

Şirket

Zaman kısıtı: 2 sn

Memory kısıtı: 128 MB

Bir yazılım şirketi kendi işleyişini düzenlemek için bir hiyerarşik düzen oluşturmuştur. Her çalışanın bir yöneticisi vardır ancak CTO (Chief Technology Officer veya Teknoloji Direktörü) hariç. Şirket her sabah projeleri takip etmek için günlük toplantılar yapmaktadır. Toplantıların kısa sürmesi için her toplantının büyüklüğünün tam olarak K kişi olmasına karar verilmiştir. Her toplantı bir yönetici ve o yöneticinin hemen altında çalışan tam olarak $K - 1$ çalışan ile gerçekleşecektir. Tabii ki her insan sadece bir toplantıya dahil olabileceği için tüm toplantı grupları ayrıktır. Şirketin yöneticisi haberleşmenin düzgün olması için her gün farklı bir toplantı grup konfigürasyonu oluşturulmasını istemiştir ve bir önceki toplantı konfigürasyonunu tekrar etmeden en fazla kaç günlük toplantı planlanabilir merak etmektedir. Sizden istenen bu sayıyı hesaplamanız ve bildirmenizdir.

Girdi Formatı

Girdinin ilk satırında çalışan sayısını belirten N tam sayısı ve toplantı büyüklüğünü veren K tam sayısı yer alacaktır.

İkinci satırda sırasıyla 2 numaralı çalışandan başlayıp N numaralı çalışana kadar çalışanın yöneticisinin numarasını belirten $N - 1$ tam sayı yer alacaktır.

CTO 1 numaralı çalışandır ve her çalışan bir hiyerarşi zinciriyle CTO ya bağlıdır (diğer bir deyişle şirket çalışanları bir ağaç yapısı oluşturmaktadır.)

Çıktı Formatı

Çıktıda tek bir satırda kaç gün boyunca farklı geçerli toplantı konfigürasyonu oluşturulabileceğini belirten tek bir tam sayı yer alacaktır. Bu sayı çok büyük olabileceği için $10^9 + 7$ modundaki değerini yazdırmanız beklenmektedir.

Kısıtlar

- $2 \leq K \leq N \leq 200\,000$
- $2 \leq K \leq 500$

Alt Görevler

Alt Görev 1 (13 puan)

- $2 \leq K \leq N \leq 10$

Alt Görev 2 (14 puan)

- $1 \leq N \leq 100$
- $2 \leq K \leq 4$

Alt Görev 3 (15 puan)

- $1 \leq N \leq 1000$
- $K = 3$

Alt Görev 4 (17 puan)

- $1 \leq N \leq 200\,000$
- $K = 3$
- Her çalışanın en fazla 2 adet alt çalışanı bulunmaktadır.

Alt Görev 5 (41 puan)

- Ekstra kısıt bulunmamaktadır.

Örnekler

| Örnek Girdi | Örnek Çıktı | Açıklama |
|------------------------|-------------|--|
| 5 3 1 1 3 3 | 3 | Gün 1: {1,2,3} Gün 2: {3,4,5} Gün 3: {} |
| 6 3 1 1 1 2 2 | 6 | Gün 1: {1,2,3} Gün 2: {1,3,4} Gün 3: {1,2,4} Gün 4: {1,3,4}, {2,5,6} Gün 5: {2,5,6} Gün 6: {} |
| 9 9 1 2 3 4 5 6 7 8 | 1 | Gün 1: {} |

Edi ile Bd

Zaman kısıtı: 3 sn

Memory kısıtı: 256 MB

Edi ile Bd beraber oyunlar oynayıp Őarkılar syleyen iki yakın arkadaŐmıŐ. Bir gn Edi N adet pul ve bu pulları baēlayan $N - 1$ adet de ip bulmuŐ. İplerin her birinin uzunluēu bir birim olmakla birlikte zerlerinde sayılar da yazmaktaymıŐ. Bu iplerin her birinin iki farklı pulu birbirine baēlamakta olduēunu gren Edi, aynı zamanda iki farklı pul seēildiēinde ipleri takip ederek birinden diēerine tam olarak bir yolla ulaŐılabildiēini fark etmiŐ. Diēer bir deyiŐle, pullar ve ipler bir aēaē temsil etmektelemiŐ.

