

# XOR Sort

Σας δίνεται ένας ακέραιος αριθμός  $S$  και ένας πίνακας  $A$  που αποτελείται από  $N$  μη αρνητικούς ακέραιους αριθμούς, με δείκτες από το 1. Επιτρέπεται να εκτελέσετε την ακόλουθη λειτουργία σε αυτήν: επιλέξτε οποιοδήποτε δείκτη  $i$  ( $1 \leq i \leq N$ ), επιλέξτε έναν από τους γειτονικούς του  $j$  ( $1 \leq j \leq N$ , είτε  $j = i - 1$  ή  $j = i + 1$ ) και αντικαταστήστε  $A_i$  με  $(A_i \oplus A_j)$  όπου  $\oplus$  είναι η λειτουργία bitwise/δυναδική/ψηφιακή XOR. Μπορείτε να δείτε τον ορισμό της XOR στο τέλος της εκφώνησης.

Ο στόχος σας είναι να μετατρέψετε τον  $A$  σε μια ταξινομημένη σειρά:

- Εάν  $S = 1$  τότε ο τελικός πίνακας πρέπει να αυξάνεται αυστηρά, i.e.  $A_i < A_{i+1}$  για  $1 \leq i < N$
- Εάν  $S = 2$  τότε ο τελικός πίνακας πρέπει να μην μειώνεται, i.e.  $A_i \leq A_{i+1}$  για  $1 \leq i < N$

Βρείτε οποιαδήποτε ακολουθία λειτουργιών που επιτυγχάνει τον στόχο σας.

Δεν απαιτείται να ελαχιστοποιήσετε τον αριθμό των εργασιών, αρκεί το ποσό τους να μην υπερβαίνει τις 40000.

## Είσοδος

Η πρώτη γραμμή περιέχει δύο ακέραιους αριθμούς:  $N$  και  $S$

Η επόμενη γραμμή περιέχει  $N$  ακέραιους αριθμούς: στοιχεία του  $A$

## Έξοδος

Η πρώτη γραμμή εξόδου πρέπει να περιέχει έναν ακέραιο  $K$  ( $0 \leq K \leq 40000$ ) - ο αριθμός των εργασιών.

Οι επόμενες  $K$  γραμμές θα πρέπει να περιέχουν δύο ακέραιους ο καθένας, περιγράφοντας τις λειτουργίες με χρονολογική σειρά: ο πρώτος ακέραιος είναι ένας δείκτης  $i$  του στοιχείου που αντικαθίσταται και ο δεύτερος είναι ένας δείκτης  $j$  ενός άλλου στοιχείου που εμπλέκεται στη λειτουργία.

## Περιορισμοί

- $1 \leq S \leq 2$
- $2 \leq N \leq 1000$
- $0 \leq A_i < 2^{20}$

## Υποπροβλήματα

1. (25 βαθμοί)  $2 \leq N \leq 150$ ,  $S = 1$ , Όλα τα στοιχεία του A είναι διακριτά
2. (35 βαθμοί)  $2 \leq N \leq 200$ ,  $S = 1$ , Όλα τα στοιχεία του A είναι διακριτά
3. (40 βαθμοί)  $2 \leq N \leq 1000$ ,  $S = 2$

## Παραδείγματα

Είσοδος	Έξοδος
5 1 3 2 8 4 1	3 1 2 4 3 5 4
5 2 4 4 2 0 1	3 3 2 4 3 5 4

Πρώτο παράδειγμα εξήγησης εξόδου:

$[3, 2, 8, 4, 1] \rightarrow [1, 2, 8, 4, 1] \rightarrow [1, 2, 8, 12, 1] \rightarrow [1, 2, 8, 12, 13]$

Δεύτερο παράδειγμα εξήγησης εξόδου:

$[4, 4, 2, 0, 1] \rightarrow [4, 4, 6, 0, 1] \rightarrow [4, 4, 6, 6, 1] \rightarrow [4, 4, 6, 6, 7]$

Κατά την εκτέλεση της λειτουργίας XOR μεταξύ  $a$  και  $b$  ψηφίων, το αποτέλεσμα θα είναι 0 εάν  $a = b$ , αλλιώς θα είναι 1.

Κατά την εκτέλεση της bitwise/δυναδικής/ψηφιακής λειτουργίας XOR μεταξύ ακέραιων  $a$  και  $b$ , τα αποτελέσματα XOR θα πραγματοποιούνται για καθένα από τα αντίστοιχα ψηφία:

$$75 \oplus 29 = 86$$

$$1001011 \oplus 0011101 = 1010110$$

Στη C/C++/Java μπορείτε να χρησιμοποιήσετε τον τελεστή “^” για να εκτελέσετε την XOR.