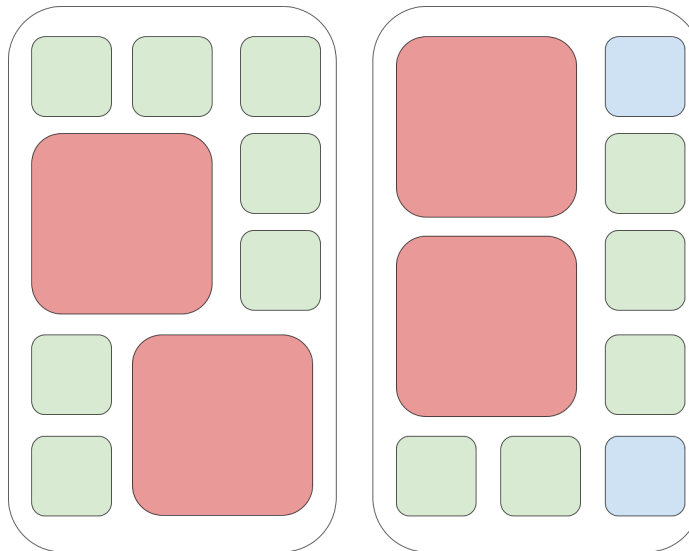


STEFANOS SET1

1. Έστω μια επιφάνεια ενός κινητού. Αυτή η επιφάνεια μπορεί να περιέχει κάποια παράθυρα. Κάθε παράθυρο έχει μέγεθος 5×3 , δηλαδή 5 γραμμές και 3 στήλες. Έστω ότι έχουμε x εφαρμογές με μέγεθος εικόνας 1×1 , δηλαδή 1 κελί. Επιπλέον έχουμε y εφαρμογές με μέγεθος εικόνας 2×2 (κάθε μια από αυτές πιάνει δηλαδή 4 κελιά στο παράθυρο). Κάθε κελί μπορεί να αντιστοιχεί σε μόνο μια εικόνα. Να γράψετε αλγόριθμο που αφού διαβάσει έναν ακέραιο αριθμό t που θα δηλώνει τον αριθμό των *test cases*, διαβάζει t φορές τους δύο αριθμούς x και y και τοποθετεί όλες τις εικόνες των εφαρμογών σε παράθυρα με τέτοιο τρόπο ώστε να χρησιμοποιούνται όσα λιγότερα παράθυρα γίνεται. Να εμφανίζει για κάθε *test case* τα ελάχιστα παράθυρα που απαιτούνται. Επιπλέον να δώσετε περιγραφική απόδειξη του αλγορίθμου σας.



Σχήμα 1: Παράδειγμα για $x = 12$ και $y = 4$

Στο παραπάνω παράδειγμα βλέπουμε ότι απαιτούνται 2 παράθυρα.

2. Έστω ο Στέφανος ο οποίος έχει ένα *string* s , το οποίο αποτελείται μόνο από πεζά λατινικά γράμματα. Κωδικοποιεί αυτό το *string* χρησιμοποιώντας τον παρακάτω αλγόριθμο:

(α') Πρώτα φτιάχνει ένα *string* r το οποίο περιλαμβάνει όλα τα γράμματα του s από μια φορά (σβήνει δηλαδή τα *duplicates*) σε αλφαβητική σειρά.

(β') Αντικαθιστά κάθε γράμμα του s με το αντίστοιχο συμμετρικό γράμμα στο *string* r .

Για παράδειγμα το *string* *codeforces* κωδικοποιείται ως εξής:

Το *string* r είναι το *cdefors*, ο πρώτος χαρακτήρας c αντικαθίσταται από το s , ο δεύτερος χαρακτήρας o από το e , το d από το r και πάει λέγοντας μέχρι το τελευταίο s να αντικατασταθεί από το c . Άρα το αποτέλεσμα είναι το *serofedsoc*.