Bu yeni keŐfinden ok heyecanlanan Edi, hemen Bd'nn yanına gitmiŐ ve bu pullarla bir oyun oynamayı teklif etmiŐ. Oyunun kuralları ise Őu ŐekildeymiŐ:

- Bd nceden belirlenmiŐ A numaralı puldan baŐlayan yine nceden belirlenmiŐ k birim uzunluēunda bir yol seēer. Diēer bir deyiŐle, $k + 1$ tane pul seēer yle ki bu pulların numaraları $(p_1 = A, p_2, p_3, \dots, p_{k+1})$ olarak gsterilirse her $1 \leq i \leq k$ iin p_i ile p_{i+1} pulları arasında ip bulunsun.
- Ardından Edi, bu pullardan istediēi iki tanesini seēip Bd'ye syler. Diēer bir deyiŐle $1 \leq x, y \leq k + 1$ olacak Őekilde p_x ve p_y numaralı pulları seēip bunu Bd'ye syler.
- Son olarak Bd, baŐta seētiēi $k + 1$ pulun her birine herhangi bir pul ikilisinin arasındaki ipin stnde bulunan sayı bu iki pulun stnde yazan sayının farkına eŐit olacak Őekilde birer sayı yazar. Bu hamleyle birlikte oyun biter ve iki arkadaŐın kazandıēı puan Edi'nin seētiēi iki pulun stnde yazan sayıların farkına eŐit olur. Diēer bir deyiŐle, Bd $k + 1$ uzunluēunda bir sayı dizisi oluŐturup p_i numaralı pulun stnde bu sayılardan i . si yazılacak Őekilde pullara sayılar yazar. Bu numaralar her $1 \leq i \leq k$ iin p_i ile p_{i+1} numaralı pulların stnde yazan sayının farkı bu iki pulu baēlayan ipin stnde yazan sayıya eŐit olacak Őekilde verilmek zorundadır. Ardından bu oyundan Edi ile Bd, p_x ile p_y numaralı pullarda yazan sayıların farkı kadar puan kazanırlar.

Edi'nin hedefi kazandıkları puanı ykseltmekken erkenden oyunu bitirip kitap okumak isteyen Bd'nn hedefi ise bu puanı olabildiēince azaltmıŐ.

Bu bilgiler ışığında, Edi ile BÜdü sizden aşağıdaki iki tip sorguyu gerçekleştiren bir program yazmanızı istemektedirler:

- Verilen bir ipin üstünde yazan sayıyı güncelle.
- Önceden belirlenmiş olan A, k sayılarının verilmesi durumunda oyun sonunda kazanılacak puanı hesapla. İki oyuncunun da optimal oynadığını düşünebilirsiniz. Eğer BÜdü belirlenen sayılar için A numaralı puldan başlayan k birim uzunluğunda bir yol seçemiyorsa -1 yazarak bu oyunun oynanamayacağını belirtmeniz gerekmektedir.

Girdi:

İlk satırda test senaryosu sayısını belirten bir t tamsayısı olacaktır. Ardından her bir test senaryosu için açıklama aşağıdaki gibidir:

Her bir test senaryosu ilk satırında sırayla pul sayısı ve sorgu sayısını belirten N ve Q tamsayıları.

Takip eden $N - 1$ satırda boşlukla ayrılmış şekilde pullar arasındaki ipler. i . satırda (U_i, V_i, W_i) üçlüsü verilir bu üçlü U_i ve V_i numaralı pullar arasında üzerinde W_i yazan bir ip olduğunu belirtir. Verilen yapının bir ağaç oluşturduğu garanti edilmektedir.

Takip eden Q satırda ilk sayı hangi sorgunun yapıldığını belirten bir tamsayı (1 ya da 2) olmak üzere aşağıdakilerden biri verilecektir:

- **1 s e** - Girdide s numaralı sırada verilen ipin üstünde yazan sayıyı e olarak güncelle, yani $W_s = e$ olarak güncelle.
- **2 A k** - Verilen A, k sayıları için oyun sonunda kazanılacak puanı hesapla. İki oyuncunun da optimal oynadığını düşünebilirsiniz. Eğer BÜdü verilen sayılar için A numaralı puldan başlayan k birim uzunluğunda pullar seçemiyorsa -1 yazarak bu oyunun oynanamayacağını belirtmeniz gerekmektedir.

