

Πρόβλημα 1 (sndlast)

Εύκολο 0

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να διαβάζει ένα φυσικό αριθμό $N > 10$ και να εκτυπώνει το προτελευταίο ψηφίο του.

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει τον αριθμό N . Θεωρήστε δεδομένο ότι το N θα είναι έγκυρος φυσικός αριθμός και ότι $10 < N < 100.000.000$.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει ακριβώς ένα ψηφίο: την απάντησή σας.

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

9923

Παράδειγμα εξόδου

2

Πρόβλημα 2 (mult67)

Εύκολο 2 (2008-09)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να διαβάζει ένα φυσικό αριθμό N και να ελέγχει αν είναι πολλαπλάσιο του 7 αλλά όχι πολλαπλάσιο του 6.

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει τον αριθμό N . Θεωρήστε δεδομένο ότι το N θα είναι έγκυρος φυσικός αριθμός.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει "yes", αν ο αριθμός είναι πολλαπλάσιο του 7 αλλά όχι πολλαπλάσιο του 6, διαφορετικά "no".

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

700

Παράδειγμα εξόδου

yes

Παράδειγμα εισόδου 2

42

Παράδειγμα εξόδου 2

no

Πρόβλημα 3 (threedig)

Εύκολο 1 (2008-09)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να διαβάζει ένα φυσικό αριθμό N και να ελέγχει αν είναι ή όχι τριψήφιος.

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει τον αριθμό N. Θεωρήστε δεδομένο ότι το N θα είναι έγκυρος φυσικός αριθμός.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει "yes", αν ο αριθμός είναι τριψήφιος, διαφορετικά "no".

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

789

Παράδειγμα εξόδου

yes

Παράδειγμα εισόδου 2

5934

Παράδειγμα εξόδου 2

no

Πρόβλημα 4 (end24)

Εύκολο 3 (2008-09)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να διαβάζει ένα θετικό φυσικό αριθμό N και να ελέγχει αν το τετράγωνό του τελειώνει σε 24 (γραμμένο στο δεκαδικό σύστημα αρίθμησης).

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει τον αριθμό N . Θεωρήστε δεδομένο ότι το N θα είναι έγκυρος θετικός φυσικός αριθμός και ότι το τετράγωνό του θα μπορεί να αναπαρασταθεί από τον τύπο integer της Pascal.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει "yes", αν το τετράγωνο του N (γραμμένο στο δεκαδικό σύστημα αρίθμησης) τελειώνει σε 24, διαφορετικά "no".

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

82

Παράδειγμα εξόδου

yes

Παράδειγμα εισόδου 2

37

Παράδειγμα εξόδου 2

no

Πρόβλημα 5 (maxfact)

Δύσκολο 0

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να διαβάζει ένα φυσικό αριθμό $N > 0$ και να εκτυπώνει το μέγιστο φυσικό αριθμό του οποίου το παραγοντικό δεν υπερβαίνει το N .

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει τον αριθμό N . Θεωρήστε δεδομένο ότι το N θα είναι έγκυρος φυσικός αριθμός και ότι $0 < N < 2.000.000.000$.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει ακριβώς έναν φυσικό αριθμό: την απάντησή σας.

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

10

Παράδειγμα εξόδου

3

Πρόβλημα 6 (cntprim)

Δύσκολο 1 (2008-09)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να διαβάζει ένα φυσικό αριθμό N και να εκτυπώνει το πλήθος των πρώτων αριθμών που δεν υπερβαίνουν το N .

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει τον αριθμό N . Θεωρήστε δεδομένο ότι το N θα είναι έγκυρος φυσικός αριθμός και ότι $N \leq 5.000.000$.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει ακριβώς έναν φυσικό αριθμό: την απάντησή σας.

Περιορισμοί

Αν χρειαστεί, θεωρήστε δεδομένο ότι δε θα υπάρχουν περισσότεροι από 500.000 πρώτοι αριθμοί που δεν υπερβαίνουν το N .

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

29

Παράδειγμα εξόδου

10

Πρόβλημα 7 (cntdivs)

Δύσκολο 2 (2008-09)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να διαβάζει ένα φυσικό αριθμό N και να εκτυπώνει το πλήθος των θετικών φυσικών αριθμών που δεν υπερβαίνουν το N και έχουν μη τετριμμένους κοινούς διαιρέτες με το N . (Οι μη τετριμμένοι διαιρέτες είναι μεγαλύτεροι του 1.)

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει τον αριθμό N . Θεωρήστε δεδομένο ότι το N θα είναι έγκυρος φυσικός αριθμός και ότι $N \leq 3.000.000$.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει ακριβώς έναν φυσικό αριθμό: την απάντησή σας.

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

10

Παράδειγμα εξόδου

Πρόβλημα 8 (sumofpow)

Δύσκολο 3 (2008-09)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να διαβάζει ένα φυσικό αριθμό N και να εκτυπώνει το πλήθος των φυσικών αριθμών που δεν υπερβαίνουν το N και μπορούν να γραφούν ως άθροισμα μιας δύναμης του 2, μιας δύναμης του 3 και μιας δύναμης του 5 (π.χ. $42 = 16 + 1 + 25 = 2^4 + 3^0 + 5^2$).

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει τον αριθμό N . Θεωρήστε δεδομένο ότι το N θα είναι έγκυρος φυσικός αριθμός και ότι $N \leq 5.000.000$.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει ακριβώς έναν φυσικό αριθμό: την απάντησή σας.

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

10

Παράδειγμα εξόδου

8

Πρόβλημα 9 (greater)

Εύκολο 1 (2009-10)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να διαβάζει δύο φυσικούς αριθμούς και να εκτυπώνει τον μεγαλύτερο.

