



Задача Пермутация

 0.69 sec.  256 MB

За някои от вас това може да бъде първата Ви encode-decode задача. Сигурни сме, че ще ви хареса 😊

Дадена е пермутация* на числата от 1 до N . Вие се стремите да сложите възможно най-много -1 в нея. Подробното описание е по-долу.

*Пермутация - каквато и да е редица съставена от N елемента, в която всяко число от 1 до N се среща точно един път.



Детайли по имплементацията

Задачата Ви е да имплементирате две функции:

```
std::vector<int> encode(std::vector<int> a);
```

Тя ще бъде извикана веднъж от журито и трябва да кодира пермутацията a . Вие я кодирате, като замествате числа от пермутацията с -1 . След като сте преценили кои числа да заместите, връщате променената пермутация.

```
std::vector<int> decode(std::vector<int> a);
```

Тази функция ще бъде извикана веднъж от журито и трябва да декодира пермутацията, която сте кодирали в предишната функция (тоест като аргумент получавате Вашата пермутация със сложените от Вас -1). Вие трябва да я декодирате, като замените с -1 числото, което е най-малкото неизползвано до момента. Целта е след тези промени възстановената пермутация да е същата като тази, която е подадена в `encode`.

Вашият код **не може** да съдържа глобална памет! Това означава, че двете функции не могат да комуникират и да съхраняват данни, които другата може да достъпи.



Ограничения

- $1 \leq N \leq 10^5$



Оценяване

Нека броя -1, които сте поставили при encode е g , а оптималния брой е x . Ако пермутацията, която връща decode, е различна от оригиналната, получавате 0 точки на съответния тест. В противен случай, точките се смятат по следната формула:

$$points = \min(1, \frac{g}{x})$$



Подзадачи

Подзадача	Точки	Допълнителни ограничения
1	9	$a_i = i + 1, 0 \leq i \leq N - 1$
2	11	$1 \leq N \leq 16$ или 17
3	13	В редицата с оптимално поставени -1, то се среща само веднъж.
4	67	Няма ограничения.



Пример

Жури	Състезател
$encode(4, 1, 2, 3, 5)$	$4, -1, 2, 3, 5$
$decode(4, -1, 2, 3, 5)$	$4, 1, 2, 3, 5$



Примерен грейдър

На първия ред се въвежда размера на редицата N , а на следващия ред-самата редица. Грейдърът извежда дали редицата съвпада при двете извиквания и колко -1 са сложени.