



Εθνικό Μετσόβιο Πολυτεχνείο

Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών

Αλγόριθμοι και Πολυπλοκότητα

Διδάσκοντες: Άρ. Παγουρτζής, Δ. Φωτάκης, Δ. Σούλιου, Π. Γροντάς

2η Σειρά Προγραμματιστικών Ασκήσεων - Ημ/νία Παράδοσης 3/1/2024

Άσκηση 1: Εμπόριο Σοκολάτας

Ο Θεός Σκρουτζ χρειάζεται να αυξήσει τα έσοδά του, ενόψει των εορτών των Χριστουγέννων και των δώρων που θα αγοράσει για τα ανήψια του! Έτσι αποφάσισε να ασχοληθεί (και) με το εμπόριο σοκολάτας, το οποίο πιστεύει ότι μπορεί να αποβεί ιδιαίτερα επικερδές αυτή την εποχή του χρόνου.

Ο Θεός Σκρουτζ συγκέντρωσε πληροφορίες και έχει καταφέρει να προβλέψει με ακρίβεια την τιμή της σοκολάτας (ανά τόνο, στην αγορά εμπορευμάτων) για καθεμία από τις N ημέρες που απομένουν μέχρι τα Χριστούγεννα. Έχει καταγράψει την ακολουθία τιμών $p(1), \dots, p(N)$ και θέλει να βρει πότε πρέπει να αγοράσει και πότε πρέπει να πουλήσει ώστε να μεγιστοποιήσει τα κέρδη του. Κάθε φορά θα αγοράζει και θα πουλάει την ίδια ποσότητα σοκολάτας, οπότε ενδιαφέρεται να μεγιστοποιήσει το κέρδος ανά τόνο. Επιπλέον, προσπαθώντας να ελαχιστοποιήσει τα έξοδα και τις προμήθειες και να κρατήσει την εμπλοκή του στην αγορά σοκολάτας διακριτική, έχει αποφασίσει ότι οι αγορές και οι πωλήσεις θα εναλλάσσονται και ότι δεν θα πραγματοποιήσει περισσότερες από K αγοραπωλησίες μέχρι τα Χριστούγεννα. Χρειάζεται λοιπόν να υπολογίσει το βέλτιστο πλήθος αγορών (και πωλήσεων) M , όπου $0 \leq M \leq K$, τις M ημέρες b_1, \dots, b_M που θα αγοράσει σοκολάτα, και τις M ημέρες s_1, \dots, s_M που θα πουλήσει. Για τις επιλεγμένες ημέρες θα ισχύει ότι $1 \leq b_1 < s_1 < b_2 < s_2 < \dots < b_M < s_M \leq N$. Στόχος του Θείου Σκρουτζ είναι να μεγιστοποιήσει το συνολικό του κέρδος ανά τόνο σοκολάτας, που είναι ίσο με $\sum_{i=1}^M (p(s_i) - p(b_i))$ (ή ίσο με 0, αν δεν πραγματοποιήσει καμία αγοραπωλησία).

Προσβλέποντας σε ένα γενναίο Χριστουγεννιάτικο μπόνους, προσφέρεστε να βοηθήσετε τον Θείο Σκρουτζ στη νέα του επιχειρηματική δραστηριότητα. Πρέπει λοιπόν να γράψετε ένα πρόγραμμα που θα υπολογίζει το μέγιστο κέρδος (ανά τόνο σοκολάτας) που μπορεί να έχει ο Θεός Σκρουτζ αν πραγματοποιήσει το πολύ K αγοραπωλησίες.

Λεδομένα Εισόδου: Αρχικά, το πρόγραμμα θα διαβάσει από το standard input δύο θετικούς ακέραιους αριθμούς που αντιστοιχούν στο πλήθος των ημερών N και στο μέγιστο πλήθος αγοραπωλησιών K . Στην επόμενη γραμμή, θα υπάρχουν N φυσικοί αριθμοί (χωρισμένοι με κενό) που αντιστοιχούν στις τιμές $p(1), \dots, p(N)$ ενός τόνου σοκολάτας για καθεμία από τις επόμενες N ημέρες. Για το 60% της βαθμολογίας, θα ισχύει ότι $N \leq 1.000$ και $K \leq 100$.