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει τους αριθμούς M και N , χωρισμένους με ένα κενό διάστημα. Θεωρήστε δεδομένο ότι τα M και N θα είναι έγκυροι φυσικοί αριθμοί.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει ακριβώς έναν αριθμό: τον μεγαλύτερο των M και N .

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

99 23

Παράδειγμα εξόδου

99

Παράδειγμα εισόδου 2

0 42

Παράδειγμα εξόδου 2

42

Παράδειγμα εισόδου 3

16 16

Παράδειγμα εξόδου 3

16

Πρόβλημα 10 (swaps)

Δύσκολο 1 (2009-10)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Δίνεται μία ακολουθία αποτελούμενη από N ακέραιους αριθμούς, διάφορους του μηδέν. Ξεκινάμε να παίζουμε το εξής παιχνίδι: Σε κάθε βήμα του παιχνιδιού, βρίσκουμε **όλα τα ζεύγη γειτονικών αριθμών** της ακολουθίας, για τα οποία ο αριστερός αριθμός είναι θετικός και ο δεξιός αριθμός είναι αρνητικός. Για κάθε ένα τέτοιο ζεύγος, αντιμετωπίζουμε τους δύο αριθμούς. Στη συνέχεια προχωρούμε στο επόμενο βήμα. Επαναλαμβάνουμε με όσα βήματα χρειαστούν, μέχρι που να μην μπορούν να γίνουν άλλες αντιμεταθέσεις.

Γράψτε ένα πρόγραμμα που να υπολογίζει: πόσες αντιμεταθέσεις θα γίνουν σε όλη τη διάρκεια του παιχνιδιού, ποιος θα είναι ο πρώτος και ποιος ο τελευταίος αριθμός της ακολουθίας μετά το τελευταίο βήμα.

Δεδομένα εισόδου

Η πρώτη γραμμή της εισόδου θα περιέχει το πλήθος των στοιχείων της ακολουθίας N . Η δεύτερη γραμμή της εισόδου θα περιέχει τους N ακέραιους αριθμούς της ακολουθίας, χωρισμένους με κενά διαστήματα. Θεωρήστε δεδομένο ότι η είσοδος θα είναι έγκυρη και ότι $2 \leq N \leq 100.000$.

Δεδομένα εξόδου

Η έξοδος πρέπει να αποτελείται από τρεις γραμμές, κάθε μία από τις οποίες θα περιέχει ακριβώς έναν ακέραιο αριθμό. Η πρώτη γραμμή θα περιέχει το πλήθος των αντιμεταθέσεων, ενώ η δεύτερη και η τρίτη θα περιέχουν αντίστοιχα τον πρώτο και τον τελευταίο αριθμό της ακολουθίας μετά το τελευταίο βήμα.

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.

Παράδειγμα εισόδου

7
10 -9 -4 5 -2 3 8

Παράδειγμα εξόδου

4
-9
8

Πρόβλημα 11 (even)

Εύκολο 2 (2009-10)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να διαβάζει δύο φυσικούς αριθμούς και να εκτυπώνει πόσοι από αυτούς είναι άρτιοι.

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει τους αριθμούς M και N , χωρισμένους με ένα κενό διάστημα. Θεωρήστε δεδομένο ότι τα M και N θα είναι έγκυροι φυσικοί αριθμοί.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει ακριβώς ένα φυσικό αριθμό μεταξύ 0 και 2 (συμπεριλαμβανομένων).

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

99 23

Παράδειγμα εισόδου 2

24 99

Παράδειγμα εξόδου 2

1

Παράδειγμα εισόδου 3

24 98

Παράδειγμα εξόδου 3

2

Πρόβλημα 12 (citycirc)

Δύσκολο 2 (2009-10)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Δίνονται N πόλεις, αριθμημένες από 1 μέχρι N (συμπεριλαμβανομένων) και διατεταγμένες κατά μήκος μιας κυκλικής διαδρομής (η επόμενη της πόλης k στην κυκλική διαδρομή είναι η πόλη $k+1$, όταν $k < N$, και η πόλη 1, όταν $k = N$). Ένας κρατικός υπάλληλος πρόκειται να επισκεφθεί όλες τις πόλεις με τη σειρά που προαναφέρθηκε και να διευθετήσει τις φορολογικές εκκρεμότητες των πολιτών απέναντι στο κράτος (καταβολή φόρου) και του κράτους απέναντι στους πολίτες (επιστροφή φόρου) σε κάθε πόλη. Για κάθε πόλη k (όπου $1 \leq k \leq N$) δίνεται ένας ακέραιος b_k που ισούται με το άθροισμα των φορολογικών εκκρεμοτήτων των κατοίκων της πόλης k (ο αριθμός αυτός μπορεί να είναι θετικός, αρνητικός, ή 0). Έτσι αν το b_k είναι θετικό, ο υπάλληλος φεύγει από την πόλη k με b_k ευρώ περισσότερα απ' όσα είχε όταν ήρθε, αν το b_k είναι αρνητικό, ο υπάλληλος φεύγει από την πόλη k με $|b_k|$ ευρώ λιγότερα, και αν το b_k είναι 0, ο υπάλληλος φεύγει από την πόλη k με το ίδιο διαθέσιμο ποσό. Ο υπάλληλος δεν μπορεί να συνεχίσει την περιοδεία του αν εμφανίζεται ότι διαθέτει αρνητικό ποσό μετά την επίσκεψή του σε κάποια πόλη, αφού αυτό δηλώνει αδυναμία εκπλήρωσης κάποιων υποχρεώσεων επιστροφής φόρου στην πόλη.

Γράψτε ένα πρόγραμμα που να βρίσκει την πρώτη πόλη (δηλαδή αυτή με τον μικρότερο αριθμό), από την οποία αν ξεκινήσει ο κρατικός υπάλληλος την περιοδεία του με μηδέν ευρώ, θα μπορέσει να την ολοκληρώσει.

