



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

Αλγόριθμοι και Πολυπλοκότητα

Διδάσκοντες: Σ. Ζάχος, Δ. Φωτάκης

3η Σειρά Προγραμματιστικών Ασκήσεων - Ημ/νία Παράδοσης 17/2/2014

Άσκηση 1: Μετακινήσεις

Στη μακρινή Χώρα των Αλγορίθμων υπάρχουν N πόλεις που συνδέονται μεταξύ τους με M λεωφόρους διπλής κατεύθυνσης. Κάθε λεωφόρος συνδέει δύο πόλεις. Μάλιστα η μετακίνηση στις λεωφόρους είναι τόσο γρήγορη που μπορούμε να θεωρήσουμε ότι γίνεται στιγμιαία! Έτσι είναι συνηθισμένο οι κάτοικοι να διαμένουν σε διαφορετική πόλη από αυτή στην οποία εργάζονται.

Οι λεωφόροι, δυστυχώς, είναι παλιές και χρειάζονται καθημερινή συντήρηση, με αποτέλεσμα να κλείνουν από μία χρονική στιγμή και μετά. Ξέρουμε ότι η i -οστή λεωφόρος κλείνει για συντήρηση τη χρονική στιγμή $d_i > 0$, και συνεπώς μπορεί να χρησιμοποιηθεί για μετακινήσεις μόνο κατά τις χρονικές στιγμές $t \in [0, d_i]$. Βέβαια οι λεωφόροι έχουν σχεδιαστεί ώστε τη χρονική στιγμή 0, όλοι να μπορούν να μεταβούν στην εργασία τους.

Οι εκλογές πλησιάζουν, και ο Πρόεδρος της χώρας σας έχει αναθέσει να βελτιώσετε τις συγκοινωνίες (και τη ζωή των ψηφοφόρων). Το χρήματα που έχετε στη διάθεσή σας επαρκούν για K οδικά έργα. Κάθε οδικό έργο περιλαμβάνει είτε την πλήρη ανακατασκευή μιας υπάρχουσας λεωφόρου είτε τη δημιουργία μιας νέας. Μετά από αυτό, η αντίστοιχη λεωφόρος δεν θα χρειάζεται πλέον συντήρηση, και θα μπορεί να χρησιμοποιηθεί για μετακινήσεις οποιαδήποτε χρονική στιγμή. Στόχος αυτών των έργων είναι να μεγιστοποιήσουμε την τελευταία χρονική στιγμή T^* όπου εξασφαλίζεται ότι όλοι μπορούν να μεταβούν στην εργασία τους (ή ισοδύναμα, εξασφαλίζεται ότι η συνδεσιμότητα της χώρας είναι αντίστοιχη με τη συνδεσιμότητα κατά τη χρονική στιγμή 0). Αυτό είναι σημαντικό γιατί οι κάτοικοι δεν θα χρειάζεται να ξεκινήσουν για την εργασία τους πριν τη χρονική στιγμή T^* , και έτσι θα έχουν περισσότερο χρόνο για να ξεκουραστούν. Πρέπει λοιπόν να υπολογίσετε τη μέγιστη χρονική στιγμή T^* που μπορεί να επιτευχθεί με K οδικά έργα. **Σημείωση:** Προσέξτε ότι δεν γνωρίζουμε ποιος δουλεύει πού. Γνωρίζουμε μόνο ότι τη χρονική στιγμή 0, ο καθένας μπορεί να μεταβεί από την κατοικία στην εργασία του. Αυτό δεν σημαίνει απαραίτητα ότι τη χρονική στιγμή 0 μπορούμε να μεταβούμε από κάθε πόλη σε οποιαδήποτε άλλη!

Δεδομένα Εισόδου: Το πρόγραμμά σας θα διαβάζει από το standard input τρεις θετικούς ακέραιους N , M , και K , που αντιστοιχούν στο πλήθος των πόλεων, στο πλήθος των λεωφόρων, και στο πλήθος των οδικών έργων. Σε κάθε μία από τις επόμενες M γραμμές, θα δίνονται τρεις θετικοί ακέραιοι a_i , b_i , και t_i που δηλώνουν ότι υπάρχει μια λεωφόρος που συνδέει τις πόλεις a_i και b_i και κλείνει για συντήρηση τη χρονική στιγμή t_i . Μεταξύ κάθε ζεύγους πόλεων, θα υπάρχει το πολύ μία λεωφόρος που τις συνδέει.

