

**КОНТРОЛНО ПОДБОРНО СЪСТЕЗАНИЕ
НА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР
Русе, 12-14 май 2022 г., Група А**

Задача А6. Minimize

Дадена е редица a_0, a_1, \dots, a_{N-1} от N елемента и цяло число K . Върху редицата се извършват Q_M модификации и се задават Q_I въпроси. Двата типа заявки са:

- *Модификация*. Определя се чрез стойностите pos и val и индикира стойността на a_{pos} да стане равна на val
- *Въпрос*. Да дефинираме сумата (като с \oplus бележим побитов ксор)

$$v(a) = (a_0 \oplus a_1) + (a_1 \oplus a_2) + \dots + (a_{N-2} \oplus a_{N-1})$$

Търси се минималната възможна стойност на $v(a)$ ако е позволено да се променят най-много K елемента.

Забелжете, че редицата не се модифицира при въпрос. Промяната на тези K елемента е само за изчисление на оптималната стойност и не се отразява на следващите заявки.

Напишете програма, която поддържа описаните заявки.

Детайли по имплементацията

Задачата е интерактивна. Вие трябва да имплементирате следните три функции:

```
void initialize(int N, int K, std::vector<int> a);  
void modify(int pos, int val);  
long long calculate();
```

Изпълнението на програмата ще протече по следния начин:

1. Функцията `initialize` ще бъде извикана точно веднъж в началото на програмата с аргументи стойностите N , K и низът a
2. Ще бъдат изпълнени общо Q_M извиквания на функцията `modify`, отговаряща на заявка за *модификация*, и Q_I извиквания на функцията `calculate`, отговаряща на заявка *въпрос*.

Вашата програма трябва да имплементира трите функции, но не трябва да съдържа функцията `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 `minimize.h` чрез указание към препроцесора:

```
#include "minimize.h"
```

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

КОНТРОЛНО ПОДБОРНО СЪСТЕЗАНИЕ НА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР

Русе, 12-14 май 2022 г., Група А

Локално тестване

Предоставени са Ви файловете `minimize.h` и `Lgrader.cpp`, които можете да компилирате заедно с Вашата програма, за да я тествате. Грейдърът чете данните от стандартния вход в следния формат:

- Ред 1: $N K$
- Ред 2: a_0, a_1, \dots, a_{N-1}
- Ред 3: Q
- Всеки от следващите Q реда съдържа по една заявка във един от двата формата:
 - “0” – заявка *въпрос*
 - “1 pos val” – заявка *модификация*

Забележете, че $Q = Q_M + Q_I$

Ограничения

- $1 \leq N \leq 100\,000$
- $1 \leq K \leq 20$
- $K \leq N$
- $1 \leq Q_I \leq 1\,000$
- $0 \leq Q_M \leq 50\,000$
- $0 \leq a_i < 2^{31}$

Подзадачи

Подзадача	N	Допълнителни ограничения	Точки
1	≤ 15	-	5
2	$\leq 1\,000$	$Q_I \leq 1$ $Q_M = 0$	5
3	$\leq 100\,000$	$Q_I \leq 1$ $Q_M = 0$	10
4	$\leq 1\,000$	-	10
5	$\leq 100\,000$	$Q_M \leq 1\,000$	30
6	$\leq 100\,000$	$K = 1$	2
7	$\leq 100\,000$	$K \leq 2$	2
8	$\leq 100\,000$	$K \leq 10$	6
9	$\leq 100\,000$	-	30

**КОНТРОЛНО ПОДБОРНО СЪСТЕЗАНИЕ
НА РАЗШИРЕНИЯ НАЦИОНАЛЕН ОТБОР
Русе, 12-14 май 2022 г., Група А**

Примерна интеракция

Грейдър	Вашата програма
<code>initialize(5, 2, [1, 9, 4, 6, 2])</code>	-
<code>calculate()</code>	returns 6
<code>modify(1, 3)</code>	-
<code>modify(2, 3)</code>	-
<code>calculate()</code>	returns 1

Обяснение на примерната интеракция

Първоначално $N = 5, K = 2, a = [1, 9, 4, 6, 2]$

За първия въпрос можем да променим първите две стойности получавайки редицата $[4, 4, 4, 6, 2]$, чиято стойност е

$$(4 \oplus 4) + (4 \oplus 4) + (4 \oplus 6) + (6 \oplus 2) = 0 + 0 + 2 + 4 = 6$$

След модификация $a_1 = 3$ имаме $a = [1, 3, 4, 6, 2]$

След модификация $a_2 = 3$ имаме $a = [1, 3, 3, 6, 2]$

За втория въпрос можем да променим първите стойностите на позиции 0 и 3 получавайки редицата $[3, 3, 3, 3, 2]$, чиято стойност е

$$(3 \oplus 3) + (3 \oplus 3) + (3 \oplus 3) + (3 \oplus 2) = 0 + 0 + 0 + 1 = 1$$