИ не раскрывают сути проекта, но известно, что в процессе его реализации возникла проблема, и именно вам, как главному программисту БОИ, поручили её решить.

Необходимо по двум целым числам **K** и **Y** найти такую последовательность **X** из **K** различных целых чисел, которая будет удовлетворять следующему соотношению, и при этом каждое из чисел должно состоять не более чем из 30 двоичных битов:

$$X_1 \text{ AND } X_2 \text{ AND } \dots \text{ AND } X_K = Y.$$

**Побитовое И (AND)** – это бинарная операция, действие которой эквивалентно применению логического **И** к каждой паре битов, которые стоят на одинаковых позициях в двоичных представлениях операндов. Другими словами, если оба соответствующих бита операндов равны 1, то результирующий



двоичный разряд равен 1. Если хотя бы один бит из пары равен 0, то результирующий двоичный разряд будет равен 0.

Например,  $1010_2 \text{ AND } 1100_2 = 1000_2$ , так как пары битов, соответствующих первому, второму и третьему биту операндов, содержат хотя бы один 0, поэтому результирующий бит равен 0, а четвертый бит в обоих операндах равен 1 – результирующий бит равен 1.

Напишите программу, которая формирует последовательность, удовлетворяющую заданному выше требуемому соотношению, либо сообщит, что такой последовательности не существует.

### Входные данные

Единственная строка входного файла содержит два целых числа  $K$  и  $Y$  ( $1 \leq K \leq 10^5$ ,  $0 \leq Y < 2^{30}$ ). Числа во входном файле разделены одиночными пробелами.

### Выходные данные

Единственная строка выходного файла должна содержать любую удовлетворяющую требуемому соотношению последовательность из  $K$  различных целых чисел –  $X_1 X_2 \dots X_K$  ( $0 \leq X_i < 2^{30}$ ). Если последовательности, удовлетворяющей условию задачи, не существует, то выведите одно число  $-1$ .

Числа в выходном файле следует разделять одиночными пробелами.

| <i>input.txt</i> | <i>output.txt</i> | Пояснение   |
|------------------|-------------------|---|
| 3 0              | 1 3 2             | $1_2 \text{ AND } 11_2 \text{ AND } 10_2 = (1_2 \text{ AND } 11_2) \text{ AND } 10_2 = 1_2 \text{ AND } 10_2 = 0$ .   |
| 2 1073741823     | -1                | $1073741823_{10} = 2^{30} - 1 = 11\dots111_2$ (тридцать единичных битов). Так как хотя бы одно число $X_1$ или $X_2$ будет содержать нулевой бит (среди первых тридцати), то требуемый результат не может быть получен. |

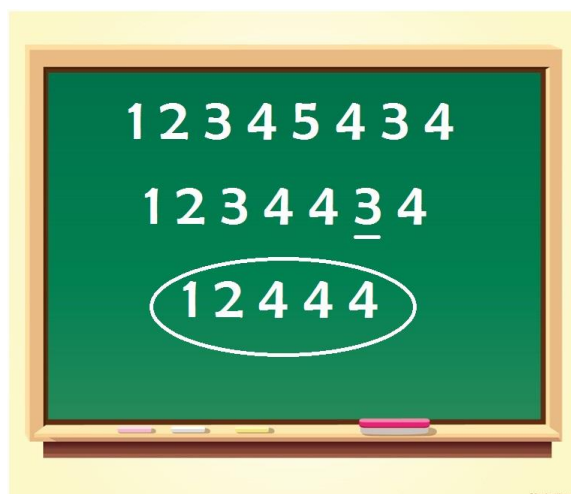
## Интересная задача

### Тип 1, задача 3

В этом году в Байтландской гимназии на факультативных занятиях дети учатся решать разные задачи на числовые последовательности. С приближением середины учебного года задачи становятся всё сложнее и интереснее.