Δεδομένα εισόδου

Η πρώτη γραμμή της εισόδου θα περιέχει το πλήθος των πόλεων N . Η δεύτερη γραμμή της εισόδου θα περιέχει τους N ακέραιους αριθμούς b_1, b_2, \dots, b_N , χωρισμένους με κενά διαστήματα. Θεωρήσετε δεδομένο ότι η είσοδος θα είναι έγκυρη και ότι $2 \leq N \leq 1.000.000$.

Δεδομένα εξόδου

Η έξοδος πρέπει να αποτελείται από μία μόνο γραμμή που να περιέχει ακριβώς έναν φυσικό αριθμό: τον αριθμό της πρώτης πόλης από την οποία αν ξεκινήσει ο κρατικός υπάλληλος την περιοδεία του με μηδέν ευρώ, μπορεί να την ολοκληρώσει. Αν δεν υπάρχει τέτοια πόλη, ο αριθμός στην έξοδο πρέπει να είναι 0.

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.

Παράδειγμα εισόδου

6
-50 20 15 30 0 -5

Παράδειγμα εξόδου

2

Παράδειγμα εισόδου 2

8
35 10 -55 -40 30 -70 60 30

Παράδειγμα εξόδου 2

7

Παράδειγμα εισόδου 3

6
-50 20 -15 30 18 -5

Παράδειγμα εξόδου 3

0

Πρόβλημα 13 (sum42)

Εύκολο 3 (2009-10)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να διαβάζει δύο φυσικούς αριθμούς και να ελέγχει αν το άθροισμά τους τελειώνει σε 42 (γραμμένο στο δεκαδικό σύστημα αρίθμησης).

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει τους αριθμούς M και N, χωρισμένους με ένα κενό διάστημα. Θεωρήστε δεδομένο ότι τα M και N θα είναι έγκυροι φυσικοί αριθμοί.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει "yes", αν το άθροισμα των M και N (γραμμένο στο δεκαδικό σύστημα αρίθμησης) τελειώνει σε 42, διαφορετικά "no".

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

99 23

Παράδειγμα εξόδου

no

Παράδειγμα εισόδου 2

99 43

Παράδειγμα εξόδου 2

yes

Πρόβλημα 14 (sumx)

Δύσκολο 3 (2009-10)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Δίνεται μια ακολουθία n διαφορετικών μεταξύ τους θετικών ακεραίων a_1, a_2, \dots, a_n , με τιμές μεταξύ 1 και 1.000.000. Δίνεται επίσης ένας ακέραιος x . Γράψτε ένα πρόγραμμα που να βρίσκει το πλήθος των ζευγών (a_i, a_j) , όπου $1 \leq i < j \leq n$ και $a_i + a_j = x$.

Δεδομένα εισόδου

Η πρώτη γραμμή της εισόδου θα περιέχει τους αριθμούς n και x , χωρισμένους με ένα κενό διάστημα. Η δεύτερη γραμμή της εισόδου θα περιέχει τους n ακέραιους αριθμούς a_i της ακολουθίας, χωρισμένους με κενά διαστήματα. Θεωρήστε δεδομένο ότι η είσοδος θα είναι έγκυρη, $1 \leq n \leq 100.000$, $1 \leq a_i \leq 1.000.000$, και $1 \leq x \leq 2.000.000$.

Δεδομένα εξόδου

Η έξοδος πρέπει να αποτελείται από μία γραμμή που να περιέχει ακριβώς έναν ακέραιο αριθμό: το ζητούμενο πλήθος των ζευγών.

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

```
9 13
5 12 7 10 9 1 2 3 11
```

Παράδειγμα εξόδου

3

Πρόβλημα 15 (divprev)

Δύσκολο 1 (2010-11)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Δίνεται μία ακολουθία αποτελούμενη από N θετικούς ακέραιους αριθμούς. Ζητείται να βρεθεί ο μικρότερος αριθμός της ακολουθίας, ο οποίος διαιρεί ακριβώς όλους τους αριθμούς που προηγούνται αυτού στην ακολουθία.

Προφανώς ο αριθμός που εμφανίζεται πρώτος στην ακολουθία διαιρεί ακριβώς όλους τους προηγούμενούς του (γιατί δεν έχει κανέναν προηγούμενο). Άρα, αν η ακολουθία δεν είναι κενή, υπάρχει πάντα λύση στο πρόβλημα.

Δεδομένα εισόδου

Η πρώτη γραμμή της εισόδου θα περιέχει το πλήθος των στοιχείων της ακολουθίας N . Η δεύτερη γραμμή της εισόδου θα περιέχει τους N ακέραιους αριθμούς της ακολουθίας, χωρισμένους με κενά διαστήματα. Θεωρήστε δεδομένο ότι η είσοδος θα είναι έγκυρη και ότι $1 \leq N \leq 3.000.000$.

Δεδομένα εξόδου

Η έξοδος πρέπει να αποτελείται από μία γραμμή που να περιέχει ακριβώς έναν ακέραιο αριθμό: το μικρότερο αριθμό της ακολουθίας που διαιρεί ακριβώς όλους τους προηγούμενούς του.

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

10
210 84 126 462 42 51 7 77 2 3

Παράδειγμα εξόδου

42

Πρόβλημα 16 (oddeven)

Δύσκολο 3 (2010-11)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Δίνεται ακολουθία a_1, \dots, a_N αποτελούμενη από N θετικούς ακέραιους αριθμούς. Ζητείται να υπολογισθεί το εύρος του μεγαλύτερου διαστήματος στην ακολουθία στο οποίο το πλήθος των άρτιων είναι ίσο με το πλήθος των περιττών.

