

en[i]gma 0x02 Senior

15 Δεκεμβρίου 2024

Προβλήματα

Πρόβλημα	Χρονικό Όριο	Όριο Μνήμης	Βαθμολογία
Το κλειδί του θησαυρού	1 sec	16MB	100
Χριστουγεννιάτικα κέρασματα	1 sec	16MB	100
Ο μαγικός αριθμός	1 sec	64MB	100
Η παλιά γραφομηχανή	1 sec	64MB	100
Τώρα σκέφτεσαι με πύλες	1 sec	64MB	100
Σύνολο			500

Το κλειδί του θησαυρού

Η Αννούλα και ο αδερφός της ο Τοτός, ψάχνοντας στην αποθήκη του σπιτιού τους για να βρουν κάποια παλιά τους άλμπουμ ανακάλυψαν ένα ξεχασμένο σεντούκι. Μέσα στο σεντούκι υπάρχει ένας παλιός χάρτης που ανήκε στον θρυλικό πειρατή Henry Every. Σύμφωνα με τον χάρτη, ένας πολύτιμος θησαυρός είναι κρυμμένος κάπου στο νησί, αλλά για να φτάσουν σε αυτόν, πρέπει να λύσουν τρεις γρίφους.

Η Αννούλα και ο Τοτός είναι περίεργοι να δουν τι κρύβει μέσα το σεντούκι. Όμως, πρέπει πρώτα να ανοίξουν το λουκέτο.

Γρίφος 1ος

Στο κάτω μέρος του σεντουκιού έχει χαραγμένους δύο αριθμούς. Ο πειρατής, δίπλα από αυτούς είχε χαράξει και μια σημείωση:

«**1.** Το πλήρωμά μου αποτελείται από τόσους ναυτικούς όσο το γινόμενο των δύο αριθμών που είναι χαραγμένοι εδώ.

Αν καταφέρετε να βρείτε το γινόμενο και βάλετε τον αριθμό στην κλειδαριά, τότε το σεντούκι θα ανοίξει!»

Γρίφος 2ος

Και λίγο πιο δίπλα γράφει:

«**2.** Αφού ανοίξετε το σεντούκι, μένει μόνο να βρείτε που είναι κρυμμένος ο θησαυρός στον χάρτη. Από τους αριθμούς A και B μπορεί είτε ο A, είτε ο B, είτε και ο A και ο B να είναι 0.

Μπορεί, όμως να μην είναι κανένας 0.

- Αν κανένας δεν είναι 0, τότε ο θησαυρός κρύβεται κάτω από τον αριθμό 0 στον χάρτη,
- Αν μόνο ο A είναι 0, τότε ο θησαυρός κρύβεται κάτω από τον αριθμό 1 στον χάρτη,
- Αν μόνο ο B είναι 0, τότε ο θησαυρός κρύβεται κάτω από τον αριθμό 2 στον χάρτη,
- Αν και ο A και ο B είναι 0, τότε ο θησαυρός κρύβεται κάτω από τον αριθμό 3 στον χάρτη»

Ο Τοτός και η Αννούλα χρειάζονται τη βοήθειά σας, καθώς είναι αρκετά αγχωμένοι και πρέπει κάπως να επαληθεύσουν ότι κάνουν σωστά την πράξη και τελικά βρίσκουν το σημείο του θησαυρού!

Θέλουμε να γράψουμε ένα πρόγραμμα που θα διαβάζει από την είσοδο `STDIN` δύο ακέραιους, μονοψήφιους, θετικούς αριθμούς A και B, χωρισμένους από έναν χαρακτήρα αλλαγής γραμμής (`\n`) και θα τυπώνει στην έξοδο `STDOUT` δύο γραμμές.

Στην πρώτη θα τυπώνει το γινόμενο M των αριθμών A και B.

Στην δεύτερη θα τυπώνει έναν από τους αριθμούς 0, 1, 2 ή 3, σύμφωνα με τη σημείωση **2.** του πειρατή.