На последнем занятии учитель предложил ребятам решить одну увлекательнейшую задачу. В задаче задаётся последовательность  $A$  из  $N$  целых чисел  $A_i$ . К последовательности можно применять единственное преобразование: выбрать любое число, входящее в эту последовательность, и удалить все его вхождения. В результате этого преобразования получается новая последовательность меньшей длины, к которой можно снова применять описанное выше преобразование. Например, если к последовательности, включающей числа 1, 2, 3, 4, 5, 4, 3, 4, применить это преобразование, выбрав для удаления число 4, то новая последовательность будет содержать числа 1, 2, 3, 5, 3.

Таковыми преобразованиями можно получить много различных последовательностей, но учителя интересуют



только интересные последовательности. Последовательность **B** длины **M** называется *интересной*, если не найдётся такой тройки индексов  $1 \leq i < j < k \leq M$ , что  $B_i = B_k$  и  $B_i \neq B_j$ . Из последовательности **A** необходимо получить *интересную* последовательность **B** максимальной длины. Например, последовательность, содержащая числа 1, 2, 3, 4, 5, 4, 3, 4, не является *интересной*, так как не удовлетворяет требуемому условию, т.е. третий и седьмой элементы последовательности равны, а третий и четвёртый элементы не равны. Одним из способов получения *интересной* последовательности (в данном примере) является применение двух преобразований: одно удаляет число 5, другое – число 3. В результате будет получена интересная последовательность 1, 2, 4, 4, 4.

Напишите программу, которая из заданной последовательности получает интересную последовательность максимальной длины.

### Входные данные

Первая строка входного файла содержит единственное натуральное число **N** ( $1 \leq N \leq 10^5$ ) – длину заданной последовательности **A**.

Вторая строка входного файла содержит **N** целых чисел  $A_i$  ( $1 \leq A_i \leq 100$ ) – элементы последовательности. Числа в строке разделены одиночными пробелами.

### Выходные данные

Единственная строка выходного файла должна содержать единственное целое число – максимальную длину интересной последовательности.

| <i>input.txt</i>     | <i>output.txt</i> | Пояснение  |
|----------------------|-------------------|--|
| 3<br>1 1 1           | 3                 | Исходная последовательность <b>A</b> является интересной, поэтому для получения интересной последовательности максимальной длины не требуется удалять элементы.  |
| 8<br>1 2 3 4 5 4 3 4 | 5                 | Из исходной последовательности можно получить 16 различных интересных последовательностей:<br>(пустая последовательность), (1), (2), (3, 3), (4, 4, 4), (5), (1, 2), (1, 3, 3), (1, 4, 4, 4), (1, 5), (2, 3, 3), (2, 4, 4, 4), (2, 5), (1, 2, 3, 3), (1, 2, 4, 4, 4), (1, 2, 5).<br>Для получения интересной последовательности максимальной длины первым преобразованием можно удалить все элементы равные 3, а вторым – равные 5. Итоговая последовательность будет (1, 2, 4, 4, 4). |

## Диск

### Тур 1, задача 4

В Байтландии каждый 1024-й День Святого Байта объявляется Днём Святого Килобайта. Как и в любой государственный праздник, в этот день проводятся массовые гуляния, дарятся подарки и, конечно, решаются головоломки.

Неудивительно, что в Британии наиболее популярной детской головоломкой является Диск Инвариантности Святого Килобайта (или сокращённо ДИСК). ДИСК представляет собой  $N$  концентрических колец. Каждое кольцо разделено на  $K$  равных секторов, которые пронумерованы целыми числами от 1 до  $K$  против часовой стрелки. Дополнительно каждый сектор с номером  $i$  любого кольца совмещён по углу с сектором с номером  $i$  любого другого кольца. Каждый сектор колец в головоломке покрашен в один из  $K$  цветов. На Рисунке 1 изображён пример ДИСКА для  $N=3$ ,  $K=8$ .

