

XORting

Χρονικό όριο: 2 δευτερόλεπτα

Όριο μνήμης: 1024 MiB

Αυτό το πρόβλημα είναι διαδραστικό.

Ο Ίλιγιαν έκανε μία φάρσα στον Γιόνας: έχει κρύψει μια μετάθεση $p_0, p_1, \dots, p_{N-1}$ των ακεραίων από 0 έως $N - 1$. Ο Γιόνας είναι απελπισμένος και θέλει να την βρει. Εσύ έχεις συμφωνήσει να τον βοηθήσεις.

Ο Ίλιγιαν δεν θα αποκαλύψει την μετάθεση κατευθείαν, αλλά συμφώνησε να απαντά σε ερωτήσεις της ακόλουθης μορφής: μπορείς να επιλέξεις δύο δείκτες i και j τέτοιους ώστε $0 \leq i, j < N$, και σου λέει το bitwise XOR (αποκλειστικό OR) του p_i και του p_j .

Το bitwise XOR (αποκλειστικό - exclusive OR) δύο αριθμών, που συμβολίζεται με $\oplus$, εκτελεί τη λογική πράξη exclusive OR σε κάθε ζευγάρι αντίστοιχων bits. Το αποτέλεσμα σε κάθε θέση είναι 1 αν ακριβώς ένα από τα bit είναι 1, αλλά θα είναι 0 αν και τα δύο είναι 0 ή και τα δύο είναι 1. Για παράδειγμα, $75 \oplus 29 = 86$, επειδή $1001011_{(2)} \oplus 0011101_{(2)} = 1010110_{(2)}$.

Στη C++, μπορείς να υπολογίσεις το XOR των `a` και `b` χρησιμοποιώντας τον τελεστή `^`.

Ο Ίλιγιαν είναι διατεθειμένος να απαντήσει **το πολύ** σε Q ερωτήσεις.

Ωστόσο, αυτές οι ερωτήσεις δεν μπορούν πάντα να προσδιορίσουν την κρυμμένη μετάθεση με μοναδικό τρόπο. Ορίζουμε μια μετάθεση $a_0, a_1, \dots, a_{N-1}$ να είναι *δυσδιάκριτη* (indistinguishable) από την p αν $a_i \oplus a_j = p_i \oplus p_j$ για κάθε ζεύγος δεικτών i και j τέτοιο ώστε $0 \leq i, j < N$. Κάθε απάντηση που δίνει ο Ίλιγιαν είναι ίδια είτε η κρυμμένη μετάθεση είναι p , είτε είναι a , οπότε δεν υπάρχει κάποια ακολουθία ερωτήσεων που να μπορεί να τις διακρίνει. Σημειώστε ότι η p είναι πάντα δυσδιάκριτη (indistinguishable) από τον εαυτό της και ότι μπορεί να μην υπάρχει κάποια άλλη τέτοια μετάθεση.

Πρέπει να βρείτε τη λεξικογραφικά μικρότερη μετάθεση που είναι δυσδιάκριτη (indistinguishable) από την p .

Μια μετάθεση $x_0, x_1, \dots, x_{N-1}$ είναι λεξικογραφικά μικρότερη από την $y_0, y_1, \dots, y_{N-1}$ αν, στην πρώτη θέση k όπου διαφέρουν, $x_k < y_k$. Για παράδειγμα, η

$x = [2, 0, 1, 4, 3]$ είναι λεξικογραφικά μικρότερη από την $y = [2, 0, 3, 1, 4]$, επειδή η πρώτη θέση στην οποία διαφέρουν είναι $k = 2$, και $x_2 = 1 < 3 = y_2$.

Η κρυμμένη μετάθεση είναι καθορισμένη πριν ξεκινήσει το πρόγραμμά σας και δεν αλλάζει καθώς απαντώνται οι ερωτήσεις σας.

Λεπτομέρειες υλοποίησης

Το πρόγραμμά σας πρέπει να υλοποιήσει τη συνάρτηση `solve`:

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

- N : το πλήθος των στοιχείων της μετάθεσης.

Αυτή η συνάρτηση καλείται ακριβώς T φορές, μία φορά για κάθε κρυμμένη μετάθεση. Πρέπει να επιστρέψει τη λεξικογραφικά μικρότερη μετάθεση που δεν μπορεί να διακριθεί (είναι indistinguishable) από τη μετάθεση που είναι κρυμμένη για αυτήν την κλήση συνάρτησης.