Παραδείγματα

1ο

Είσοδος (STDIN)

```
6  
8
```

Έξοδος (STDOUT)

```
48  
0
```

Εξήγηση παραδείγματος 1:

Το πλήρωμα με τους ναύτες του καπετάνιου αποτελείται από $6 \times 8 = 48$ ναύτες και **κανένας** από τους δύο αριθμούς δεν είναι το 0, άρα στην έξοδο τυπώνουμε **0**.

2ο

Είσοδος (STDIN)

```
0  
3
```

Έξοδος (STDOUT)

```
0  
1
```

Εξήγηση παραδείγματος 2:

Το πλήρωμα με τους ναύτες του καπετάνιου αποτελείται από $0 \times 3 = 0$ ναύτες και ο πρώτος αριθμός, ο Α, είναι το 0, άρα στην έξοδο τυπώνουμε **1**.

Προσοχή! Κάθε γραμμή εισόδου ή εξόδου (θα πρέπει να) τερματίζει με το χαρακτήρα αλλαγής γραμμής `'\n'`.

Ένα μικρό σκονάκι!

Python

```
from sys import exit  
  
# Η εκτέλεση του προγράμματος ξεκινά εδώ!  
if __name__ == "__main__":  
    x = int(input())  
  
    if(x == 1) : ...  
    elif(x == 2) : ...  
  
    exit(0)
```

C/C++ (με `cstdio`):

```
#include <stdio.h>

// Η εκτέλεση του προγράμματος ξεκινά εδώ!
int main() {
    int x;
    scanf("%d", &x);

    if(x == 1){...};
    else if(x == 2){...};

    return 0;
}
```

C/C++ (με `iostream`):

```
#include <iostream>

// Η εκτέλεση του προγράμματος ξεκινά εδώ!
int main() {
    int x;
    std::cin >> x; // Use std::cin for input

    if (x == 1) {
        // Προσθέστε τον κώδικα για x == 1 εδώ
    } else if (x == 2) {
        // Προσθέστε τον κώδικα για for x == 2 εδώ
    }

    return 0;
}
```

Χριστουγεννιάτικα κεράσματα

Εν όψει της γιορτής των Χριστουγέννων η μαμά το Τοτού και της Αννούλας ετοίμασε ένα σωρό γλυκά:

μελομακάρονα (1), κουραμπιέδες (2), κουλουράκια (3), δίπλες (4) και σοκολατάκια (5).

Αφού τα ολοκλήρωσε, τα συσκεύασε σε μεγάλα κουτιά, για να τα δώσει ως δώρο στους καλεσμένους. Πριν ξεκινήσει να διακοσμεί τα κουτιά-δώρα, διαπίστωσε πως δεν είχε σημειώσει πόσα κομμάτια από ένα συγκεκριμένο γλυκό είχε κάθε κουτί. Αυτό ήταν αρκετά σημαντικό, καθώς κάθε καλεσμένος είχε διαφορετικό αγαπημένο γλυκό και του αντιστοιχούσε διαφορετικό κουτί-δώρο. Αμέσως, η Αννούλα και ο Τοτός ανέλαβαν δράση και ξεκίνησαν να μετράνε πόσα κομμάτια από έναν συγκεκριμένο τύπο γλυκού έχει το κάθε κουτί.

Θέλουμε να γράψουμε ένα πρόγραμμα που θα διαβάζει από την είσοδο `STDIN` δύο θετικούς ακέραιους αριθμούς N και S , χωριζομένους από έναν χαρακτήρα κενού (' '), ακολουθούμενους από S θετικούς ακέραιους αριθμούς S_i , χωριζομένους από έναν χαρακτήρα κενού (' '). Το πρόγραμμα θα εκτυπώνει στην έξοδο `STDOUT` το πλήθος των επαναλήψεων του N ανάμεσα στους S αριθμούς.