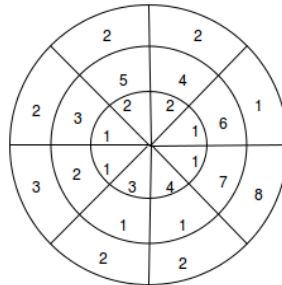


Рисунок 1. Пример ДИСКА. Числами обозначены цвета секторов

Головоломка считается решённой, если один из секторов ДИСКА оказывается полностью закрасенным в один цвет, т.е. существует такое число  $i$ , что для всех колец сектора с номером  $i$  имеют один цвет.

Для достижения этой цели игроку разрешается поворачивать любое кольцо по часовой либо против часовой стрелки ровно на  $360/K$  градусов, т.е. на один сектор. Длительность каждого поворота кольца составляет ровно одну секунду. На Рисунках 2 и 3 показаны изображения ДИСКов, полученных вращением среднего кольца исходного диска (Рисунок 1) в течение ровно одной секунды по часовой стрелке и против часовой стрелки соответственно.

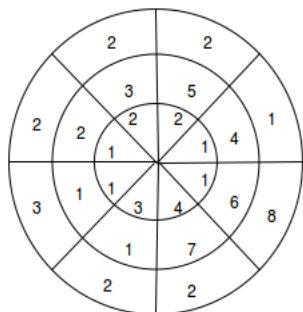


Рисунок 2. Поворот среднего кольца по часовой стрелке

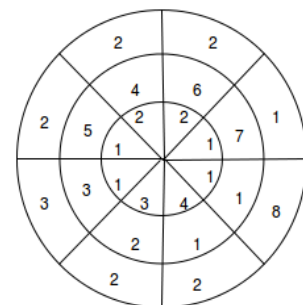


Рисунок 3. Поворот среднего кольца против часовой стрелки

Например, для ДИСКА с Рисунка 1 достаточно повернуть среднее кольцо против часовой стрелки два раза, получив, таким образом, сектор, полностью закрашенный в цвет с номером 1 (Рисунок 4).

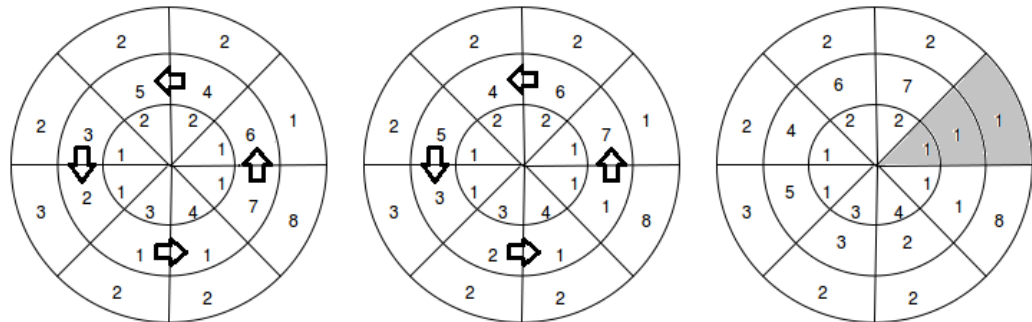


Рисунок 4. Решение головоломки ДИСК, изображенной на Рисунке 1

Составьте программу, которая за минимальное время решит головоломку.

### Входные данные

Первая строка входного файла содержит два натуральных числа  $N$  и  $K$  ( $2 \leq N \leq 2000$ ,  $3 \leq K \leq 2000$ ) – количество колец и секторов соответственно.

Следующие  $N$  строк содержат описания колец в порядке возрастания их радиусов. Каждое из описаний состоит из  $K$  натуральных чисел:  $i$ -е число в строке задает цвет  $i$ -го сектора соответствующего кольца.

Цвета закодированы натуральными числами от 1 до  $K$ .

Числа в строках входного файла разделены одиночными пробелами.