Δεδομένα Εξόδου: Το πρόγραμμα σας πρέπει να τυπώνει στο standard output (στην πρώτη γραμμή) έναν θετικό ακέραιο, τη μέγιστη χρονική στιγμή T^* όπου εξασφαλίζεται ότι όλοι μπορούν να μεταβούν στην εργασία τους, αν κάνουμε τα οδικά έργα με βέλτιστο τρόπο. Αν τα οδικά έργα μπορούν να εξασφαλίσουν απεριόριστο χρόνο ξεκούρασης για τους κατοίκους, το πρόγραμμα πρέπει να τυπώνει μόνο μία γραμμή με τη λέξη “infinity”.

Περιορισμοί:	Παράδειγμα Εισόδου:	Παράδειγμα Εξόδου:
$1 \leq N, M, K \leq 3 \cdot 10^5$	4 5 1	8
$1 \leq t_i \leq 2 \cdot 10^9$	4 2 4	
Όριο χρόνου εκτέλεσης: 1 sec.	3 4 10	
Όριο μνήμης: 64 MB.	4 1 5	
	2 1 3	
	3 2 8	

Άσκηση 2: Τηλεπικοινωνιακά Κόστη

Μια τηλεπικοινωνιακή εταιρεία προσπαθεί να προσελκύσει έναν νέο πελάτη. Ο πελάτης ενδιαφέρεται για την ενοικίαση οπτικών ινών που εξασφαλίζουν τη διασύνδεση πόλεων. Η εταιρεία πρέπει να καταθέσει προσφορά με το κόστος c_{ij} για την απευθείας σύνδεση κάθε ζεύγους πόλεων i και j , $1 \leq i < j \leq n$. Με βάση την προσφορά, ο πελάτης θα επιλέξει ένα Ελάχιστο Συνδετικό Δέντρο, το οποίο καλύπτει όλες τις πόλεις και εξασφαλίζει ελάχιστο συνολικό κόστος σύνδεσης.

Η εταιρεία έχει σημαντικούς λόγους να χειραγωγήσει την επιλογή του πελάτη, οδηγώντας τον στην επιλογή ενός συγκεκριμένου συνδετικού δέντρου T για τη διασύνδεση των πόλεων. Με δεδομένο το κόστος των συνδέσεων που εντάσσονται στο T , η εταιρεία επιθυμεί να προσαρμόσει το κόστος των υπόλοιπων συνδέσεων, ώστε το δέντρο T να προκύπτει ως το μοναδικό Ελάχιστο Συνδετικό Δέντρο που καλύπτει όλες τις πόλεις. Όμως για να μην φανεί στον πελάτη ότι η προσφορά είναι ακριβή, η εταιρεία επιθυμεί το συνολικό προτεινόμενο κόστος κόστους για όλα τα ζεύγη πόλεων να είναι το ελάχιστο δυνατό.

Να γράψετε ένα πρόγραμμα που υπολογίζει την προσφορά της εταιρείας με βάση τους παραπάνω περιορισμούς. Για τη διαμόρφωση της προσφοράς, να θεωρήσετε ότι όλα τα κόστη c_{ij} πρέπει να είναι φυσικοί αριθμοί.

Δεδομένα Εισόδου: Αρχικά, το πρόγραμμά σας θα διαβάξει από το standard input έναν θετικό ακέραιο N που αντιστοιχεί στο πλήθος των πόλεων. Οι πόλεις αριθμούνται από το 1 έως το N . Σε κάθε μία από τις επόμενες $N - 1$ γραμμές, θα υπάρχουν τρεις θετικοί ακέραιοι i , j και c_{ij} χωρισμένοι με κενό. Αυτοί δηλώνουν ότι η πόλη i συνδέεται απευθείας με την πόλη j στο επιλεγμένο δέντρο T και με κόστος c_{ij} .

Δεδομένα Εξόδου: Το πρόγραμμά σας πρέπει να τυπώνει στο standard output (στην πρώτη γραμμή) το ελάχιστο συνολικό προτεινόμενο κόστος για όλα τα ζεύγη πόλεων, ώστε το επιλεγμένο δέντρο T να αποτελεί το μοναδικό Ελάχιστο Συνδετικό Δέντρο. Σημειώστε ότι για μεγάλες τιμές του N , το ελάχιστο συνολικό βάρος ακμών του G (καθώς και κάποια από τα ενδιάμεσα αποτελέσματα που χρειάζονται για τον υπολογισμό του) μπορεί να υπερβαίνουν το 2^{32} .

Περιορισμοί:	Παράδειγμα Εισόδου:	Παράδειγμα Εξόδου:
$3 \leq N \leq 5 \cdot 10^5$	3	19
$1 \leq i, j \leq N$	1 2 4	
$1 \leq c_{ij} \leq 10^4$	2 3 7	
Όριο χρόνου εκτέλεσης: 1 sec.		
Όριο μνήμης: 64 MB.	4	12
	1 2 1	
	1 3 1	
	1 4 2	