- Ο N συμβολίζει το γλυκό που πρέπει να μετρήσουμε πόσες φορές υπάρχει στο κουτί.
- Ο S συμβολίζει το συνολικό πλήθος των γλυκών που βρίσκονται στο κουτί.
- Οι S_i συμβολίζουν κάθε γλυκό που βρίσκεται μέσα στο κουτί.

Προσοχή! Κάθε γραμμή εισόδου ή εξόδου (θα πρέπει να) τερματίζει με το χαρακτήρα αλλαγής γραμμής '\n'.

Παράδειγμα

Είσοδος (STDIN)

```
1 10
2 1 1 3 1 5 1 2 1 1
```

Έξοδος (STDOUT)

```
6
```

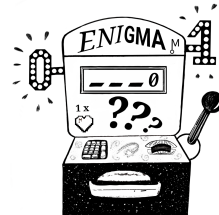
Εξήγηση παραδείγματος:

Ο Τοτός και η Αννούλα ψάχνουν να δουν πόσα κομμάτια του γλυκού 1 υπάρχουν ανάμεσα στα 10 γλυκά του κουτιού. Μετρώντας, διαπιστώνουν ότι υπάρχουν 6 τέτοια γλυκά.

Ο μαγικός αριθμός

Ο Τοτός προετοιμάζεται να μπορεί να μαντέψει τον μαγικό αριθμό και να κερδίσει στο ομώνυμο τυχερό παιχνίδι. Γι' αυτό αποφάσισε να φτιάξει το παρακάτω πρόγραμμα στον υπολογιστή για να προπονηθεί.

Αρχικά, το πρόγραμμα θα διαβάζει από την **είσοδο** τον μαγικό αριθμό. Στη συνέχεια, θα διαβάζει μερικούς αριθμούς, όσους χρειαστεί, μέχρι να βρει κάποιον που να ισούται με τον μαγικό αριθμό. Για κάθε αριθμό που διαβάζει θα εκτυπώνει στην **έξοδο**:



- 1, αν ο αριθμός που διάβασε είναι μικρότερος από τον μαγικό αριθμό,
- 2, αν ο αριθμός που διάβασε είναι μεγαλύτερος από τον μαγικό αριθμό,
- 0, όταν ο αριθμός που διάβασε είναι ίσος με τον μαγικό αριθμό.

Παραδείγματα

1ο

STDIN

```
100  
5  
100
```

STDOUT

```
1  
0
```

Εξήγηση παραδείγματος 1:

Η είσοδος περιέχει το μαγικό αριθμό 100 και 2 μαντεψιές, το 5 και το 100.

Στην έξοδο εκτυπώνεται πρώτα το 1 καθώς το $5 < 100$ και ύστερα εκτυπώνεται το 0 καθώς το $100 = 100$.

2ο

STDIN

```
10  
11  
9  
12  
10
```

STDOUT

```
2  
1  
2  
0
```

Εξήγηση παραδείγματος 2:

Η είσοδος περιέχει το μαγικό αριθμό 10 και 4 μαντεψιές το 11, το 9, το 12 και το 10.

Στην έξοδο εκτυπώνεται πρώτα το 2 καθώς το $11 > 10$, ύστερα το 1 καθώς το $9 < 10$, ακολούθως το 2 καθώς το $12 > 10$, και τέλος το 0 καθώς το $10 = 10$.

3ο

STDIN

```
10
3
10
11
```

STDOUT

```
1
0
```

Εξήγηση παραδείγματος 3:

Η είσοδος περιέχει το μαγικό αριθμό 10 και 3 μαντεψιές, το 3, το 10 και το 11.

Στην έξοδο εκτυπώνεται πρώτα το 1 καθώς το $5 < 10$ και τέλος εκτυπώνεται το 0 καθώς το $10 = 10$.

Δεν εκτύπωνεται άλλος αριθμός, επειδή το πρόγραμμα **τερμάτισε** όταν βρέθηκε ο μαγικός αριθμός.