### Выходные данные

В выходной файл выведите одно целое число – минимальное время в секундах, которое требуется для решения головоломки. Если головоломка решения не имеет, то выведите число  $-1$ .

| <i>input.txt</i>   | <i>output.txt</i> | Пояснение  |
|--|-------------------|--|
| 3 8<br>1 2 2 1 1 3 4 1<br>6 4 5 3 2 1 1 7<br>1 2 2 2 3 2 2 8 | 2                 | Пример из условия задачи.  |
| 2 4<br>1 1 1 1<br>2 2 2 2                                    | -1                | Головоломку решить невозможно. Кольца полностью закрашены в различные цвета.     |
| 2 3<br>1 1 2<br>2 1 3  | 0                 | Головоломка уже решена. Второй сектор у обоих колец покрашен в цвет с номером 1. |

## Обзорный лист

### I тур

| Название задачи                               |               | Морской бой  | Технологии                                   | Интересная задача                                | Диск   |
|---|---------------|--|--|--|--|
| <b>Параметры</b>                              |               |  |  |  |  |
| <b>Входной файл</b>                           |               | input.txt  | input.txt                                    | input.txt  | input.txt  |
| <b>Выходной файл</b>                          |               | output.txt   | output.txt                                   | output.txt                                       | output.txt   |
| <b>Время на тест</b>                          |               | 1 сек  | 1 сек  | 1 сек  | 5 сек  |
| <b>Ограничение памяти</b>                     |               | 128 МВ   | 128 МВ                                       | 128 МВ   | 256 МВ   |
| <b>Ограничение на размер сдаваемых файлов</b> |               | 64 Кбайт   | 64 Кбайт                                     | 64 Кбайт   | 64 Кбайт   |
| <b>Опции компилятора</b>                      | <b>C++</b>    | -O2  | -O2  | -O2  | -O2  |
|   | <b>Pascal</b> | -O2  | -O2  | -O2  | -O2  |
| <b>Количество тестов</b>                      |               | 20   | 20   | 20   | 20   |
| <b>Максимум баллов</b>                        |               | 100  | 100  | 100  | 100  |
| <b>Сдать на проверку</b>                      | <b>C++</b>    | Исходный файл с расширением <b>cpp</b>                                   | Исходный файл с расширением <b>cpp</b>       | Исходный файл с расширением <b>cpp</b>           | Исходный файл с расширением <b>cpp</b>   |
|   | <b>Pascal</b> | Исходный файл с расширением <b>pas</b>                                   | Исходный файл с расширением <b>pas</b>       | Исходный файл с расширением <b>pas</b>           | Исходный файл с расширением <b>pas</b>   |
| <b>Частичная оценка</b>                       |               | На поле расположен только один однопалубный корабль - не менее 20 баллов | $K \leq 3, Y \leq 1$ - не менее 20 баллов    | $A_i \leq 2, N \leq 100$ - не менее 20 баллов    | $N \leq 2, K \leq 100$ , все сектора каждого из колец имеют различные цвета - не менее 20 баллов   |
|   |               | На поле расположен только один корабль - не менее 40 баллов              | $K \leq 3, Y \leq 1000$ - не менее 30 баллов | $A_i \leq 6, N \leq 1000$ - не менее 40 баллов   | $N \leq 100, K \leq 100$ , все сектора каждого из колец имеют различные цвета - не менее 40 баллов |
|   |               |  | $K, Y \leq 1000$ - не менее 60 баллов        | $A_i \leq 10, N \leq 10000$ - не менее 60 баллов | $N \leq 300, K \leq 300$ - не менее 60 баллов  |
|   |               |  |  | $A_i \leq 20$ - не менее 80 баллов               | $N \leq 1000, K \leq 1000$ - не менее 80 баллов  |

Тестирование решений будет производиться на компьютере с тактовой частотой процессора не менее 2 GHz и объемом оперативной памяти не менее 1024 MB.