

Рисунки по числам

"Рисунки по числам" - хорошо известная игра-головоломка. Рассмотрим простой одномерный вариант этой головоломки. В этой головоломке, игрок получает ряд n ячеек. Ячейки пронумерованы от 0 до $n - 1$ слева направо. Игрок должен закрашивать каждую ячейку черным или белым цветом. Мы используем 'X', чтобы обозначить черные ячейки и '_', чтобы обозначить белые.

Игрок получает последовательность $c = [c_0, \dots, c_{k-1}]$ из k целых положительных чисел: ключей к разгадке. Он должен закрасить ячейки так, чтобы черные ячейки в ряду создали точно k блоков последовательно расположенных ячеек. Кроме того, количество черных ячеек в i -ом блоке, начиная слева (нумерация с 0) должно быть равно c_i . Например, если ключами к разгадке являются $c = [3, 4]$, то решаемая головоломка должна иметь ровно два блока последовательных черных ячеек: один длиной 3, а затем другой - длиной 4. Следовательно, если $n = 10$ и $c = [3, 4]$, то единственным верным решением является "_XXX_XXXX". Обратите внимание, на то что "XXXX_XXX_" не является допустимым решением: блоки черных ячеек не находятся в правильном порядке. Кроме того, "_XXXXXXXX_" не является допустимым решением: имеется один единственный блок черных ячеек, а не два отдельных блока.

Вам дается частично решенная головоломка "Рисунки по числам". То есть, вы знаете, n and c , и, вы дополнительно знаете, что некоторые ячейки должны быть черного цвета, а некоторые клетки должны быть белого цвета. Ваша задача состоит в том, чтобы вывести дополнительную информацию. В частности, вы должны найти ячейки, которые окрашены в черный цвет в каждом верном решении, и ячейки, которые окрашены в белый цвет в каждом верном решении. Предполагается, что задаются такие исходные данные, что есть, по крайней мере, одно верное решение.

Детали реализации

Вы должны написать следующую функцию (метод):

- `string solve_puzzle(string s, int[] c)`.
 - s : string of length n . Для каждого i ($0 \leq i \leq n - 1$) i -ый символ в выходной строке должен быть равен:
 - 'X', если i -ая ячейка черная,
 - '_', если i -ая ячейка белая,
 - '.', если нет никакой информации об i -ой ячейке.
 - c : массив длиной k , содержащий ключи, как описано выше,
 - функция должна возвращать строку длиной n . Для каждого i (

$0 \leq i \leq n - 1$) i -ый символ в выходной строке должен быть:

- 'X', если i - ая ячейка является черной в каждом верном решении,
- '_', если i - ая ячейка является белой в каждом верном решении,,
- '?', во всех других случаях (т.е., если существуют два правильных решения таких, что i - ая ячейка в одном из них черная, в другом - белая).

In the C language the function signature is a bit different:

- `void solve_puzzle(int n, char* s, int k, int* c, char* result)`
 - n : длина строки s (число ячеек),
 - k : длина массива c (количество ключей),
 - другие параметры - такие же, как описаны выше,
 - вместо возвращаемой строки с n символами, функция должна записать ответ в строку `result`.

Примеры

Пример 1

```
solve_puzzle(".....", [3, 4])
```

Ниже приведены все правильные решения головоломки:

- "XXX_XXXX_",
- "XXX__XXXX_",
- "XXX___XXXX",
- "_XXX_XXXX_",
- "_XXX__XXXX",
- "___XXX_XXXX".

Можно увидеть, что ячейки (нумерация с 0) с индексами 2, 6, и 7 - черные в каждом правильном решении. Каждая из других ячеек может быть, а может и не быть, черной. Таким образом, правильный ответ - "??X???XX??".

Пример 2

```
solve_puzzle(".....", [3, 4])
```

В этом примере полное решение однозначно определяется и правильный ответ - "XXX_XXXX".

Пример 3

```
solve_puzzle("..._. ....", [3])
```

В этом примере мы можем сделать вывод, что ячейка 4 должна быть белой, а также - нет никакого способа, чтобы закрасить три последовательные черные ячейки между белыми ячейками с индексами 3 и 5. Таким образом, правильный ответ - "?? ___ ???".

Пример 4

```
solve_puzzle(".X.....", [3])
```

Имеется только два правильных решения, которые подходят выше приведенному описанию:

- "XXX _____",
- " _XXX _____".

Итак, правильный ответ - "?XX? _____".

Подзадачи

Во всех подзадачах $1 \leq k \leq n$, и $1 \leq c_i \leq n$ для каждого $0 \leq i \leq k - 1$.

1. (7 points) $n \leq 20$, $k = 1$, s contains only содержит только '.' (empty puzzle),
2. (3 points) $n \leq 20$, s содержит только '.',
3. (22 points) $n \leq 100$, s содержит только '.',
4. (27 points) $n \leq 100$, s содержит только '.' и '_' (информация только о белых ячейках),
5. (21 points) $n \leq 100$,
6. (10 points) $n \leq 5\,000$, $k \leq 100$,
7. (10 points) $n \leq 200\,000$, $k \leq 100$.

Sample grader

Пример проверяющего модуля

Проверяющий модуль получает данные в следующем формате:

- строка 1: строка s ,
- строка 2: целое k и k -целых чисел c_0, \dots, c_{k-1} .