Η παλιά γραφομηχανή

Τα Χριστούγεννα, όλες οι οικογένειες μαζεύονται για φαγητό και πολλές, μα πολλές, ιστορίες· παλιά κατορθώματα των μεγάλων, ζαβολιές των μικρών και τα κατώματα της Κούλας της γειτόνισσας. Κρίμα δεν είναι να μην καταγραφούν αυτές;

Κάπως έτσι λοιπόν ξεκίνησε ο Τοτός να καταγράφει στα μουλωχτά τις ιστορίες που ακούει. Αλλά δεν ήθελε να βγάλει λάπτοπ δίπλα στην γαλοπούλα, οπότε χρησιμοποίησε ένα απλό σημειωματάριο. Αργότερα, η Αννούλα του ζήτησε να καθαρογράψει τις ιστορίες και με την παλιά γραφομηχανή του παππού!

Η γραφομηχανή παίρνει ένα ιδιαίτερο μέγεθος χαρτιού που χωράει το πολύ K ($20 \leq K \leq 200$) χαρακτήρες σε κάθε γραμμή (θεωρούμε ότι όλοι οι χαρακτήρες, όπως και τα κενά, έχουν το ίδιο μέγεθος). Δυστυχώς, ο παππούς είχε σπάσει το πλήκτρο για την παύλα (-), οπότε δεν μπορείτε να σπάσετε μία λέξη σε 2 μέρη. Ο Τοτός πρέπει λοιπόν να ξαναγράψει τις σημειώσεις, αλλάζοντας το πού σπάτε το κείμενο σε γραμμές, έτσι ώστε να χωράνε όλες οι γραμμές στο χαρτί της γραφομηχανής.

Θέλουμε να γράψουμε ένα πρόγραμμα που θα διαβάζει από την είσοδο `STDIN` τον ακέραιο αριθμό K , το μέγιστο μέγεθος γραμμής στο χαρτί, ακολουθούμενη από γραμμές που περιέχουν λέξεις. Κάθε λέξη έχει μήκος **το πολύ 20** χαρακτήρες.

Το πρόγραμμα θα εκτυπώνει στην έξοδο `STDOUT` κάθε λέξη της εισόδου, με κατάλληλη αλλαγή γραμμών έτσι ώστε κάθε γραμμή να περιέχει το πολύ K χαρακτήρες.

Αν μία γραμμή περιέχει λιγότερους από K χαρακτήρες, τότε το κείμενο θα πρέπει να είναι στοιχισμένο προς τα αριστερά, όπως φαίνεται στο παράδειγμα.

Προσοχή: Η πρώτη γραμμή πρέπει να ξεκινάει με **4 χαρακτήρες κενού** (' '), καθώς συμβολίζει την αρχή παραγράφου.

Υποπροβλήματα

- Η υλοποίηση της παραπάνω εκφώνησης θα βαθμολογηθεί με 70%.
- Για επιπλέον 20%, πρέπει να σπάτε το κείμενο σε διαφορετικούς παραγράφους. Όταν διαβάζετε τη λέξη `PAR`, θα πρέπει **να μην την εκτυπώνετε** και αντ' αυτής να αφήνετε μία **κενή γραμμή**. Κάθε παράγραφος (συμπεριλαμβανομένης και της πρώτης) θα πρέπει να ξεκινάει με **4 κενά**.
- Για επιπλέον 10%, η είσοδος μπορεί να έχει μια λέξη "σπασμένη" σε 2 γραμμές με τη βοήθεια ενός χαρακτήρα παύλας (-): να τυπώσετε στην έξοδο ενωμένες αυτές τις λέξεις.