Δεδομένα εισόδου

Η πρώτη γραμμή της εισόδου θα περιέχει το πλήθος των στοιχείων της ακολουθίας N . Η δεύτερη γραμμή της εισόδου θα περιέχει τους N θετικούς ακέραιους αριθμούς της ακολουθίας, χωρισμένους με κενά διαστήματα. Να θεωρήσετε δεδομένο ότι η είσοδος θα είναι έγκυρη και ότι $2 \leq N \leq 1.000.000$.

Δεδομένα εξόδου

Η έξοδος πρέπει να αποτελείται από μία γραμμή που να περιέχει ακριβώς έναν ακέραιο αριθμό, το εύρος του μεγαλύτερου διαστήματος στην ακολουθία στο οποίο το πλήθος των άρτιων είναι ίσο με το πλήθος των περιττών. Σε περίπτωση που δεν υπάρχει τέτοιο διάστημα, η έξοδος πρέπει να είναι 0.

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

5
7 8 9 6 5

Παράδειγμα εξόδου

4

Παράδειγμα εισόδου 2

10
1 2 1 2 1 2 2 1 1 1

Παράδειγμα εξόδου 2

8

Παράδειγμα εισόδου 3

10
4 6 8 2 2 6 8 4 2 8

Παράδειγμα εξόδου 3

0

Πρόβλημα 17 (eqfst)

Εύκολο 1 (2010-11)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να υπολογίζει τη λύση της πρωτοβάθμιας εξίσωσης $a \cdot x + b = 0$.

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει τους πραγματικούς αριθμούς a και b , χωρισμένους με ένα κενό διάστημα. Θεωρήστε δεδομένο ότι τα a και b θα είναι έγκυροι πραγματικοί αριθμοί.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει τα εξής:

- τη λύση της εξίσωσης, αν υπάρχει, με ακρίβεια 6 δεκαδικών ψηφίων (writeln με :0:6).
 - τη λέξη "impossible", αν η εξίσωση είναι αδύνατη.
 - τη λέξη "trivial", αν η εξίσωση είναι αόριστη.
-

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

3 2

Παράδειγμα εξόδου

-0.666667

Παράδειγμα εισόδου 2

0 -1

Παράδειγμα εξόδου 2

impossible

Παράδειγμα εισόδου 3

0 0

Παράδειγμα εξόδου 3

trivial

Πρόβλημα 18 (modeq)

Εύκολο 3 (2010-11)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να δέχεται τρεις φυσικούς αριθμούς και να ελέγχει αν το υπόλοιπο της διαίρεσής τους με το 42 είναι ίσο.

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει τους τρεις φυσικούς αριθμούς, χωρισμένους ανά δύο με ένα κενό διάστημα. Θεωρήστε δεδομένο ότι οι αριθμοί θα είναι έγκυροι.

Δεδομένα εξόδου

Μόνο μία γραμμή:

- Αν το υπόλοιπο της διαίρεσης των αριθμών με το 42 είναι ίσο, η γραμμή θα περιέχει μόνο έναν ακέραιο: αυτό το υπόλοιπο.
 - Διαφορετικά, η γραμμή θα περιέχει τη λέξη "no".
-

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

45 87 129

Παράδειγμα εξόδου

3

Παράδειγμα εισόδου 2

42 42 84

Παράδειγμα εξόδου 2

0

Παράδειγμα εισόδου 3

41 42 43

Παράδειγμα εξόδου 3

no

Πρόβλημα 19 (intvsum)

Δύσκολο 2 (2010-11)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Δίνεται ακολουθία a_1, \dots, a_N αποτελούμενη από N θετικούς ακέραιους αριθμούς. Ζητείται να διαπιστωθεί αν υπάρχουν θέσεις i, j , με $i < j$, τέτοιες ώστε $a_i + a_j = a_{i+1} + \dots + a_{j-1}$, δηλαδή το άθροισμα των αριθμών στις θέσεις i και j της ακολουθίας να ισούται με το άθροισμα των αριθμών στις θέσεις από $i+1$ μέχρι και $j-1$. Αν υπάρχουν πολλά τέτοια ζεύγη θέσεων i, j στην ακολουθία, ζητείται να υπολογισθεί η μέγιστη θέση j για την οποία ισχύει η παραπάνω σχέση.

Δεδομένα εισόδου

Η πρώτη γραμμή της εισόδου θα περιέχει το πλήθος των στοιχείων της ακολουθίας N . Η δεύτερη γραμμή της εισόδου θα περιέχει τους N θετικούς ακέραιους αριθμούς της ακολουθίας, χωρισμένους με κενά διαστήματα. Να θεωρήσετε δεδομένο ότι η είσοδος θα είναι έγκυρη και ότι $3 \leq N \leq 100.000$.

Δεδομένα εξόδου

Η έξοδος πρέπει να αποτελείται από μία γραμμή που να περιέχει ακριβώς έναν ακέραιο αριθμό j , $3 \leq j \leq N$, που αντιστοιχεί στη μέγιστη θέση της ακολουθίας για την οποία υπάρχει θέση i , $i < j$, έτσι ώστε $a_i + a_j = a_{i+1} + \dots + a_{j-1}$. Αν δεν υπάρχει τέτοια θέση, η έξοδος πρέπει να είναι 0.

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

10
78 14 8 1 2 32 16 45 47 64

Παράδειγμα εξόδου

8

Παράδειγμα εισόδου 2

10
3 6 1 2 5 1 4 7 14 8

Παράδειγμα εξόδου 2

9

Παράδειγμα εισόδου 3

10
256 128 64 32 16 32 64 128 256 512

Παράδειγμα εξόδου 3

0

Πρόβλημα 20 (smthree)

Εύκολο 2 (2010-11)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να υπολογίζει τον μικρότερο μεταξύ τριών ακέραιων αριθμών.

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει τους τρεις ακέραιους αριθμούς, χωρισμένους ανά δύο με ένα κενό διάστημα. Θεωρήστε δεδομένο ότι οι αριθμοί θα είναι έγκυροι.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει τον μικρότερο από τους τρεις.