Λεδομένα Εξόδου: Το πρόγραμμα πρέπει να τυπώνει στο standard output (στην πρώτη γραμμή) το μέγιστο κέρδος (ανά τόνο σοκολάτας) που μπορεί να επιτύχει ο Θεός Σκρουτζ αν πραγματοποιήσει το πολύ K αγοραπωλησίες.

Περιορισμοί:

Παραδείγματα Εισόδου:

Παράδειγμα Εξόδου:

$1 \leq N \leq 10^3$ (για το 60%)

10 3

13

$1 \leq K \leq 10^2$ (για το 60%)

12 12 7 10 15 8 3 4 8 8

$1 \leq N \leq 10^5$ (για το 100%)

$1 \leq K \leq 10^3$ (για το 100%)

5 2

0

$1 \leq p(i) \leq 10^4$

10 9 8 7 6

Όριο χρόνου εκτέλεσης: 1 sec.

Όριο μνήμης: 64 MB.

Άσκηση 2: Χριστουγεννιάτικη Σκυταλοδρομία

Ο Αι Βασίλης μόλις διαπίστωσε ότι δεν έχουν φτάσει ακόμη στο Ροβανιέμι τα γράμματα με τις επιθυμίες των παιδιών από τη χώρα των Αλγορίθμων. Οι βοηθοί του πρέπει να φροντίσουν ώστε οι επιθυμίες των παιδιών να φτάσουν άμεσα στα χέρια του, για να προλάβει να έχει έτοιμα τα δώρα τους για τα Χριστούγεννα.

Η χώρα των Αλγορίθμων έχει N μεγάλες πόλεις V_1, \dots, V_N στις οποίες έχουν συγκεντρωθεί τα γράμματα των παιδιών από τις κοντινές περιοχές. Το οδικό δίκτυο της χώρας είναι ιδιαίτερα απλό και έχει δενδρική δομή. Κάθε πόλη V_i έχει μοναδική διαδρομή προς την πρωτεύουσα V_1 , και οι βοηθοί του Αι Βασίλη έχουν υπολογίσει ακριβώς την απόσταση D_i της V_i από την επόμενη πόλη στη διαδρομή προς την πρωτεύουσα V_1 . Σε κάθε πόλη V_i , βρίσκεται ένας βοηθός του Αι Βασίλη που χρειάζεται P_i δευτερόλεπτα για να φορτώσει τα γράμματα και να ξεκινήσει με το έλκυθρό του και S_i δευτερόλεπτα για να διανύσει κάθε χιλιόμετρο της διαδρομής $V_i - V_1$.

Οι οδηγίες του Αι Βασίλη είναι σαφείς, οι βοηθοί δεν επιτρέπεται να παρεκκλίνουν από τη διαδρομή προς την πρωτεύουσα. Ένα έλκυθρο που ξεκινάει από την πόλη V_i μπορεί να κινηθεί μόνο στη διαδρομή $V_i - V_1$. Για να επιταχυνθεί η διαδικασία, όταν το έλκυθρο που ξεκίνησε από τη V_i φτάσει σε κάποια ενδιάμεση πόλη V_j , μπορεί είτε να συνεχίσει προς τη V_1 , είτε να παραδώσει τα γράμματα που μεταφέρει στον βοηθό της πόλης V_j , στην περίπτωση που αυτός μπορεί να κινηθεί ταχύτερα. Ο βοηθός V_j θα χρειαστεί, με τη σειρά του, P_j δευτερόλεπτα για να ξεκινήσει με το έλκυθρό του και S_j δευτερόλεπτα για κάθε χιλιόμετρο της διαδρομής $V_j - V_1$.