Çıktı:

Her bir ikinci tip sorgu için birer satırda istenen cevabı yazdırın.

Kısıtlar

- $1 \leq t \leq 10^4$
- $1 \leq N, Q \leq 5 * 10^4$
- $1 \leq W_i \leq 10^9$
- $1 \leq U_i, V_i \leq N, U_i \neq V_i$
- $1 \leq s < N, 1 \leq e \leq 10^9$
- $1 \leq A, k \leq N$

- Tüm test senaryoları için verilen N sayılarının toplamının $5 * 10^4$, Q sayılarının toplamının da $5 * 10^4$ sayısını geçmeyeceği garanti edilmektedir.

Alt Görevler

Alt Görev 1 [4 puan]

$$1 \leq N, W_i, e \leq 5, 1 \leq Q \leq 50$$

Tüm test senaryoları için verilen N sayılarının toplamının $5 * 10^3$, Q sayılarının toplamının da $5 * 10^3$ sayısını geçmeyeceği garanti edilmektedir.

Alt Görev 2 [8 puan]

$$1 \leq N, Q \leq 5 * 10^3$$

Tüm test senaryoları için verilen N sayılarının toplamının $5 * 10^3$, Q sayılarının toplamının da $5 * 10^3$ sayısını geçmeyeceği garanti edilmektedir.

Alt Görev 3 [5 puan]

Girdide verilen i numaralı ip 1 ve $i + 1$ numaralı pulları bağlamaktadır.

Alt Görev 4 [11 puan]

Girdide verilen i numaralı ip i ve $i + 1$ numaralı pulları bağlamaktadır.

Alt Görev 5 [20 puan]

$$1 \leq W_i, e \leq 5$$

Alt Görev 6 [17 puan]

Girdide ilk tip sorgu bulunmamaktadır.

Alt Görev 7 [35 puan]

Herhangi bir kısıtlama bulunmamaktadır.

Örnek Girdi

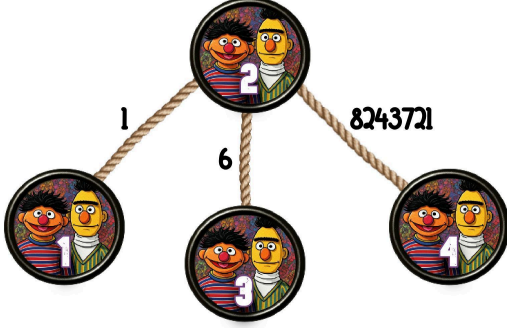
2
4 3
1 2 1
2 3 6
4 2 8247312
2 1 2
1 3 1
2 1 2
8 7
3 2 4
2 5 4
2 1 8
3 4 2
5 8 3
6 4 10
5 7 1
2 4 6
1 5 6
2 6 4
2 7 6
1 1 5
1 6 6
2 2 2

Örnek Çıktı

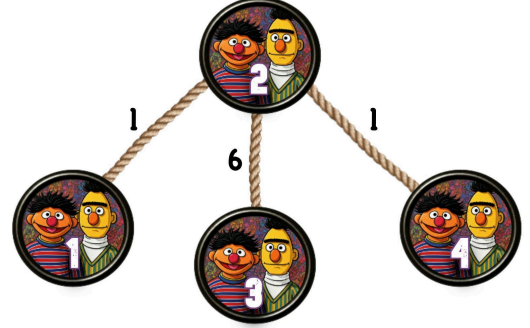
6
1
-1
10
-1
4

Örnek Açıklama

Birinci test senaryosu için oyun oynanan pullar aşağıdaki gibi görünmektedir:



Resim 1: İkinci sorgudan önce pullar



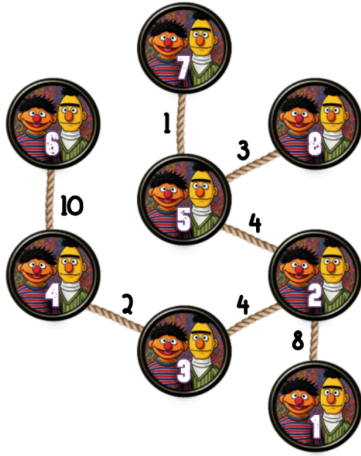
Resim 2: İkinci sorgudan sonra pullar

İlk sorguda BÜDÜ (1, 2, 3) pullarını seçer. Ardından EDİ bu pullardan (2, 3) ikilisini seçer. BÜDÜ pulları (3, 4, -2) olarak numaralandırırsa puan $|4 - (-2)| = 6$ olur. Farklı hamlelerle de bu puana ulaşabileceğini, ama optimal oynanması durumunda puanın 6'dan farklı olamayacağı gösterilebilir.

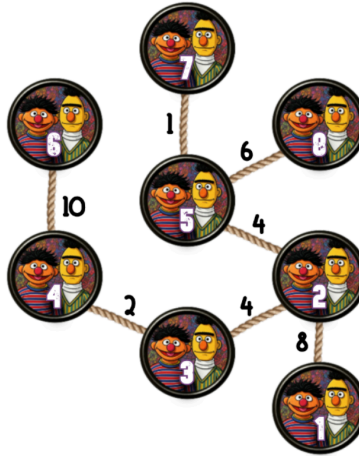
İkinci sorguda üç numaralı ipin üzerinde sayan yazı 1 olarak güncellenir. Sorgunun etkisini resimlere bakarak görebilirsiniz.

Üçüncü sorguda BÜDÜ (1, 2, 4) pullarını seçer. EDİ (1, 2) pullarını seçerse BÜDÜ'nün yapacağı herhangi bir eşlendirmede puanın 1 olacağı gösterilebilir. Eğer EDİ (1,4) pullarını seçerse BÜDÜ'nün (5, 4, 5) şeklinde bir sayı ataması yaparak puanı 0 yapabileceğini de not ediniz.

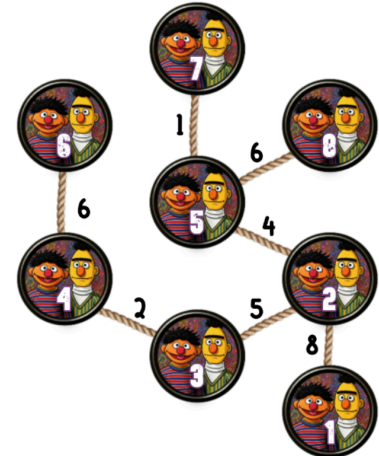
İkinci test senaryosu için oyun oynanan pullar aşağıdaki gibi görünmektedir:



Resim 3: İlk sorgu esnasında pullar



Resim 4: Üç ve dördüncü sorgular esnasında pullar



Resim 5: Yedinci sorgu esnasında pullar

İlk sorguda Būdū'nün seçebileceği 4 numaralı puldan başlayan 6 birim uzunluğunda bir yol bulunmadığı için cevap -1' dir.

İkinci sorguda 5. sırada verilen 5 ve 8 numaralı pulları bağlayan ipin üstünde yazan sayı 6 olarak güncellenir.

Üçüncü sorguda Būdū'nün seçebileceği olası hamlelerden;

- (6, 4, 3, 2, 1) pullarını seçmesi durumunda Edi (6, 4) pullarını seçer ve Būdū (3, 13, 11, 7, -1) sayılarını pulların üstüne yazarsa puan $|3 - 13| = 10$ olur.
- (6, 4, 3, 2, 5) pullarını seçmesi durumunda Edi (5, 4) pullarını seçer ve Būdū (11, 1, 3, 7, 11) sayılarını pulların üstüne yazarsa puan $|11 - 1| = 10$ olur.

Dördüncü sorguda 7 numaralı pul başlayan 6 birim uzunluğunda bir yol bulunmadığı için cevap -1'dir.