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

2 1 3

Παράδειγμα εξόδου

1

Παράδειγμα εισόδου 2

4 2 2

Παράδειγμα εξόδου 2

2

Παράδειγμα εισόδου 3

7 7 7

Παράδειγμα εξόδου 3

7

Πρόβλημα 21 (lowsum)

Εκφώνηση

Δίνεται μια ακολουθία N διαφορετικών ακεραίων (αρνητικών και θετικών) σε αύξουσα σειρά. Να γράψετε ένα πρόγραμμα το οποίο θα βρίσκει δύο διαφορετικούς αριθμούς της ακολουθίας που το άθροισμά τους είναι όσο το δυνατόν πιο κοντά στο 0.

Δεδομένα εισόδου

Η πρώτη γραμμή της εισόδου θα περιέχει το πλήθος των στοιχείων της ακολουθίας N . Η δεύτερη γραμμή της εισόδου θα περιέχει N διαφορετικούς ακέραιους αριθμούς (αρνητικούς και θετικούς) σε αύξουσα σειρά. Οι αριθμοί θα χωρίζονται με κενά διαστήματα, και μπορεί να είναι και όλοι θετικοί ή όλοι αρνητικοί. Η ακολουθία δεν θα περιέχει το 0, η απόλυτη τιμή των αριθμών θα είναι μικρότερη ή ίση του 10^9 , και θα ισχύει ότι $2 \leq N \leq 1.000.000$.

Να θεωρήσετε ως δεδομένο ότι η είσοδος θα είναι έγκυρη.

Δεδομένα εξόδου

Η έξοδος πρέπει να αποτελείται από μία γραμμή που περιέχει ακριβώς έναν ακέραιο αριθμό, ο οποίος αντιστοιχεί στο άθροισμα δύο διαφορετικών αριθμών της ακολουθίας που είναι πλησιέστερα στο 0.

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
 - Όριο μνήμης: 64 MB.
-

Παράδειγμα εισόδου

5
-101 -4 -1 6 95

Παράδειγμα εξόδου

2

Παράδειγμα εισόδου 2

8

-110 -22 -1 28 35 105 140 145

Παράδειγμα εξόδου 2

-5

Πρόβλημα 22 (count7)

Εύκολο 1 (2011-12)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να δέχεται ένα διψήφιο φυσικό αριθμό και να βρίσκει πόσα από τα ψηφία του είναι ίσα με 7.

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει το διψήφιο φυσικό αριθμό.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει ακριβώς ένα φυσικό αριθμό μεταξύ 0 και 2 (συμπεριλαμβανομένων).

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
 - Όριο μνήμης: 16 MB.
-

Παράδειγμα εισόδου

17

Παράδειγμα εξόδου

1

Παράδειγμα εισόδου 2

42

Παράδειγμα εξόδου 2

0

Παράδειγμα εισόδου 3

77

Παράδειγμα εξόδου 3

2

Πρόβλημα 23 (maxsum)

Δύσκολο 2 (2011-12)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Δίνεται μια ακολουθία N ακεραίων αριθμών (που μπορεί να είναι αρνητικοί, θετικοί ή και μηδέν). Να γράψετε ένα πρόγραμμα το οποίο θα υπολογίζει το μέγιστο άθροισμα (οσωνδήποτε) διαδοχικών όρων της ακολουθίας.

Δεδομένα εισόδου

Η πρώτη γραμμή της εισόδου θα περιέχει το πλήθος N των στοιχείων της ακολουθίας. Η δεύτερη γραμμή της εισόδου θα περιέχει N ακέραιους αριθμούς. Οι αριθμοί θα χωρίζονται με κενά διαστήματα, και η απόλυτη τιμή τους θα είναι μικρότερη ή ίση του 1000.

Να θεωρήσετε ως δεδομένο ότι $2 \leq N \leq 1.000.000$, και ότι η είσοδος θα είναι έγκυρη.

Δεδομένα εξόδου

Η έξοδος πρέπει να αποτελείται από μία γραμμή που να περιέχει ακριβώς έναν ακέραιο αριθμό, ο οποίος αντιστοιχεί στο μέγιστο άθροισμα διαδοχικών όρων της ακολουθίας.

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
 - Όριο μνήμης: 64 MB.
-

Παράδειγμα εισόδου

```
10
1 5 -1 -6 14 -8 7 2 -5 4
```

Παράδειγμα εξόδου

15

Παράδειγμα εισόδου 2

10

1 7 -10 -11 9 9 -7 2 12 -4

Παράδειγμα εξόδου 2

25

[Πρόβλημα 24 \(mul4dig\)](#)

Εύκολο 2 (2011-12)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να διαβάζει δύο φυσικούς αριθμούς και να ελέγχει αν το γινόμενό τους είναι ή όχι τετραψήφιος αριθμός.

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει τους αριθμούς M και N, χωρισμένους με ένα κενό διάστημα. Θεωρήστε δεδομένο ότι τα M και N θα είναι έγκυροι φυσικοί αριθμοί.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει "yes", αν το γινόμενο των M και N είναι τετραψήφιος αριθμός, διαφορετικά "no".

