

ПЪРВО КОНТРОЛНО СЪСТЕЗАНИЕ НА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР

Група G, 25 април 2021 г.

Задача G3. Острови

Журито си е намислило скрита пермутация на числата от 0 до $N - 1$. Вие трябва да я познаете използвайки възможно най-малко заявки. Пермутацията π можем да третираме като функция, която за число m между 0 и $N - 1$ връща друго (или същото) такова число, а именно числото на позиция m в пермутацията π . Една заявки се състои от това да разделите числата от 0 до $N - 1$ в K непразни множества (където Вие избирате K , което наричаме размера на заявката). Ще ги отбелязваме с S_1, S_2, \dots, S_K . Всяко число от 0 до $N - 1$ трябва да се среща в точно едно от тези множества. След това за всяко S_i журито прилага π (като функция) на всички негови елементи и образува S'_i . По-формално, ако $S_i = \{a_i^1, a_i^2, \dots, a_i^{|S_i|}\}$, то $S'_i = \{\pi(a_i^1), \pi(a_i^2), \dots, \pi(a_i^{|S_i|})\}$. Накрая журито разбърква елементите на тези S'_i -ове, както и самата редица от S'_i -ове, и Ви връща този разбъркан резултат. Например, нека разгледаме $N = 5$ и $\pi = 3, 1, 0, 4, 2$. Сега, ако Вие направите заявката $\{\{0, 3\}, \{4, 1, 2\}\}$, един възможен нейн отговор е $\{\{1, 2, 0\}, \{3, 4\}\}$.

Напишете програма `islands.cpp`, която правилно да намира скритата пермутация на журито използвайки възможно най-малко и заявки с потенциално ограничение за техните размер (тяхното K).

Забележка: Цялата задача се развива на някакви острови, но не остана време за фабулата.

Детайли по имплементацията

Вашата функция `solve` трябва да има следния прототип:

```
std::vector<int> solve(int N);
```

Тя ще бъде извикана точно веднъж и ще получи като аргумент дължината на пермутацията π . Функцията трябва да върне вектор с дължина N , който да съдържа π .

Функция `ask` на журито има следния прототип:

```
std::vector<std::vector<int>>  
ask(const std::vector<std::vector<int>> &partition);
```

Вашата програма може да я вика колкото си пъти иска. Като аргумент ѝ се дава разделение от описания вид. Елементите на вектора `partition` представляват по едно множество S_i , а елементите на един от тези представляват елементите на S_i . Редът на нито едно от двете не е от значение. Резултатът на функцията са данни в същия формат, носещи отговора на заявката, както е описано по-горе. Пак обръщаме внимание, че елементите на всеки вектор биват произволно разбъркани.

Вашата програма трябва да имплементира функцията `solve`, но не трябва да съдържа функция `main`. Освен това, тя не трябва да чете от стандартния вход или да печата на стандартния изход. Програмата Ви също така трябва да включва хедър файла `islands.h` чрез указание към препроцесора:

```
#include "islands.h"
```

Стига да спазва тези условия, програмата Ви може да съдържа каквито и да е помощни функции, променливи, константи и прочие.

ПЪРВО КОНТРОЛНО СЪСТЕЗАНИЕ НА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР

Група G, 25 април 2021 г.

Ограничения

$$3 \leq N \leq 10^4$$

Максимален брой заявки, които може да направите: 2×10^4

Максимален размер на заявка: *SizeLimit* (зависи от подзадачата)

Локално тестване

Предоставени са Ви файловете `islands.h` и `Lgrader.cpp`, които можете да компилирате заедно с Вашата програма, за да я тествате. При стартиране на програмата трябва първо да се въведе N . На следващия ред трябва се въведат N различни числа между 0 и $N - 1$, елементите на пермутацията. След това програмата ще изведе отговора на Вашата `solve` функция, както и броя използвани заявки и максималния размер на направена заявка.

Подзадачи и оценяване

Точките, които решението Ви ще получи на дадена подзадача, се определят от най-лошия му резултат на който и да е тест от подзадачата. Ако на даден тест функцията Ви направи невалидна заявка, превиши лимита за брой заявки или върне грешна пермутация, резултатът Ви за теста е 0. В противен случай, резултатът Ви за теста (от 0 до 1) зависи от броя направени заявки Q по-следната формула (*Target* зависи от подзадачата):

$$0.6^{\max(Q - \text{Target}, 0)}$$

Подзадачите са както следва:

Подзадача	Точки	N	<i>SizeLimit</i>	<i>Target</i>
1	11	= 15	N	15
2	15	= 4997	N	$Opt(N) + 15$
3	15	= 5000	N	$Opt(N) + 15$
4	17	= 5010	N	$Opt(N) + 5$
5	16	$\leq 10^4$	N	$Opt(N)$
6	9	= 8778	500	$Opt(N)$
7	8	= 9889	500	$Opt(N)$
8	9	$\leq 10^4$	500	$Opt(N)$

Тук $Opt(N)$ е доказуемо минималният възможен брой заявки за даденото N .

**ПЪРВО КОНТРОЛНО СЪСТЕЗАНИЕ
НА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР
Група G, 25 април 2021 г.**

Примерна комуникация

№	Действия на solve	Действия и отговори на журито
1.		solve(3)
2.	ask({{0}, {1, 2}})	return {{1, 2}, {0}}
3.	ask({{2}, {1, 0}})	return {{1}, {0, 2}}
4.	return {0, 2, 1}	

Обяснение

$$N = 3$$

$$\pi = \{0, 2, 1\}$$