



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

Αλγόριθμοι και Πολυπλοκότητα

Διδάσκοντες: Δημήτρης Φωτάκης, Δώρα Σούλιου, Θανάσης Λιανέας

1η Σειρά Προγραμματιστικών Ασκήσεων - Ημ/νία Παράδοσης 17/11/2020

Άσκηση 1: Ενοικίαση Αυτοκινήτου

Πρόκειται να ενοικιάσουμε αυτοκίνητο και να ταξιδέψουμε οδικώς από την πόλη της Ταξινόμησης στην πόλη της Επιλογής για το γενέθλιο πάρτυ του αλγόριθμου Δυαδικής Αναζήτησης. Ο δρόμος που συνδέει τις πόλεις της Ταξινόμησης και της Επιλογής είναι μια μεγάλη ευθεία μήκους D χιλιομέτρων. Μεταξύ των πόλεων της Ταξινόμησης και της Επιλογής, υπάρχουν K ενδιάμεσοι σταθμοί (μερικοί μπορεί να βρίσκονται πολύ κοντά μεταξύ τους) οι οποίοι αποτελούν και τα μοναδικά σημεία όπου μπορεί κάποιος να ανεφοδιάσει το αυτοκίνητό του με καύσιμα. Συγκεκριμένα, ο σταθμός j βρίσκεται σε απόσταση d_j χιλιόμετρα από την πόλη της Επιλογής.

Ο Δημήτρης έφτασε καθυστερημένος στο γραφείο ενοικίασης αυτοκινήτων και σχεδιάζει να ταξίδι του για την πόλη της Επιλογής, ώστε να προλάβει την έναρξη του πάρτυ, σε T λεπτά ακριβώς. Για το ταξίδι του, θα νοικιάσει ένα από τα N αυτοκίνητα που είναι διαθέσιμα. Κάθε αυτοκίνητο i χαρακτηρίζεται από το κόστος ενοικίασης p_i και την χωρητικότητά του c_i σε καύσιμα. Κατά τα άλλα, τα αυτοκίνητα είναι τυποποιημένα. Όλα διαθέτουν δύο ρυθμούς λειτουργίας, τον *οικονομικό*, με τον οποίο καλύπτουν ένα χιλιόμετρο σε T_s λεπτά και καταναλώνουν C_s λίτρα καυσίμου το χιλιόμετρο, και τον *σπορ*, με τον οποίο καλύπτουν ένα χιλιόμετρο σε $T_f < T_s$ λεπτά και καταναλώνουν $C_f > C_s$ λίτρα καυσίμου το χιλιόμετρο. Ευτυχώς, τουλάχιστον, ο χρόνος εναλλαγής από τον ένα ρυθμό λειτουργίας στον άλλο είναι μηδενικός και ο οδηγός μπορεί να αλλάξει όσες φορές θέλει τη λειτουργία του οχήματός του. Ο Δημήτρης αγαπάει τη σπορ οδήγηση και βιάζεται να φτάσει στον προορισμό του, αλλά θέλει να είναι σίγουρος ότι δεν θα μείνει από καύσιμα στη μέση της διαδρομής.

Ο Δημήτρης θέλει να γράψει ένα πρόγραμμα που υπολογίζει το ελάχιστο κόστος ενοικίασης αυτοκινήτου που μπορεί να καλύψει την απόσταση από την πόλη της Ταξινόμησης στην πόλη της Επιλογής σε χρόνο όχι μεγαλύτερο από T λεπτά.