Παράδειγμα

Είσοδος (STDIN)

```
42
GENTLE READER: This is a handbook about TeX, a new typesetting system
intended for the creation of beautiful books—and especially for books
that contain a lot of mathematics. By preparing a manuscript in TeX format,
you will be telling a computer exactly how the manuscript is to be transformed
into pages whose typographic quality is comparable to that of the world's finest
printers; yet you won't need to do much more work than would be involved if
you were simply typing the manuscript on an ordinary typewriter. In fact, your
total work will probably be significantly less, if you consider the time it ordinarily
takes to revise a typewritten manuscript, since computer text files are so easy
to change and to reprocess. (If such claims sound too good to be true, keep in
mind that they were made by TeX's designer, on a day when TeX happened to
be working, so the statements may be biased; but read on anyway.)
```

Έξοδος (STDOUT)

```
GENTLE READER: This is a handbook
about TeX, a new typesetting system
intended for the creation of beautiful
books—and especially for books that
contain a lot of mathematics. By preparing
a manuscript in TeX format, you will be
telling a computer exactly how the
manuscript is to be transformed into pages
whose typographic quality is comparable to
that of the world's finest printers; yet
you won't need to do much more work than
would be involved if you were simply
typing the manuscript on an ordinary
typewriter. In fact, your total work will
probably be significantly less, if you
consider the time it ordinarily takes to
revise a typewritten manuscript, since
computer text files are so easy to change
and to reprocess. (If such claims sound
too good to be true, keep in mind that
they were made by TeX's designer, on a
day when TeX happened to be working, so
the statements may be biased; but read on
anyway.)
```

Ένα μικρό σκονάκι!

Ανάγνωση **απροσδιόριστου πλήθους λέξεων** από την είσοδο: για να διαβάσετε από την είσοδο STDIN τις λέξεις που ακολουθούν μετά τον αριθμό K, μπορείτε να χρησιμοποιήσετε τους παρακάτω κώδικες:

Python:

```
import sys

for word in sys.stdin.split():
    # Η μεταβλητή word αποθηκεύει 1 νέα λέξη από την είσοδο
```

C/C++ (με `cstdio`):

```
#include <stdio.h>
...
char word[21]; // μέγιστο μήκος λέξης + 1
```

```
while (scanf("%s", word) == 1) {  
    // Η μεταβλητή word αποθηκεύει 1 νέα λέξη από την είσοδο  
}
```

C++ (με `iostream`):

```
#include <iostream>  
...  
string word;  
while (cin >> word) {  
    // Η μεταβλητή word αποθηκεύει 1 νέα λέξη από την είσοδο  
}
```

Για να εκτυπώσετε στην `rython`, χωρίς να αλλάζετε γραμμή ακριβώς μετά, χρησιμοποιείτε την παράμετρο `end= ''` στην `print()`, όπως παρακάτω:

```
print(..., end='')
```

Τώρα σκέφτεσαι με πύλες

Βρισκόμαστε σε έναν ουρανοξύστη και θέλουμε να φτάσουμε στον T-οστό όροφο. Ξεκινάμε από τον S-οστό και μπορούμε να χρησιμοποιήσουμε 2 τρόπους να αλλάξουμε όροφο:

- Μέσω **σκαλών**: κάθε σκάλα μας πηγαίνει από έναν όροφο στον αμέσως από πάνω ή στον αμέσως από κάτω.
- μέσω **πυλών**: κάθε όροφος μας προσφέρει (μέχρι) 2 πύλες. Κάθε πύλη μπορεί να μας πάει σε έναν ψηλότερο όροφο ή σε έναν χαμηλότερο όροφο.
Σε ποιους ορόφους όμως; **Δεν ξέρουμε!** Οι πύλες είναι προκαθορισμένες, αλλά άγνωστες σε εμάς εκ των προτέρων.
Το μόνο που ξέρουμε είναι ότι:
 - αν χρησιμοποιήσουμε την επάνω πύλη του i-οστού ορόφου, δεν υπάρχει καμία μετέπειτα πύλη που φτάνει στον i-οστό όροφο ή παρακάτω από αυτόν και
 - αν χρησιμοποιήσουμε την κάτω πύλη, δεν υπάρχει καμία μετέπειτα που φτάνει στον i-οστό όροφο ή παραπάνω από αυτόν άρα, υπάρχει πιθανότητα ένας όροφος να έχει λιγότερες από 2 πύλες.