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
 - Όριο μνήμης: 16 MB.
-

Παράδειγμα εισόδου

6 7

Παράδειγμα εξόδου

no

Παράδειγμα εισόδου 2

85 47

Παράδειγμα εξόδου 2

yes

Παράδειγμα εισόδου 3

812 67

Παράδειγμα εξόδου 3

no

Πρόβλημα 25 (skating)

Δύσκολο 3 (2011-12)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Ο μικρός Λουκάς παίζει με το πατίνι του. Κουράζεται όμως εύκολα κι έτσι βασίζεται αποκλειστικά στην κλίση του δρόμου. Έστω ότι ο δρόμος όπου κάνει πατίνι ο Λουκάς χωρίζεται σε N μικρά τμήματα και ότι η κλίση καθενός από αυτά παριστάνεται με έναν ακέραιο αριθμό. Αν, για παράδειγμα, ο αριθμός που αντιστοιχεί σε κάποιο τμήμα είναι 3, τότε το τμήμα αυτό είναι κατηφορικό και η ταχύτητα του πατινιού θα αυξηθεί κατά 3 όταν περάσει από εκεί. Αντίθετα, αν ο αριθμός είναι -4, τότε το τμήμα αυτό είναι ανηφορικό και η ταχύτητα του πατινιού θα μειωθεί κατά 4 όταν περάσει από εκεί. Τα επίπεδα τμήματα παριστάνονται με μηδέν.

Ο Λουκάς μπορεί να διαλέξει από ποιο σημείο του δρόμου θα ξεκινήσει με το πατίνι του. Η αρχική του ταχύτητα είναι μηδενική και θέλει να φτάσει στο τέλος του δρόμου με τη μέγιστη δυνατή ταχύτητα.

Να γράψετε ένα πρόγραμμα το οποίο θα βρίσκει τη μέγιστη δυνατή ταχύτητα στο τέλος του δρόμου.

Δεδομένα εισόδου

Η πρώτη γραμμή της εισόδου θα περιέχει το πλήθος N των τμημάτων του δρόμου. Η δεύτερη γραμμή της εισόδου θα περιέχει N ακέραιους αριθμούς. Οι αριθμοί θα χωρίζονται με κενά διαστήματα, και η απόλυτη τιμή τους θα είναι μικρότερη ή ίση του 1000.

Να θεωρήσετε ως δεδομένο ότι $2 \leq N \leq 1.000.000$, και ότι η είσοδος θα είναι έγκυρη.

Δεδομένα εξόδου

Η έξοδος πρέπει να αποτελείται από μία γραμμή που να περιέχει ακριβώς έναν ακέραιο αριθμό: τη μέγιστη ταχύτητα με την οποία μπορεί να φτάσει ο Λουκάς στο τέλος του δρόμου.

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
- Όριο μνήμης: 16 MB.

Παράδειγμα εισόδου

8
-1 3 -4 5 -1 4 0 -2

Παράδειγμα εξόδου

6

Παράδειγμα εισόδου 2

10
5 -6 7 -8 14 12 -11 9 -5 4

Παράδειγμα εξόδου 2

23

Παράδειγμα εισόδου 3

5
1 2 1 2 -10

Παράδειγμα εξόδου 3

0

Πρόβλημα 26 (mundial)

Εύκολο 3 (2011-12)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Το παγκόσμιο πρωτάθλημα ποδοσφαίρου (μουντιάλ) διεξήχθη για πρώτη φορά το 1930. Από τότε, διοργανώνεται κάθε 4 χρόνια. Όμως, το 1942 και το 1946 δεν έγινε, λόγω του πολέμου.

Να γράψετε ένα πρόγραμμα που να διαβάζει ένα έτος και να ελέγχει αν είναι ή όχι χρονιά μουντιάλ.

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει έναν αριθμό Y: τό έτος που μας ενδιαφέρει.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει "yes", αν το έτος Y έγινε ή πρόκειται να γίνει παγκόσμιο πρωτάθλημα ποδοσφαίρου, διαφορετικά "no".

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
 - Όριο μνήμης: 16 MB.
-

Παράδειγμα εισόδου

1970

Παράδειγμα εξόδου

yes

Παράδειγμα εισόδου 2

1942

Παράδειγμα εξόδου 2

no

Παράδειγμα εισόδου 3

2012

Παράδειγμα εξόδου 3

no

Πρόβλημα 27 (istriangle)

Εύκολο 1 (2012-13)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να διαβάζει τρεις πραγματικούς αριθμούς X , Y και Z και να ελέγχει αν οι αριθμοί αυτοί μπορούν να είναι τα μήκη των πλευρών ενός τριγώνου στο επίπεδο.

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει τους αριθμούς X , Y και Z , χωρισμένους ανά δύο με ένα κενό διάστημα. Θεωρήστε δεδομένο ότι θα είναι έγκυροι πραγματικοί αριθμοί.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει "yes", αν οι αριθμοί X , Y και Z μπορούν να είναι τα μήκη των πλευρών ενός τριγώνου στο επίπεδο, διαφορετικά "no".

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

3 10.5 8

Παράδειγμα εξόδου

yes

Παράδειγμα εισόδου 2

9.5 8 21.3

Παράδειγμα εξόδου 2

no

Παράδειγμα εισόδου 3

17 8 9

Παράδειγμα εξόδου 3

no

Πρόβλημα 28 (twoends)

Δύσκολο 1 (2012-13)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Δίνεται μια ακολουθία N θετικών ακεραίων αριθμών. Ζητείται ο μεγαλύτερος δυνατός αριθμός Σ , τέτοιος ώστε:

- Να υπάρχουν X όροι στην αρχή της ακολουθίας που να έχουν άθροισμα Σ .
- Να υπάρχουν Y όροι στο τέλος της ακολουθίας που να έχουν άθροισμα Σ .
- Να είναι $X+Y \leq N$, δηλαδή οι όροι που αθροίζουμε στα δύο άκρα να μην επικαλύπτονται.

Προσέξτε ότι το πρόβλημα έχει πάντα λύση (για $\Sigma=0$ και $X=Y=0$).

Δεδομένα εισόδου

Η πρώτη γραμμή της εισόδου θα περιέχει το πλήθος N των στοιχείων της ακολουθίας. Η δεύτερη γραμμή της εισόδου θα περιέχει N ακέραιους αριθμούς, χωρισμένους ανά δύο με ένα κενό διάστημα.

