



Εθνικό Μετσόβιο Πολυτεχνείο  
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών  
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών

**Αλγόριθμοι και Πολυπλοκότητα**

Διδάσκοντες: Σ. Ζάχος, Δ. Φωτάκης

**4η Σειρά Προγραμματιστικών Ασκήσεων - Ημ/ρία Παράδοσης 6/3/2013**

## Άσκηση 1: Γαλαξιακός Αγώνας Ταχύτητας

Ο Anakin φέτος συμμετέχει στους ετήσιους αγώνες ταχύτητας στον πλανήτη Tatooine και πιστεύει ότι μπορεί εύκολα να κερδίσει, αφού το νέο αγωνιστικό ιπτάμενο όχημα που κατασκεύασε έχει εξαιρετικά υψηλές επιδόσεις. Μοναδική του ανησυχία είναι ότι το νέο όχημα καταναλώνει πάρα πολλά καύσιμα, έτσι πρέπει να ανεφοδιαστεί αρκετές φορές κατά τη διάρκεια του αγώνα για να είναι σίγουρος ότι θα τερματίσει.

Στον πλανήτη Tatooine υπάρχουν συνολικά  $N$  πόλεις και  $M$  δρόμοι διπλής κατεύθυνσης. Κάθε δρόμος  $e$  συνδέει 2 διαφορετικές πόλεις και έχει μήκος  $x_e$  χιλιόμετρα. Η διαδρομή του αγώνα είναι προκαθορισμένη και περνάει από  $K$  διαφορετικές πόλεις, τις  $T_1, T_2, \dots, T_{K-1}, T_K$  με αυτή τη σειρά (είναι ασφαλές να υποθέσουμε ότι μεταξύ κάθε δύο διαδοχικών πόλεων από τις παραπάνω υπάρχει δρόμος). Ο Anakin πρέπει να ακολουθήσει την αγωνιστική διαδρομή, χωρίς παρεκτροπές, και χρειάζεται να ανεφοδιαστεί με καύσιμα τουλάχιστον  $L$  φορές κατά τη διάρκεια του αγώνα. Ο ανεφοδιασμός μπορεί να γίνει σε οποιοσδήποτε  $L$  διαφορετικές πόλεις από τις  $T_2, \dots, T_{K-1}$  (δεν έχει νόημα ο ανεφοδιασμός στην αφετηρία και στον τερματισμό).

Στον πλανήτη Tatooine υπάρχουν βενζινάδικα σε  $B$  πόλεις, τις  $C_1, \dots, C_B$ , και σε καθένα από αυτά υπάρχουν συνεργάτες του Anakin έτοιμοι να κατευθυνθούν προς αυτόν και να τον ανεφοδιάσουν με καύσιμα. Για να ανεφοδιαστεί, ο Anakin πρέπει να σταματήσει σε κάποια πόλη  $T_i$ ,  $2 \leq i \leq K - 1$ , κατά μήκος της αγωνιστικής διαδρομής. Αμέσως μόλις ο Anakin σταματήσει στην πόλη  $T_i$ , ένας συνεργάτης του ξεκινά από το πλησιέστερο βενζινάδικο και κατευθύνεται στην πόλη  $T_i$  για τον ανεφοδιασμό. Έτσι ο Anakin καθυστερεί στην  $T_i$  ακριβώς τόσο χρόνο όσο χρειάζεται ο συνεργάτης του για να φτάσει από το πλησιέστερο βενζινάδικο, αφού υποθέτουμε ότι ο χρόνος για τον ανεφοδιασμό είναι αμελητέος. Αμέσως μετά τον ανεφοδιασμό, ο Anakin συνεχίζει την πορεία του προς την επόμενη πόλη  $T_{i+1}$ .

Ο Anakin σας ζητάει να γράψετε ένα πρόγραμμα για τον υπολογισμό του ελάχιστου χρόνου που χρειάζεται για να ολοκληρώσει τον αγώνα. Για τον υπολογισμό του ελάχιστου χρόνου υποθέτουμε ότι η ταχύτητα με την οποία κινούνται όλα τα οχήματα (τόσο του Anakin όσο και των συνεργατών του) είναι 1 χιλιόμετρο το δευτερόλεπτο, ότι τα οχήματα των συνεργατών του μπορούν να χρησιμοποιήσουν όποιον δρόμο θέλουν, ακόμα και αυτούς της αγωνιστικής διαδρομής, και ότι οι συνεργάτες του ξεκινούν πάντα από το πλησιέστερο βενζινάδικο (που μπορεί να είναι το ίδιο για δύο ή περισσότερα σημεία ανεφοδιασμού).