Λεδομένα Εξόδου: Το πρόγραμμα αρχικά θα διαβάζει από το standard input τέσσερις θετικούς ακεραίους, το πλήθος N των διαθέσιμων αυτοκινήτων, το πλήθος K των ενδιάμεσων σταθμών, την απόσταση D των πόλεων της Ταξινόμησης και της Επιλογής σε χιλιόμετρα, και το χρονικό περιθώριο T που έχει στη διάθεσή του ο Δημήτρης για να ολοκληρώσει το ταξίδι του. Σε καθεμία από τις επόμενες N γραμμές, θα υπάρχουν δύο θετικοί ακέραιοι p_i και c_i που αντιστοιχούν στο κόστος ενοικίασης και την χωρητικότητα του αυτοκινήτου i (η αρίθμηση των αυτοκινήτων είναι αυθαίρετη, ενώ μπορεί να υπάρχουν αυτοκίνητα i και j με $p_i > p_j$ και $c_i < c_j$). Στην επόμενη γραμμή, θα υπάρχουν K θετικοί ακέραιοι αριθμοί d_1, \dots, d_K που αντιστοιχούν στις αποστάσεις (σε χιλιόμετρα) των ενδιάμεσων σταθμών από την αρχική πόλη της Ταξινόμησης. Η αρίθμηση των ενδιάμεσων σταθμών είναι αυθαίρετη (δηλ. οι ενδιάμεσοι σταθμοί δεν έχουν αριθμηθεί απαραίτητα σε αύξουσα ή φθίνουσα απόσταση από την πόλη της Επιλογής), ενώ μπορεί να υπάρχουν σταθμοί $i \neq j$ με $d_i = d_j$. Στην τελευταία γραμμή της εισόδου, θα υπάρχουν τέσσερις θετικοί ακέραιοι T_s, C_s, T_f, C_f που αντιστοιχούν στην χρονική απόδοση και την κατανάλωση του οικονομικού και του σπορ ρυθμού λειτουργίας αντίστοιχα.

Λεδομένα Εξόδου: Το πρόγραμμα πρέπει να τυπώνει στο standard output έναν ακέραιο, το ελάχιστο κόστος ενοικίασης αυτοκινήτου που μπορεί να καλύψει την απόσταση από την πόλη της Ταξινόμησης

στην πόλη της Επιλογής σε χρόνο όχι μεγαλύτερο από T λεπτά. Στην περίπτωση που δεν υπάρχει τέτοιο αυτοκίνητο, το πρόγραμμα πρέπει να τυπώνει -1 .

Περιορισμοί:	Παράδειγμα Εισόδου:	Παράδειγμα Εξόδου:
$1 \leq N \leq 4 \cdot 10^5$	3 1 8 10	10
$1 \leq K \leq 2 \cdot 10^5$	10 8	
$1 \leq D, p_i, c_i \leq 10^9$	5 7	
$1 \leq T \leq 2 \cdot 10^9$	11 9	
$0 < d_j < D$	3	
$1 \leq T_f < T_s \leq 2 \cdot 10^9$	2 1 1 2	
$1 \leq C_s < C_f \leq 10^9$		
Όριο χρόνου εκτέλεσης: 1 sec.		
Όριο μνήμης: 64MB.		

Άσκηση 1: Τηλεμεταφορές!

Όπως ίσως γνωρίζετε, υπάρχουν άπειρα παράλληλα σύμπαντα. Σε N από αυτά, ο Morty, ο εγγονός του Rick, έκλεψε το portal gun του παππού του για να εντυπωσιάσει την Jessica. Όμως, δίχως να το καταλάβει, τηλεμεταφέρθηκε σε ένα άλλο παράλληλο σύμπαν! Για την ακρίβεια, γνωρίζουμε ότι ο M_i , δηλ. ο Morty του σύμπαντος i , τηλεμεταφέρθηκε στο σύμπαν $c_i \in \{1, \dots, N\}$. Μετά από όλες αυτές τις ατυχείς τηλεμεταφορές, συνεχίζουμε να έχουμε έναν και μόνο Morty σε καθένα από τα N παράλληλα σύμπαντα (δηλ. η ακολουθία $c = (c_1, \dots, c_N)$ αποτελεί μια μετάθεση των στοιχείων του $\{1, \dots, N\}$). Για καλή τους τύχη, έμειναν ανοιχτά M portals μεταξύ μερικών συμπάντων τα οποία δύο Mortys μπορούν να χρησιμοποιήσουν για να ανταλλάξουν σύμπαντα. Κάθε ανοιχτό portal j συνδέει δύο παράλληλα σύμπαντα a_j και b_j , έχει πλάτος w_j , και μπορεί να χρησιμοποιηθεί απεριόριστες φορές. Οι Mortys πρέπει να επιστρέψουν στα σωστά σύμπαντα (δηλ. να επαναφέρουν την ακολουθία c στην αρχική ταυτοτική της μορφή) πριν το καταλάβουν οι Ricks! Οι Mortys είναι γκρινιάρηδες σε όλα τα N σύμπαντα, και δεν θέλουν να στρωμαχτούν σε ανοιχτά portals που είναι στενά.