Να θεωρήσετε ως δεδομένο ότι $1 \leq N \leq 10.000.000$, ότι η είσοδος θα είναι έγκυρη και ότι το άθροισμα όλων των αριθμών δε θα υπερβαίνει το $1.000.000.000$.

Δεδομένα εξόδου

Η έξοδος πρέπει να αποτελείται από μία γραμμή που να περιέχει ακριβώς έναν ακέραιο αριθμό, το μεγαλύτερο δυνατό Σ με τις παραπάνω ιδιότητες.

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 3 sec.
- Όριο μνήμης: 128 MB.

Παράδειγμα εισόδου

7
3 2 5 7 1 6 4

Παράδειγμα εξόδου

10

Παράδειγμα εισόδου 2

10
1 2 3 4 5 6 7 8 9 45

Παράδειγμα εξόδου 2

45

Παράδειγμα εισόδου 3

5
1 2 3 4 5

Παράδειγμα εξόδου 3

0

Πρόβλημα 29 (circdisc)

Εύκολο 2 (2012-13)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να διαβάζει δύο πραγματικούς αριθμούς X και Y και να απαντά πού βρίσκεται το σημείο με συντεταγμένες (X, Y) σε σχέση με το μοναδιαίο κυκλικό δίσκο.

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει τις συντεταγμένες X και Y , χωρισμένες με ένα κενό διάστημα. Θεωρήστε δεδομένο ότι θα είναι έγκυροι πραγματικοί αριθμοί.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει "inside", αν το σημείο με συντεταγμένες (X, Y) βρίσκεται στο εσωτερικό του μοναδιαίου κύκλου, "outside" αν βρίσκεται στο εξωτερικό του, και "border" αν βρίσκεται πάνω στον κύκλο.

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
-

Παράδειγμα εισόδου

0.5 0.5

Παράδειγμα εξόδου

inside

Παράδειγμα εισόδου 2

-1 0

Παράδειγμα εξόδου 2

border

Παράδειγμα εισόδου 3

2 -3.5

Παράδειγμα εξόδου 3

outside

Πρόβλημα 30 (icecream)

Δύσκολο 2 (2012-13)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Κατά μήκος μιας ευθείας οδού κατοικούν N παιδιά. Το κάθε παιδί i κατοικεί σε μια διαφορετική ακέραια συντεταγμένη a_i επί της οδού. Πλησιάζει το καλοκαίρι, και ο παγωτατζής της γειτονιάς σκέφτεται σε ποιο σημείο της οδού θα εγκαταστήσει το κιόσκι του. Ο παγωτατζής γνωρίζει ότι κάθε παιδί είναι διατεθειμένο να διανύσει απόσταση μικρότερη ή ίση του K για να αγοράσει παγωτό. Έτσι θέλει να εγκαταστήσει το κιόσκι του στο σημείο όπου θα μπορεί να εξυπηρετήσει όσο το δυνατόν περισσότερα παιδιά.

Να γράψετε ένα πρόγραμμα που υπολογίζει το μέγιστο πλήθος παιδιών που μπορεί να εξυπηρετήσει ο παγωτατζής.

Σημείωση: Η απόσταση δύο σημείων με συντεταγμένες x και y επί της οδού είναι ίση με $|x - y|$.

Δεδομένα εισόδου

Στην πρώτη γραμμή της εισόδου θα δίνονται δύο θετικοί ακέραιοι, χωρισμένοι με ένα κενό διάστημα, που δηλώνουν το πλήθος N των παιδιών και την απόσταση K που μπορεί να διανύσει κάθε παιδί. Σε καθεμία από τις επόμενες N γραμμές θα δίνεται ένας ακέραιος a_i που δηλώνει τη συντεταγμένη του παιδιού i . Οι συντεταγμένες θα δίνονται σε αύξουσα σειρά.

Να θεωρήσετε ως δεδομένο ότι η είσοδος θα είναι έγκυρη και ότι $1 \leq N \leq 1.000.000$, $1 \leq K \leq 5.000.000$, $0 \leq a_i \leq 40.000.000$.

Δεδομένα εξόδου

Το πρόγραμμά σας πρέπει να τυπώνει, στην πρώτη γραμμή της εξόδου μόνο έναν αριθμό: το μέγιστο πλήθος παιδιών που μπορεί να εξυπηρετήσει ο παγωτατζής.

Περιορισμοί

- Όριο χρόνου εκτέλεσης: 1 sec.
- Όριο μνήμης: 64 MB.

Παράδειγμα εισόδου

6 2
1
3
4
6
7
9

Παράδειγμα εξόδου

4

Παράδειγμα εισόδου 2

9 4
1
2
3
4
5
6
7
8
9

Παράδειγμα εξόδου 2

9

Πρόβλημα 31 (dig23)

Εύκολο 1 (2013-14)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να διαβάζει δύο φυσικούς αριθμούς A και B, να υπολογίζει το γινόμενό τους και να εκτυπώνει τα δύο προτελευταία ψηφία του.

Για παράδειγμα, αν $A=4217$ και $B=1742$, είναι $A*B=7346014$ και θα πρέπει να εκτυπώνονται τα δύο προτελευταία ψηφία: 01.

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει τους αριθμούς A και B, χωρισμένους με ένα κενό διάστημα. Θεωρήστε δεδομένο ότι θα είναι έγκυροι φυσικοί αριθμοί μεταξύ των ορίων που αναφέρονται παρακάτω.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει δύο δεκαδικά ψηφία. Προσέξτε ότι το πρώτο από αυτά θα πρέπει να εκτυπώνεται ακόμη κι αν είναι μηδέν.