**Δεδομένα Εισόδου:** Το πρόγραμμά σας θα διαβάζει από το standard input, στην πρώτη γραμμή, 5 φυσικούς αριθμούς που αντιστοιχούν στο πλήθος  $N$  των πόλεων, στο πλήθος  $M$  των δρόμων, στο πλήθος  $K$  των πόλεων στην αγωνιστική διαδρομή, στο πλήθος  $L$  των απαιτούμενων ανεφοδιασμών, και στο πλήθος  $B$  των βενζινάδικων. Οι πόλεις είναι αριθμημένες από 1 έως  $N$ . Στην  $i$ -οστή από τις επόμενες  $M$  γραμμές θα υπάρχουν 3 φυσικοί αριθμοί  $u_i, v_i, x_i$  που δηλώνουν ότι ο  $i$ -οστός δρόμος ενώνει τις πόλεις  $u_i$  και  $v_i$  και έχει μήκος  $x_i$ . Στις επόμενες  $K$  γραμμές θα υπάρχουν  $K$  διαφορετικοί φυσικοί αριθμοί  $T_1, \dots, T_K$  που δηλώνουν τις πόλεις από τις οποίες διέρχεται η αγωνιστική διαδρομή.

Τέλος, στις επόμενες  $B$  γραμμές θα υπάρχουν  $B$  φυσικοί αριθμοί  $C_1, \dots, C_B$  που δηλώνουν τις πόλεις όπου βρίσκονται βενζινάδικα.

**Δεδομένα Εξόδου:** Το πρόγραμμά σας πρέπει να τυπώνει στο standard output (στην πρώτη γραμμή) έναν ακέραιο που αντιστοιχεί στον ελάχιστο χρόνο σε δευτερόλεπτα που χρειάζεται ο Ανάκιν για να ολοκληρώσει τον αγώνα. Σημειώστε ότι για μεγάλες τιμές των  $N$  και  $M$ , ο ελάχιστος χρόνος (καθώς και κάποια από τα ενδιάμεσα αποτελέσματα που χρειάζονται για τον υπολογισμό του) μπορεί να υπερβαίνουν το  $2^{32}$ .

Περιορισμοί:	Παράδειγμα Εισόδου:	Παράδειγμα Εξόδου:
$3 \leq N \leq 60000$	5 8 4 2 1	19
$3 \leq M \leq 10^6$	1 2 4	
$1 \leq u_i, v_i, T_i, C_i \leq N$	1 3 1	
$3 \leq K \leq N$	1 4 2	
$1 \leq B \leq N$	2 4 3	
$1 \leq L \leq K - 2$	3 4 5	
$1 \leq x(i) \leq 20000$	5 1 2	
Όριο χρόνου εκτέλεσης: 1 sec.	5 3 1	
Όριο μνήμης: 64 MB.	5 2 3	
	1	
	2	
	4	
	3	
	5	

## Άσκηση 2: Κύπελλο Ποδοσφαίρου

Στην εξωτική Παραγκούπολη, οι αγώνες του κυπέλλου ποδοσφαίρου λαμβάνουν χώρα ακολουθιακά, ο ένας μετά τον άλλο (και όχι σε γύρους, όπως συνήθως). Αρχικά, έχουμε  $N$  ομάδες που συμμετέχουν στην διοργάνωση, και μετά από κάθε αγώνα, η ηττημένη ομάδα βγαίνει εκτός διοργάνωσης, ενώ η νικήτρια παραμένει (δεν υπάρχουν ισοπαλίες). Η διοργάνωση ολοκληρώνεται όταν έχει μείνει μόνο μία ομάδα, η οποία κερδίζει το κύπελλο. Έχετε αναλάβει να καθορίσετε την ακολουθία των αγώνων επιλέγοντας ημερομηνίες και αντιπάλους.