Για να επικοινωνήσετε με το πρόγραμμα του κριτή, σας παρέχεται η συνάρτηση:

```
int get_xor(int i, int j)
```

Κάθε φορά που καλείται η συνάρτηση `get_xor`, επιστρέφει $p_i \oplus p_j$. Και τα δύο ορίσματα πρέπει να είναι έγκυροι δείκτες (indices) της μετάθεσης. Αν αυτή η συνθήκη δεν ικανοποιείται, η λύση σου θα λάβει την απάντηση `Output isn't correct: Invalid argument`. Μπορείτε να υποθέσετε ότι η συνάρτηση ανταποκρίνεται σε σταθερό χρόνο $O(1)$.

Η τιμή του Q είναι σταθερή για κάθε υποπρόβλημα και δεν δίνεται στο πρόγραμμά σας.

Περιορισμοί

- $1 \leq N_i \leq 2^{20}$ για κάθε $1 \leq i \leq T$, όπου N_i είναι η τιμή του N στην i -οστή κλήση της `solve`
- $1 \leq T \leq 2^{10}$
- $T \cdot \max(N_1, N_2, \dots, N_T) \leq 2^{25}$.
- Κατά την i -οστή κλήση της `solve` σε μία περίπτωση ελέγχου μπορείς να κάνεις το πολύ Q_i ερωτήσεις, όπου Q_i είναι είτε N_i , είτε N_i^2 , ανάλογα με το υποπρόβλημα.

Παραδείγματα

Θεωρήστε την ακόλουθη αλληλεπίδραση (interaction), στην οποία η `solve` καλείται δύο φορές:

Το πρόγραμμα σας	Πρόγραμμα κριτή
	<code>solve(4)</code>
<code>get_xor(0, 1)</code>	2
<code>get_xor(0, 2)</code>	3
<code>get_xor(0, 3)</code>	1
<code>get_xor(1, 2)</code>	1
<code>get_xor(1, 3)</code>	3
<code>get_xor(2, 3)</code>	2
<code>return {0, 2, 3, 1}</code>	
	<code>solve(1)</code>
<code>return {0}</code>	

Επεξήγηση της πρώτης κλήσης

Η κρυμμένη μετάθεση είναι $[2, 0, 1, 3]$, αλλά η λεξικογραφικά μικρότερη που είναι αδιάκριτη από αυτήν είναι η $[0, 2, 3, 1]$. Έξι ερωτήσεις τέθηκαν για $N_1 = 4$, το οποίο επιτρέπεται επειδή $Q_1 = N_1^2 = 16$ σε αυτό το υποπρόβλημα.

Επεξήγηση της δεύτερης κλήσης

Η μόνη δυνατή μετάθεση με $N_2 = 1$ είναι $[0]$.

Υποπροβλήματα

Υποπρόβλημα	Πόντοι	N_i	T	Q_i	Περαιτέρω περιορισμοί
0	0	-	-	N_i^2	Το παράδειγμα.
1	7	$\leq 2^3$	2^{10}	N_i^2	-
2	18	$\leq 2^7$	2^5	N_i^2	-
3	5	$\leq 2^{11}$	2^5	N_i^2	-
4	5	$\leq 2^{11}$	2^5	N_i	-
5	5	$\leq 2^{18}$	2^5	N_i	$N_i = 2^k$ για κάποιο ακέραιο k .
6	5	$\leq 2^{18}$	2^5	N_i	N_i είναι περιττός.
7	30	$\leq 2^{18}$	2^5	N_i	-
8	25	$\leq 2^{20}$	2^5	N_i	-

Ενδεικτικός βαθμολογητής

Η μορφή της εισόδου είναι η εξής:

- γραμμή 1: δύο ακέραιοι – οι τιμές των T και Q_{type} , όπου Q_{type} είναι είτε 1 είτε 2,
- οι επόμενες $2T$ γραμμές περιγράφουν τα σενάρια:
 - γραμμή $2i$: ένας ακέραιος – η τιμή του N_i για την i -οστό σενάριο,
 - γραμμή $2i + 1$: N_i ακέραιοι – η μετάθεση $p_0, p_1, \dots, p_{N_i-1}$ για το i -οστό σενάριο.

Αν $Q_{type} = 1$, τότε $Q_i = N_i$ για την i -οστή κλήση της `solve`; αν $Q_{type} = 2$, τότε $Q_i = N_i^2$.

Η μορφή της εξόδου είναι η εξής:

- γραμμή i : η μετάθεση που επιστρέφει το πρόγραμμά σας στο i -οστό σενάριο.