



Odading Izuri

Deskripsi

Kota Naga adalah sebuah kota yang menjadi pusat kuliner dari provinsi Xing Fu Ni. Salah seorang penduduk Kota Naga adalah Izuri, yang juga seorang pecinta wisata kuliner. Suatu hari Izuri sedang bermain ponsel dan melihat berita tentang kue odading yang sedang hits. Ia pun tertarik untuk membeli kue tersebut. Ia akhirnya berencana untuk berkunjung dari toko ke toko tentunya dengan tetap melakukan 3M.

Kota tersebut terdiri atas N toko yang diberi nama R_1, R_2, \dots, R_N . Pada kota tersebut terdapat M ruas jalan biasa dua arah yang menghubungkan toko A_i dan toko B_i dengan waktu tempuh C_i detik. Terdapat pula E buah jalan eksklusif dua arah yang menghubungkan toko F_i dan G_i dengan waktu tempuh hanya 1 detik. Selain itu, di tiap jalan terdapat kupon yang dapat digunakan untuk mendapat diskon namun dengan pembelian minimum tertentu, yaitu sebanyak K_i untuk jalan biasa dan L_i untuk jalan eksklusif. Uniknya, baik K_i maupun L_i pasti dapat dinyatakan dalam a^b dengan a dan b bilangan bulat dan nilai a sama untuk seluruh jalan pada kota. Izuri ingin menggunakan semua kupon diskon dari jalan yang ia lalui namun sayangnya pembelian minimum juga meningkat yaitu sebanyak **perkalian** dari pembelian minimum masing-masing kupon. Karena Kota Naga juga menerapkan PSBB, maka akan dilakukan penutupan tiap ruas jalan setelah melewati waktu tertentu yaitu pada waktu T_i untuk jalan biasa dan U_i untuk jalan eksklusif. Semua orang yang melalui suatu jalan harus **menyelesaikan** perjalanannya sebelum atau tepat waktu tutup jalan, berarti mulai paling lambat pada detik $T_i - C_i$ untuk jalan biasa dan $U_i - 1$ untuk jalan eksklusif.

Izuri pun meminta bantuan kepada Gibsy yang merupakan asistennya untuk membantunya. Kepada Gibsy, ia akan melakukan Q buah operasi yang akan membantunya menuju ke tempat odading yaitu:

- Menanyakan ada berapa ruas jalan baik eksklusif maupun biasa yang masih bisa dikunjungi pada waktu ke- X . Jika jalan tersebut mulai ditutup pada waktu ke- X berarti jalan tersebut sudah tidak bisa dikunjungi.
- Menanyakan apakah toko S_1 dan S_2 terhubung melalui ruas jalan apa pun atau tidak. Pada pertanyaan ini, Izuri **tidak** peduli akan waktu tempuh atau waktu tutup jalan.
- Menanyakan berapa pembelian minimum yang harus dilakukan jika Izuri menggunakan semua kupon yang didapat dari perjalanannya ke S_1 jika ia sekarang berada di S_2 , dan tentu saja Izuri ingin meminimalkan pembelian minimum yang harus dilakukan. Karena hasilnya bisa sangat besar, ia sudah cukup puas jika mendapat hasil modulonya dengan 10^9+7 . Perhatikan bahwa jalan eksekutif dan jalan reguler bisa digunakan serta kupon mereka juga bisa digabung. Pada pertanyaan ini, Izuri juga **tidak** peduli akan waktu tempuh atau waktu tutup jalan.

- Menanyakan pada detik berapa Izuri **paling lambat** harus berangkat agar bisa membeli Odading di toko S_1 jika ia sekarang berada di toko S_2 dengan hanya menggunakan jalan eksklusif. (Anggap waktu awal adalah detik 0)
- Menanyakan pada detik berapa Izuri **paling lambat** harus berangkat agar bisa membeli Odading di toko S_1 jika ia sekarang berada di toko S_2 dengan hanya menggunakan jalan biasa. (Anggap waktu awal adalah detik 0)

Gibsy pun kebingungan untuk melakukan satu per satu operasi yang diminta Izuri. Ia pun meminta bantuan anda sebagai seorang programmer handal untuk membuatkan suatu program yang bisa membantunya mengerjakan satu persatu operasi Izuri karena ia tidak ingin mengecewakan Izuri. Bantulah Gibsy!