Είναι εύκολο να δει κανείς ότι το αποτέλεσμα δεν εξαρτάται μόνο από τις επιδόσεις και την προετοιμασία των ομάδων, αλλά και από την “τύχη”, δηλαδή το πρόγραμμα των αγώνων. Κι εσείς το γνωρίζετε πολύ καλά αυτό. Έχετε περάσει αρκετό καιρό βλέποντας προσεκτικά τις επιδόσεις κάθε ομάδας. Είναι πλέον προφανές ότι τα αποτελέσματα κάποιων αγώνων είναι 100% προβλέψιμα. Έχοντας αυτές τις πληροφορίες, θέλετε να δείτε αν μπορείτε να οργανώσετε το πρόγραμμα των αγώνων με τρόπο ώστε μια ομάδα  $X$  να κερδίσει το κύπελλο. Θέλετε δηλαδή να οργανώσετε τους αγώνες έτσι ώστε η ομάδα  $X$  να παίξει μόνο με αντιπάλους που σίγουρα θα κερδίσει (ώστε τελικά να κερδίσει το κύπελλο). Αν αυτό είναι δυνατό, λέμε ότι το κύπελλο μπορεί να “στηθεί” για λογαριασμό της ομάδας  $X$ . Θέλετε λοιπόν να γράψετε ένα πρόγραμμα που υπολογίζει για ποιες ομάδες μπορεί να “στηθεί” το κύπελλο.

**Δεδομένα Εισόδου:** Το πρόγραμμά σας θα διαβάζει από το standard input ένα κατευθυνόμενο γράφημα που αναπαριστά τα ζεύγη των ομάδων για τις οποίες το μεταξύ τους αποτέλεσμα είναι 100% προβλέψιμο. Στην πρώτη γραμμή, θα δίνεται το πλήθος  $N$  των ομάδων, και θα ακολουθούν  $N$  γραμμές. Στην  $i$ -οστή γραμμή, ο πρώτος αριθμός  $d_i$  θα δηλώνει από πόσες ομάδες χάνει σίγουρα η ομάδα

$i$  (μπορεί να είναι 0), και θα ακολουθούν  $d_i$  αριθμοί, που θα δηλώνουν ποιες είναι αυτές οι ομάδες. Π.χ. στο παράδειγμα, έχουμε 5 ομάδες, η 1η χάνει από μία ομάδα, την 5, η 2η χάνει από τρεις ομάδες, τις 1, 4, και 5, η 3η χάνει από δύο ομάδες, τις 1 και 4, η 4η χάνει από μία ομάδα, την 1, και η 5η χάνει από μία ομάδα, την 3.

**Παρατηρήσεις:** Τα δεδομένα για τη σχέση των αποτελεσμάτων που είναι προβλέψιμα θα περιγράφουν μια σχέση που είναι αντισυμμετρική και δεν είναι κατ' ανάγκη ούτε πλήρης ούτε μεταβατική. Δηλαδή, όσον αφορά στην αντισυμμετρικότητα, αν υπάρχει μια ομάδα  $a$  χάνει από μια ομάδα  $b$ , η  $b$  δεν θα χάνει από την  $a$ . Όσον αφορά στην πληρότητα, μπορεί να υπάρχουν ζευγάρια για τα οποία δεν γνωρίζουμε με σιγουριά ποια ομάδα χάνει από την άλλη. Όσον αφορά στην μεταβατικότητα, μπορεί να γνωρίζουμε ότι μια ομάδα  $a$  χάνει από μια ομάδα  $b$ , και ότι η  $b$  χάνει από μια ομάδα  $c$ , και να μην γνωρίζουμε με σιγουριά αν η  $a$  χάνει από την  $c$  (ή μπορεί ακόμη και να γνωρίζουμε ότι η  $c$  χάνει από την  $a$ ).

**Δεδομένα Εξόδου:** Το πρόγραμμά σας πρέπει να τυπώνει στο standard output (στην πρώτη γραμμή) έναν ακέραιο που αντιστοιχεί στο πλήθος των ομάδων για τις οποίες μπορεί να “στηθεί” η διοργάνωση του κυπέλλου.

**Περιορισμοί:**

$$3 \leq N \leq 30000$$

$$3 \leq M \leq 10^6$$

Όριο χρόνου εκτέλεσης: 1 sec.

Όριο μνήμης: 64 MB.

**Παράδειγμα Εισόδου:**

```
5
1 5
3 1 4 5
2 1 4
1 1
1 3
```

**Παράδειγμα Εξόδου:**

```
4
```