Σχήμα 2: Οι συμμετρικοί χαρακτήρες απ' όπου παίρνουμε την αντικατάσταση

Να γράψετε αλγόριθμο που αφού διαβάσει για κάθε *test case* το μέγεθος της συμβολοσειράς και ένα *string* το οποίο έχει κωδικοποιηθεί με βάση την παραπάνω διαδικασία, να εμφανίζει από πιο *string* προέρχεται (να πραγματοποιεί δηλαδή αποκωδικοποίηση). Τον αριθμό των *test cases* τον διαβάζει στην αρχή.

3. Έστω ένας πίνακας a με n ακεραίους. Για κάθε j ($1 \leq j \leq n-2$) δηλώνουμε την τριπλέτα από τα στοιχεία $[a_j, a_{j+1}, a_{j+2}]$. Δύο τριπλέτες b και c δηλώνονται *beautiful* αν διαφέρουν σε ακριβώς (δηλαδή σε μια και μόνο) θέση. Έστω τώρα ότι έχουμε t *test cases* από τα οποία στο κάθε ένα διαβάζουμε το μέγεθος του πίνακα και τα στοιχεία του πίνακα.

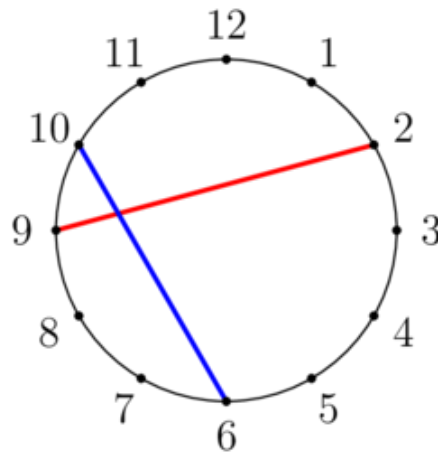
Σκεφτείτε αρχικά αλγόριθμο που λύνει το πρόβλημα σε οποιαδήποτε πολυπλοκότητα θέλετε. Τι πολυπλοκότητα έχει ο αλγόριθμος σας; Να είστε όσο πιο ακριβής γίνεται.

Μπορείτε να σκεφτείται αλγόριθμο που έχει καλύτερη πολυπλοκότητα από $O(n^2)$; Αν ναι, τότε να τον περιγράψετε.

Βοήθεια: Πιθανότατα θα χρειαστείτε κάποιο *container* της *STL*.

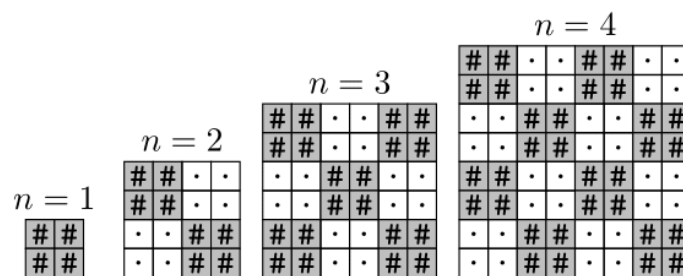
4. Έστω t *test cases* από τα οποία διαβάζουμε σε κάθε ένα έναν θετικό ακέραιο $n \leq 10^9$. Να εμφανίζει για κάθε n ένα οποιοδήποτε ζευγάρι ακεραίων (x, y) ($1 \leq x, y \leq n$) για τα οποία ισχύει ότι $x^y * y + y^x * x = n$. Αν δεν υπάρχει τέτοιο ζευγάρι τότε να εμφανίζει τον αριθμό -1 . Να δώσετε την πολυπλοκότητα του αλγορίθμου σας και να τον αποδείξετε.

5. Έστω ένα ρολόι το οποίο περιλαμβάνει τους αριθμούς από το 1 έως το 12. Η *Alice* και ο *Bob* έχουν τέσσερις διαφορετικούς αριθμούς a, b, c, d που δεν είναι μεγαλύτεροι από 12. Η *Alice* δημιουργεί ένα κόκκινο ευθύγραμμο τμήμα ενώνοντας στο ρολόι τα a και b και ο *Bob* αντίστοιχα τα c και d με ένα μπλε ευθύγραμμο τμήμα. Να δώσετε αποδοτικό αλγόριθμο που απαντάει με *YES* ή *NO* για κάθε *test case* (που αποτελεί την τετράδα $a, b, c, d \leq 12$) από τα t , αν αυτά τα ευθύγραμμο τμήματα τέμνονται.