Beşinci ve altıncı sorgularda sırasıyla 1. ve 6. sırada verilen iplerin değerleri 5 ve 6 olarak güncellenir.

Yedinci sorguda Būdū (2, 5, 7), (2, 5, 8) ya da (2, 3, 4) pullarını seçebilir. Bunlardan ilkinin seçmesi ve iki oyuncunun da optimal oynaması durumunda puan 4 olacaktır.

Akrostiş

Zaman kısıtı: 2 sn

Memory kısıtı: 256 MB

Adını ünlü ozan Simit'ten alan Simitistan diyarı, insanları arasındaki sıradışı iletişim yöntemleri ile nam salmıştır. Bunlara birbirlerinin kişisel mülkiyetlerine zarar vermek ya da akrostiş şiir yazmak örnek verilebilir. Simitistan'a özgür ruhu ve sakarlığı ile nam salmış Karagül de, arkadaşı Çelik'e sevgisini göstermek için bir akrostiş şiir yazmaya karar verir. Maalesef ki Karagül bu şiiri yazarken gözlüklerini takmayı unutur ve yazdığı şiirin dizelerinin baş harfleri "Çelik" yerine "Çilek" kelimesini oluşturur.

Bunu bir oyun daveti olarak algılayan Çelik, Karagül'ün tarlasını spesifik bir yapıda bozarak iade-i ziyarette bulunmak ister. Karagül'ün tarlası M parçadan oluşmaktadır, soldan sağa numaralandırılmıştır ve uzun çalışmalar sonucu düzleştirilmiştir, yani her parçanın yüksekliği en başta 0'dır. Çelik bu tarlaya N farklı hamleyle saldırabilir. Her bir yapabileceği saldırı L_i ve R_i parametrelerinden oluşan kapalı aralıklar ile temsil edilmektedir ve her saldırı birden fazla kez kullanılabilir. Çelik bir saldırıyı yaparken rasyonel bir x parametresi ve seçer. Bu saldırı ile Çelik, (eğer x pozitif ise bölgeye çimento dökerek yüksekliği artırır, negatif ise bölgedeki taş ve toprağı çuvalına doldurup sırtlanarak yüksekliği azaltır) numarası L_i ve R_i (sınırlar dahil) aralığı içerisinde bulunan **tüm** tarla parçalarının yüksekliğine x ekler.

Size Çelik'in bu saldırıları yaparak tarlayı istediği bir yapıya getirip getiremeyeceği sorulmaktadır. Toplamda Q sorgu olacaktır. i . sorguda size bu sorgudaki koşul sayısı K_i verilecektir. Bu i . sorgudaki j . koşul, tarlanın $S_{i,j}$ ile numaralandırılmış parçasının $T_{i,j}$ yüksekliğinde olmasıdır. Kalan parçaların yükseklikleri için herhangi bir koşul yoktur ve sorgular birbirinden bağımsızdır. Daha resmi olarak, her bir i sorgusu için bir aşağıdaki verilen koşulu sağlayan bir C_x rasyonel sayı dizisinin olup olmadığını bulmanız gerekmektedir.

$$\forall_j \sum_{x: [L_x, R_x] \ni S_{i,j}} C_x = T_{i,j}$$

Not: Rasyonel sayılar, p ve $q \neq 0$ birer tamsayı olmak üzere $\frac{p}{q}$ şeklinde yazılabilen tüm sayılardır.

İmplementasyon Detayı

Size verilen bir fonksiyonu implement etmeniz gerekmektedir.

akrostis[] (M, N, Q, L[], R[], K[], S[], T[])

- Bu fonksiyonun Q uzunluğunda *boolean* bir dizi döndürmesi gerekmektedir (Uzunluğun yanlış olması durumunda “**Protocol Violation**” hatası alacaksınız)
- M tarlanın uzunluğunu temsil etmektedir
- N Çelik’in yapabileceği farklı saldırıların sayısıdır
- Q toplam sorgu sayısıdır
- $L[]$ ve $R[]$ saldırıları temsil etmektedir
- $K[], S[]$ ve $T[]$ sorguları temsil etmektedir
- Değişken detayları için size verilen taslak koda danışabilirsiniz

Bu fonksiyon her bir testte I defa çağırılacaktır. Global değişkenlerinizin önceki fonksiyon çağrıları sebebiyle değişebileceğini unutmayın.