Για παράδειγμα, αν ο όροφος 1 έχει πύλη στον όροφο 2 (επάνω πύλη), τότε δεν γίνεται ο όροφος 2 να έχει κάτω πύλη κάπου.

Πρέπει να βρείτε τον καλύτερο τρόπο ώστε να φτάσετε από τον όροφο S στον όροφο T.

Η βέλτιστη λύση δεν χρησιμοποιεί καθόλου σκάλες!

Δεδομένα εισόδου (STDIN) - εξόδου (STDOUT)

Το πρόβλημα είναι διαδραστικό, που σημαίνει ότι δίνετε οδηγίες στον judge και αυτός σας απαντάει με πληροφορίες για να συνεχίσετε.

1. Η πρώτη γραμμή περιέχει 3 ακεραίους F, S και T: το πλήθος των ορόφων, τον όροφο από τον οποίο ξεκινάτε και τον όροφο στον οποίο βρίσκεται η έξοδος. Θεωρείτε πλέον ότι βρίσκεστε στον όροφο S.
2. Όσο δεν βρίσκεστε στον όροφο T:
 - εκτυπώνετε την κίνησή σας: LADDER UP, LADDER DOWN, PORTAL UP, PORTAL DOWN
 - ο judge εκτυπώνει με την σειρά του τον αριθμό x του ορόφου στον οποίο καταλήξατε με την μορφή OK x και
 - αν ζητήσατε κίνηση PORTAL σε όροφο που δεν έχει την αντίστοιχη, ο judge εκτυπώνει SAME
3. Μόλις βρεθείτε στον όροφο T, το παιχνίδι τελειώνει και ο judge υπολογίζει το σκορ σας.

Η βαθμολογία για αυτό το παιχνίδι θα είναι: $100\% - \frac{\text{πλήθος κινήσεων σκάλας}}{\text{συνολικές κινήσεις}}$

Παράδειγμα

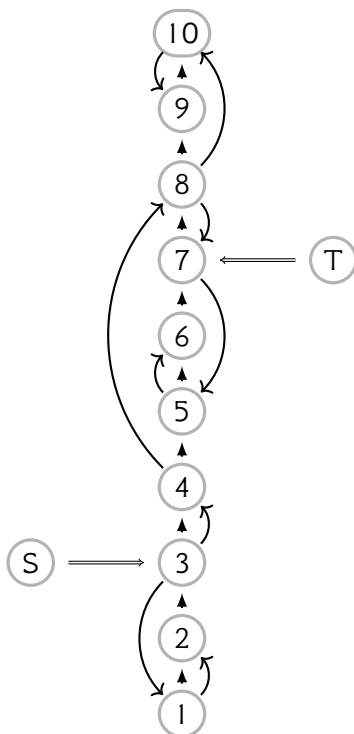
Είσοδος (STDIN)

10 3 7

Μία πιθανή ακολουθία κινήσεων θα μπορούσε να είναι:

Έξοδος προγράμματος (STDOUT)	Έξοδος βαθμολογητή (STDIN)
PORTAL DOWN	OK 1
LADDER UP	OK 2
LADDER UP	OK 3
PORTAL UP	OK 4
PORTAL UP	OK 8
LADDER DOWN	7

και αυτή η λύση θα βαθμολογηθεί με $\frac{73}{7} = 57\%$.



Σημείωση: για να λειτουργεί σωστά ο διαδραστικός grader, πρέπει να κάνετε flush τις γραμμές που εκτυπώνετε. Αυτό γίνεται ως εξής:

- Στην python: η `print()` το κάνει αυτόματα

```
...  
print("LADDER UP")  
...
```

- Στην C/C++ (`stdio`): χρησιμοποιείστε την `printf` κανονικά

```
...  
printf("LADDER UP");  
...
```

- Στην C++ (`iostream`): χρησιμοποιείστε το `endl`

```
...  
cout << "LADDER UP" << endl;  
...
```