Σχήμα 3: Παράδειγμα για το *test case* (2,9,10,6)

6. Σας δίνονται t ακέραιοι. Για κάθε έναν n από αυτούς, εμφανίστε ένα $2n \times 2n$ checkerboard φτιαγμένο από 2×2 εναλλασσόμενα τετράγωνα των χαρακτήρων '#' και '.' με το πάνω αριστερά τετράγωνο να είναι με τον χαρακτήρα '#'.



Σχήμα 4: Παραδείγματα για $n = 1, 2, 3, 4$

7. Σας δίνεται ένα *string* s με μέγεθος n που περιλαμβάνει μόνο πεζά λατινικά γράμματα. Να γράψετε αποδοτικό αλγόριθμο ο οποίος βρίσκει το ελάχιστο μέγεθος του *string* k έτσι ώστε πολλαπλά (μπορεί και ένα) αντιγραφές του k οι οποίες έχουν ενωθεί (*concatenated*) μαζί, σχηματίζουν ένα *string* με το ίδιο μέγεθος με την συμβολοσειρά s και διαφέρουν **πολύ** έναν χαρακτήρα.

Πιο αυστηρά, βρείτε το μέγεθος του μικρότερου *string* k , τέτοιο ώστε αν $c = \underbrace{k + k + \dots + k}_{x \text{ φορές}}$

για κάποιο θετικό αριθμό x , τα *strings* s και c έχουν το ίδιο μέγεθος και $c_i \neq s_i$ για το πολύ ένα i .

Ως είσοδο να διαβάζει έναν ακέραιο αριθμό t και στην συνέχεια να διαβάζει t *strings* και το μέγεθος τους. Για κάθε *string* εμφανίζει έναν ακέραιο θετικό αριθμό, ο οποίος είναι και το ζητούμενο.

8. Γράψτε έναν αλγόριθμο που διαβάζει έναν πίνακα a με n ακεραίους για τους οποίους ισχύει ότι $0 \leq a_i \leq 10^{18}$ και εμφανίζει τον αριθμό που εμφανίζεται τις περισσότερες φορές αλλά και το πόσες εμφανίζεται. Τι πολυπλοκότητα έχει ο αλγόριθμος σας; Αλλάζει κάτι αν ισχύει ότι $0 \leq a_i \leq 10^2$; Τροποποιήστε την τελευταία λύση σας ώστε να δουλεύει και για $-10^2 \leq a_i \leq 10^2$.

9. Η Μάγδα έχει ένα *grid* με 2 γραμμές και n στήλες. Ζωγραφίζει κάθε κελί με το χρώμα κόκκινο, πράσινο ή μπλε. Δυστυχώς λόγω προβλήματος της Μάγδας δεν μπορεί να ξεχωρίσει το πράσινο από το μπλε. Εμφανίστε *YES* αν η Μάγδα θεωρήσει ότι οι 2 γραμμές του *grid* έχουν ζωγραφιστεί ίδια και *NO* αν έχουν ζωγραφιστεί διαφορετικά. Ως είσοδος είναι ένας ακέραιος t (*test cases*), και για κάθε *test case* το n , και δύο *strings*.

10. n συμμετέχοντες σε έναν διαγωνισμό χωρίζονται σε m ομάδες με κάποιο τρόπο έτσι ώστε κάθε ομάδα να έχει τουλάχιστον ένα άτομο. Μετά τον διαγωνισμό κάθε ζευγάρι από τους συμμετέχοντες που βρίσκονται στην ίδια ομάδα γίνονται φίλοι. Να γράψετε πρόγραμμα που βρίσκει τον ελάχιστο και τον μέγιστο αριθμό ζευγαριών φίλων που μπορούν να γίνουν στο τέλος του διαγωνισμού. Ισχύει ότι $m, n \leq 10^9$. Τι πολυπλοκότητα έχει ο αλγόριθμος σας; Μπορείτε να λύσετε το πρόβλημα σε $O(1)$;