Περιορισμοί

- $10 \leq A < 10000$
 - $10 \leq B < 10000$
 - Όριο χρόνου εκτέλεσης: 1 sec.
 - Όριο μνήμης: 4 MB.
-

Παράδειγμα εισόδου

Παράδειγμα εξόδου

71

Παράδειγμα εισόδου 2

29 37

Παράδειγμα εξόδου 2

07

Πρόβλημα 32 (max2sum)

Δύσκολο 1 (2013-14)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Δίνεται μία ακολουθία N θετικών ακέραιων αριθμών. Επιλέγω δύο από αυτούς και υπολογίζω το άθροισμά τους συν την απόστασή τους μέσα στην ακολουθία. Για παράδειγμα, έστω ότι $N=6$ και οι αριθμοί της ακολουθίας είναι:

7	3	9	8	2	7
---	---	---	---	---	---

Αν επιλέξω το 3 και το 8, η απόστασή τους είναι 2 (γιατί το 3 είναι ο δεύτερος όρος της ακολουθίας και το 8 είναι ο τέταρτος, επομένως $4-2=2$), άρα το άθροισμα που υπολογίζω είναι $3+8+2=13$.

Γράψτε ένα πρόγραμμα που, δεδομένης μίας τέτοιας ακολουθίας, να βρίσκει τη μέγιστη δυνατή τιμή αυτού του αθροίσματος.

Προσοχή: Πρέπει να επιλέξω δύο διαφορετικούς όρους της ακολουθίας, όχι δύο φορές τον ίδιο όρο. Αυτό δεν αποκλείει να επιλεγεί ο ίδιος αριθμός, αν εμφανίζεται δύο φορές στην ακολουθία.

Δεδομένα εισόδου

Η πρώτη γραμμή της εισόδου θα περιέχει το πλήθος N των στοιχείων της ακολουθίας. Η δεύτερη γραμμή της εισόδου θα περιέχει N ακέραιους αριθμούς, χωρισμένους ανά δύο με ένα κενό διάστημα.

Να θεωρήσετε ως δεδομένο ότι $2 \leq N \leq 1.000.000$, ότι η είσοδος θα είναι έγκυρη και ότι οι αριθμοί δε θα υπερβαίνουν το 1.000.000.000.

Δεδομένα εξόδου

Η έξοδος πρέπει να αποτελείται από μία γραμμή που να περιέχει ακριβώς έναν ακέραιο αριθμό, τη μεγαλύτερη δυνατή τιμή του ζητούμενου αθροίσματος.

Περιορισμοί

- $2 \leq N \leq 1.000.000$
- Όριο χρόνου εκτέλεσης: 1 sec.
- Όριο μνήμης: 10 MB.

Παράδειγμα εισόδου

6
7 3 9 8 2 7

Παράδειγμα εξόδου

19

Πρόβλημα 33 (sum2dig)

Εύκολο 2 (2013-14)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Να γράψετε ένα πρόγραμμα που να διαβάζει έναν φυσικό αριθμό $N \geq 10$ και να εκτυπώνει το άθροισμα των δύο τελευταίων ψηφίων του.

Για παράδειγμα, αν $N=4217$, θα πρέπει να εκτυπώνεται ο αριθμός 8 (γιατί $1+7=8$).

Δεδομένα εισόδου

Μόνο μία γραμμή που θα περιέχει τον αριθμό N . Θεωρήστε δεδομένο ότι θα είναι έγκυρος φυσικός αριθμός μεταξύ των ορίων που αναφέρονται παρακάτω.

Δεδομένα εξόδου

Μόνο μία γραμμή που θα περιέχει μόνο έναν ακέραιο αριθμό: το ζητούμενο άθροισμα.

Περιορισμοί

- $10 \leq N < 1000000000$
 - Όριο χρόνου εκτέλεσης: 1 sec.
 - Όριο μνήμης: 4 MB.
-

Παράδειγμα εισόδου

4217

Παράδειγμα εξόδου

8

Παράδειγμα εισόδου 2

12345678

Παράδειγμα εξόδου 2

15

Πρόβλημα 34 (mulsum)

Δύσκολο 2 (2013-14)

[<< Επιστροφή στο 'Προπόνηση progintro Σειρά #7'](#)

Εκφώνηση

Δίνεται μία ακολουθία N φυσικών αριθμών και ένας θετικός φυσικός αριθμός M . Ζητείται το μέγιστο πολλαπλάσιο του M που μπορεί να προκύψει ως άθροισμα (οσωνδήποτε) διαδοχικών όρων της ακολουθίας.

Δεδομένα εισόδου

Η πρώτη γραμμή της εισόδου θα περιέχει ακριβώς δύο φυσικούς αριθμούς χωρισμένους μεταξύ τους με ένα κενό διάστημα: το πλήθος N των στοιχείων της ακολουθίας και τον αριθμό M . Η δεύτερη γραμμή της εισόδου θα περιέχει N ακέραιους αριθμούς, χωρισμένους ανά δύο με ένα κενό διάστημα.

Να θεωρήσετε ως δεδομένο ότι η είσοδος θα είναι έγκυρη και ότι οι αριθμοί N και M δε θα υπερβαίνουν τα παρακάτω όρια. Επίσης, το άθροισμα όλων των όρων της ακολουθίας δε θα υπερβαίνει το 1.000.000.000.

Δεδομένα εξόδου

Η έξοδος πρέπει να αποτελείται από μία γραμμή που να περιέχει ακριβώς έναν ακέραιο αριθμό, το ζητούμενο πολλαπλάσιο.

Περιορισμοί

- $1 \leq N \leq 1.000.000$
- $1 \leq M \leq 1.000.000$
- Όριο χρόνου εκτέλεσης: 1 sec.
- Όριο μνήμης: 16 MB.

Παράδειγμα εισόδου

8 7
1 2 3 4 5 6 7 8

Παράδειγμα εξόδου

35

Παράδειγμα εισόδου 2

5 17
6 4 3 7 1

Παράδειγμα εξόδου 2

0