Να γράψετε ένα πρόγραμμα που βοηθάει τους Mortys να επιστρέψουν άνετα στα (σωστά) σύμπαντά τους, δηλ. να επαναφέρουν την ακολουθία c στην αρχική ταυτοτική της μορφή, υπολογίζοντας το μέγιστο πλάτος του στενότερου portal που χρειάζεται να χρησιμοποιήσουν για αυτό τον σκοπό.

Λεδομένα Εισόδου: Το πρόγραμμά σας αρχικά θα διαβάζει από το standard input δύο θετικούς ακεραίους, το πλήθος N των παράλληλων συμπάντων που μας ενδιαφέρουν και το πλήθος M των portals που έχουν μείνει ανοιχτά. Στην επόμενη γραμμή, δίνεται μια μετάθεση $c = (c_1, \dots, c_N)$ του συνόλου $\{1, \dots, N\}$, στην οποία το στοιχείο c_i δηλώνει το παράλληλο σύμπαν στο οποίο κατέληξε ο Morty του i -οστού σύμπαντος. Στις M επόμενες γραμμές, δίνονται M τριάδες φυσικών αριθμών που περιγράφουν τα portals που έχουν μείνει ανοιχτά. Συγκεκριμένα, κάθε ανοιχτό portal j περιγράφεται από τρεις φυσικούς αριθμούς, τα σύμπαντα a_j και b_j που ενώνει, και το πλάτος του w_j .

Λεδομένα Εξόδου: Το πρόγραμμά σας πρέπει να τυπώνει στο standard output ένα φυσικό αριθμό, που εκφράζει το μέγιστο πλάτος του στενότερου portal που χρειάζεται να χρησιμοποιηθεί, ώστε η ακολουθία εισόδου $c = (c_1, \dots, c_N)$ να επανέλθει στην αρχική ταυτοτική της μορφή¹. Σε όλα

¹ **Εξήγηση Παραδείγματος:** Ένας τρόπος να επιστρέψουν όλοι οι Mortys στα σωστά σύμπαντα χρησιμοποιώντας τα portals πλάτους τουλάχιστον 73 είναι: Οι M_1 και M_2 ανταλλάσσουν σύμπαντα χρησιμοποιώντας το portal 4 (πλάτους 100). Οι M_1 και M_3 ανταλλάσσουν σύμπαντα χρησιμοποιώντας το portal 1 (πλάτους 73). Τέλος, οι M_2 και M_3 ανταλλάσσουν σύμπαντα χρησιμοποιώντας το portal 4. Δείτε ότι δεν είναι δυνατόν να επιστρέψουν όλοι οι Mortys στα σωστά σύμπαντα, αν χρησιμοποιήσουμε μόνο το portal 4, αφού σε αυτή την περίπτωση δεν είναι δυνατή η επιστροφή του M_1 στο σύμπαν 1 (από το σύμπαν 3 όπου βρίσκεται) και του M_3 στο σύμπαν 3 (από το σύμπαν 1 όπου βρίσκεται).

τα στιγμιότυπα εισόδου, θα είναι εφικτό να επανέλθει η ακολουθία εισόδου $c = (c_1, \dots, c_N)$ σε ταυτοτική μορφή.

Περιορισμοί:

$$1 \leq N \leq 10^5$$

$$1 \leq M \leq 10^5$$

$$1 \leq a_j \neq b_j \leq N$$

$$1 \leq w_j \leq 10^9$$

Όριο χρόνου εκτέλεσης: 1 sec.

Όριο μνήμης: 64MB.

Παράδειγμα Εισόδου:

4 4

3 2 1 4

1 2 7 3

1 3 4 2

2 4 1 7

2 3 1 0 0

Παράδειγμα Εξόδου:

7 3