Format Masukan

Baris pertama berisi empat buah bilangan bulat N, M, E, Q yang menyatakan **banyaknya toko** pada kota Naga, **banyaknya ruas jalan biasa** pada kota Naga, **banyaknya ruas jalan eksklusif** pada kota Naga, dan **banyaknya pertanyaan** yang akan diajukan Izuri ke Gibsy.

Baris berikutnya berisi N buah string yang menyatakan nama tiap toko.

M buah baris berikutnya berisi dua buah string A_i dan B_i dan tiga buah bilangan bulat C_i, K_i , dan T_i yang menyatakan **toko** A_i dan B_i terhubung oleh ruas jalan biasa dua arah dengan waktu tempuh sebesar C_i detik, kupon K_i , dan akan ditutup pada detik ke- T_i .

E buah baris berikutnya berisi dua buah string F_i dan G_i dan dua buah bilangan L_i dan U_i yang menyatakan **toko** F_i dan G_i terhubung oleh ruas jalan eksklusif dengan waktu tempuh 1 detik, kupon L_i , dan akan ditutup pada detik ke- U_i .

Q buah baris berikutnya berisi operasi yang akan diajukan Izuri ke Gibsy dengan format sebagai berikut:

- **TANYA_JALAN X** , yang berarti Izuri akan bertanya ada berapa ruas jalan baik eksklusif maupun biasa yang masih bisa dikunjungi pada waktu ke- X . (Jika jalan tersebut mulai ditutup pada waktu ke- X berarti jalan tersebut tidak bisa dikunjungi)
- **TANYA_HUBUNG $S_1 S_2$** , yang berarti Izuri akan bertanya apakah **toko** S_1 dan S_2 terhubung baik secara langsung ataupun tidak langsung melalui ruas jalan biasa ataupun jalan eksklusif tanpa memedulikan waktu tempuh atau waktu tutup jalan.
- **TANYA_KUPON $S_1 S_2$** yang berarti Izuri akan bertanya berapa minimal pembelian minimum yang harus dilakukan jika menggunakan kupon-kupon diskon pada perjalanan membeli Odading di toko S_1 jika ia sekarang berada di toko S_2 melalui ruas jalan biasa ataupun jalan eksklusif tanpa memedulikan waktu tempuh atau waktu tutup jalan.
- **TANYA_EX $S_1 S_2$** , yang berarti Izuri akan bertanya pada detik berapa ia paling lambat harus berangkat agar bisa membeli Odading di toko S_1 jika ia sekarang berada di toko S_2 dengan hanya menggunakan jalan eksklusif. (Anggap waktu awal adalah 0 detik)
- **TANYA_BIASA $S_1 S_2$** , yang berarti Izuri akan bertanya pada detik berapa ia paling lambat harus berangkat agar bisa membeli Odading di toko S_1 jika ia sekarang berada di toko S_2 dengan hanya menggunakan jalan biasa. (Anggap waktu awal adalah 0 detik)

Format Keluaran

Untuk masing-masing operasi di bawah ini keluarkan output sesuai yang diinginkan:

- **TANYA_JALAN X**, keluarkan sebuah bilangan bulat yang menyatakan banyak ruas jalan baik eksklusif maupun biasa yang masih bisa dikunjungi pada waktu ke-**X**.
- **TANYA_HUBUNG S1 S2**, keluarkan sebuah string yaitu **YA** jika **toko S1** dan **S2** terhubung dan **TIDAK** jika **toko S1** dan **S2** tidak terhubung.
- **TANYA_KUPON S1 S2**, keluarkan sebuah bilangan bulat yang menyatakan berapa pembelian minimum yang harus Izuri lakukan. Karena hasilnya bisa sangat besar, cukup keluarkan dalam modulo 10^9+7 . **Keluarkan -1 jika Izuri tidak akan pernah bisa membeli Odading.**
- **TANYA_EX S1 S2**, keluarkan sebuah bilangan bulat yang menyatakan pada detik berapa Izuri paling lambat harus berangkat agar bisa membeli Odading di toko **S1** jika ia sekarang berada di toko **S2** dengan hanya menggunakan jalan eksklusif. Keluarkan **-1** jika Izuri tidak akan pernah bisa membeli Odading.
- **TANYA_BIASA S1 S2**, keluarkan sebuah bilangan bulat yang menyatakan pada detik berapa Izuri paling lambat harus berangkat agar bisa membeli Odading di toko **S1** jika ia sekarang berada di toko **S2** dengan hanya menggunakan jalan biasa. Keluarkan **-1** jika Izuri tidak akan pernah bisa membeli Odading.

Batasan

- $2 \leq N \leq 1.000$
- $0 \leq M, E \leq 1.000$
- $1 \leq Q \leq 1.000$
- $1 \leq |R_i| \leq 10$ dan R_i hanya terdiri dari huruf alfabet kapital.
- Nama masing-masing **toko** unik
- $0 \leq C_i, T_i, U_i, X \leq 100.000$
- $1 \leq K_i, L_i \leq 100.000$
- K_i dan L_i dapat dinyatakan dalam suatu bentuk a^b dengan a dan b bilangan bulat dan nilai a sama untuk seluruh jalan pada kota.
- **S1, S2** $\in \mathbb{R}$ dan **S1** \neq **S2** untuk semua jenis operasi

Contoh Masukan 1

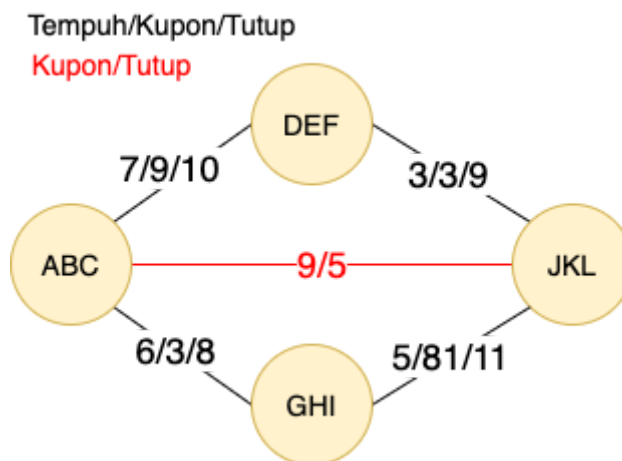
```
4 4 1 6
ABC DEF GHI JKL
ABC DEF 7 9 10
ABC GHI 6 3 8
JKL GHI 5 81 11
DEF JKL 3 3 9
ABC JKL 9 5
TANYA_JALAN 5
TANYA_HUBUNG DEF GHI
TANYA_EX JKL ABC
TANYA_BIASA JKL ABC
TANYA_BIASA ABC JKL
TANYA_KUPON JKL ABC
```

Contoh Keluaran 1

4
YA
4
0
0
9

Penjelasan 1

Berikut ini ilustrasi toko-toko di Kota Naga:



- Pada detik kelima, empat jalan di buka : ABC-DEF , GHI-JKL, DEF-JKL, ABC-GHI.
- Tanpa memedulikan waktu tempuh dan tutup jalan, jelas DEF dan GHI terhubung.
- Izuri paling lambat harus berangkat pada detik ke-4 karena jalan ABC-JKL akan ditutup pada detik ke-5 dan perjalanan membutuhkan waktu satu detik.
- Rute ABC-DEF-JKL tidak dapat diambil karena Izuri akan tiba di JKL pada detik $7 + 3 = 10$ sedangkan DEF-JKL ditutup pada detik kesembilan. Satu-satunya rute adalah ABC-GHI-JKL yang mana Izuri paling lambat harus berangkat pada detik 0 karena jalan GHI-JKL akan tutup pada detik kesebelas.
- Rute JKL-GHI-ABC tidak dapat diambil karena jalan GHI-ABC akan tutup pada detik kedelapan. Satu-satunya rute adalah JKL-DEF-ABC yang mana Izuri harus berangkat pada detik 0 karena jalan DEF-ABC tutup pada detik 10.
- Ada tiga rute yang dapat ditempuh yaitu ABC-DEF-JKL yang menyebabkan $9 \times 3 = 27$ minimum pembelian kue, ABC-JKL yang menyebabkan 9 minimum pembelian kue, atau ABC-GHI-JKL yang menyebabkan $3 \times 81 = 243$ minimum pembelian kue. Dari ketiga itu, yang paling minimal adalah ABC-JKL dengan 9 minimum pembelian.

Contoh Masukan 2

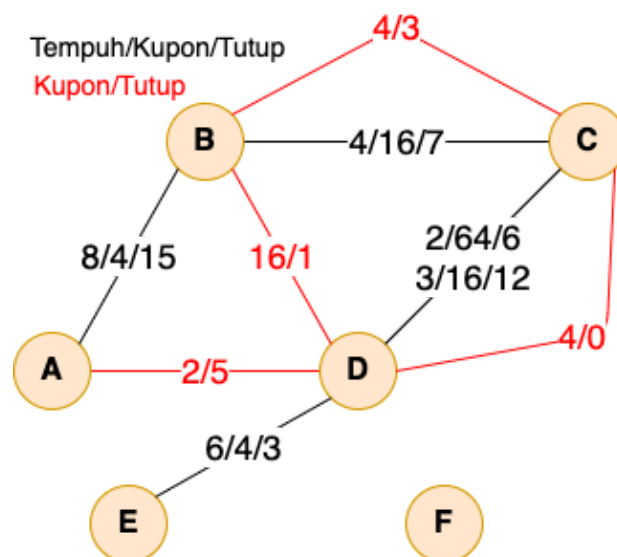
6 5 4 5
A B C D E F
D C 2 64 6
C D 3 16 12
A B 8 4 15

C B 4 16 7
 E D 6 4 3
 D B 16 1
 B C 4 3
 A D 2 5
 C D 4 0
 TANYA_JALAN 4
 TANYA_HUBUNG E A
 TANYA_HUBUNG F C
 TANYA_EX D C
 TANYA_BIASA D B

Contoh Keluaran 2

5
 YA
 TIDAK
 -1
 3

Penjelasan 2



- Saat waktu 4, terdapat 4 jalan yang masih terbuka, rute reguler B-A, rute reguler D-C, rute reguler B-C, dan rute eksekutif A-D
- Toko A terhubung ke toko E. Contoh jalur eksekutif A -> D, lalu jalur reguler D -> E.
- Akan tetapi, toko F tidak terhubung sama sekali, sehingga mustahil F -> C
- Tidak mungkin dari C ke D tepat waktu. Jalur eksekutif C -> D sudah tertutup dari awal. Mungkin kita punya ide untuk C -> B -> D, tetapi saat kita sampai ke B, jalur B -> D sudah tertutup.
- Salah satu solusi potensial adalah B -> C -> D melalui jalur 2/64/6. Akan tetapi, jika kita melewati jalur 2/64/6, kita harus berangkat dari 0 agar sampai tepat waktu. Solusi yang lain adalah B -> C -> D melalui jalur 3/16/12. Jika kita menggunakan strategi ini, kita bisa baru

berangkat saat waktu = 3, lalu sampai di C pada waktu = 7, lalu menuju D dan sampai pada waktu = 10

Pembagian Test Case

Variasi Case	Test Case Number
TANYA_JALAN	1-8
TANYA_HUBUNG	9-16
TANYA_EX	17-24
TANYA_BIASA	25-32
TANYA_KUPON	33-40
Mixed Queries	41-60