Όλοι οι βοηθοί είναι έτοιμοι να ξεκινήσουν! Χρειάζονται από σας ένα πλάνο που θα ακολουθεί τους κανόνες του Αι Βασίλη και θα εξασφαλίζει ότι τα γράμματα κάθε πόλης V_2, \dots, V_N θα φτάσουν στην πρωτεύουσα V_1 το συντομότερο δυνατόν (αφού φτάσουν στη V_1 , όλα τα γράμματα μεταφέρονται στιγμαία στο Ροβανιέμι με το έλκυθρο του ίδιου του Αι Βασίλη). Μένει λοιπόν να γράψετε ένα πρόγραμμα που θα υπολογίζει τον ελάχιστο χρόνο (σε δευτερόλεπτα) για να μεταφερθούν τα γράμματα κάθε πόλης V_i στην πρωτεύουσα V_1 .

Λεδομένα Εισόδου: Αρχικά, το πρόγραμμά σας θα διαβάσει από το standard input έναν θετικό ακέραιο N , το πλήθος των πόλεων στις οποίες έχουν συγκεντρωθεί τα γράμματα με τις επιθυμίες των παιδιών. Καθεμία από τις επόμενες $N - 1$ γραμμές θα περιέχει τρεις θετικούς ακέραιους V_i, V_j, D_{ij} , που δηλώνουν ότι υπάρχει διαδρομή μήκους D_{ij} μεταξύ των πόλεων V_i και V_j . Θα ακολουθούν $N - 1$ ακόμη γραμμές με δύο θετικούς ακέραιους η καθεμία. Το ζευγάρι αριθμών P_{i+1} και S_{i+1} στην i -οστή γραμμή δηλώνει ότι το έλκυθρο που ξεκινά από την πόλη V_{i+1} χρειάζεται P_{i+1} δευτερόλεπτα για να ξεκινήσει και S_{i+1} δευτερόλεπτα για κάθε χιλιόμετρο της διαδρομής $V_{i+1} - V_1$. Μπορείτε να θεωρήσετε ως δεδομένο ότι το δίκτυο διαδρομών θα είναι συνεκτικό και θα έχει δενδρική δομή.

Για το 80% της βαθμολογίας, θα είναι $N \leq 8000$. Για ένα επιπλέον 30% της βαθμολογίας, θα είναι $2 \cdot 10^4 \leq N \leq 10^5$ και οι πόλεις θα βρίσκονται σε μια νοητή ευθεία που εκτείνεται εκατέρωθεν της πρωτεύουσας V_1 .

Λεδομένα Εξόδου: Το πρόγραμμά σας πρέπει να τυπώνει στο standard output $N - 1$ θετικούς ακέραιους χωρισμένους με ένα κενό μεταξύ τους. Ο i -ος ακέραιος πρέπει να αντιστοιχεί στον ελάχιστο χρόνο (σε δευτερόλεπτα) για να μεταφερθούν τα γράμματα της πόλης V_{i+1} στην πρωτεύουσα V_1 . Σημειώστε ότι οι ελάχιστοι χρόνοι άφιξης των γραμμάτων στην V_1 (καθώς και κάποια από τα ενδιάμεσα αποτελέσματα που χρειάζονται για τον υπολογισμό τους) μπορεί να υπερβαίνουν το 2^{32} .

Περιορισμοί:

$3 \leq N \leq 8000$ (για 80%)

$3 \leq N \leq 10^5$ (για 110%)

$1 \leq D_{ij}, P_i, S_i \leq 10^7$

Όριο χρόνου εκτέλεσης: 1 sec.

Όριο μνήμης: 64 MB.

Παράδειγμα Εισόδου:

5

1 2 20

2 3 12

2 4 1

4 5 3

26 9

1 10

500 2

2 30

Παράδειγμα Εξόδου:

206 321 542 328