Girdi ve çıktılar size verilen grader.cpp tarafından yapılacaktır, çözüm kodunuzda standart girdi ya da çıktıyla herhangi bir şekilde etkileşim olursa (cin, cout, printf, scanf) çalışma zamanı ya da “Protocol Violation” hatası alacaktır ve yanlış sayılacaktır. Standart hata çıktısını (cerr, fprintf(stderr, ...)) istediğiniz şekilde kullanabilirsiniz.

Girdi

İlk satırda I (test sayısı) verilecektir. Her bir test aşağıdaki formatta verilecektir.

```
M N Q
L0 R0
:
LN-1 RN-1
K0
S0,0 T0,0
:
S0,K0-1 T0,K0-1
:
KQ-1
SQ-1,0 TQ-1,0
:
SQ-1,KQ-1 TQ-1,KQ-1
```

Çıktı

I testin her biri için aşağıda formatta cevaplar verilecektir.

#"testin indeksi"

A_0

\vdots

A_{Q-1}

A_i i . sorgunun kod tarafından verilen cevabını temsil etmektedir.

boolean değerler 0 ve 1'ler olarak yazdırılacaktır.

Kısıtlar

- $F(X) = X$ değişkeninin I testteki değerlerinin toplam'ını ifade etmektedir, kalan kısıtların tamamı testler arasında bağımsızdır
- $1 \leq I \leq 10^5$
- $F(N), F(Q), F(\sum K_i) \leq 10^5$
- $N, Q, K_i > 0$
- $0 \leq S_{i,j} < M$
- $|T_{i,j}| \leq 10^9$
- $0 \leq L_i \leq R_i < M \leq 10^9$
- Her $0 \leq i < Q$ ve $0 \leq j < K_i - 1$ ikilisi için $S_{i,j} < S_{i,j+1}$ garantelenmiştir
- Bir testte, aynı aralıklar (L_i, R_i ikilileri) birden fazla kez verilebilir.
- Girdide verilen her sayı bir tamsayıdır.

Alt Görevler

Alt Görev 1 [3 puan]

$$F(N), F(\sum K_i); M \leq 100$$

Alt Görev 2 [4 puan]

$$F(N), F(\sum K_i) \leq 3000; Q = 1$$

Alt Görev 3 [6 Puan]

$$F(N), F(\sum K_i) \leq 3000$$

Alt Görev 4 [8 Puan]

$$F(N), F(\sum K_i) \leq 3000; \text{Herhangi iki aralık kesişiyorsa biri diğerini içerir (kapsar).}$$

Alt Görev 5 [14 Puan]

Herhangi iki aralık kesişiyorsa biri diğerini içerir (kapsar).

Alt Görev 6 [21 Puan]

Herhangi iki aralık kesişiyorsa ve başlangıç ve bitişleri farklı noktaları farklıysa, birbirlerini asla içermezler. Daha resmi olarak $L_i < L_j$ koşulun sağlayan her $0 \leq i, j < N$ ikilisi için $R_i \leq R_j$ garantilenmiştir.

Alt Görev 7 [20 Puan]

$$1 \leq F(N), F(\sum K_i) \leq 5 \times 10^4$$

Alt Görev 8 [24 Puan]

Ek kısıt yoktur.

Örnek Girdi

```
1
11 4 4
0 4
2 6
2 8
6 10
6
0 2
1 2
2 3
6 7
7 9
11 0
3
1 6
2 6
3 7
3
3 -5
5 -10
7 -30
1
11 1
```

Örnek Çıktı

```
#0
1
0
1
0
```

Örnek Açıklama

Çelik, 1. sorguda, hepsinde x 'in mutlak değerini 1 seçip $[0, 4]$ saldırısını 2 kere çimento dökerek, $[2, 6]$ saldırısını 2 kere sırtlanarak, $[2, 8]$ saldırısını 3 kere çimento dökerek ve $[6, 10]$ saldırısını 6 kere çimento dökerek kullanıp Karagül'ün tarlasını istediği şekle getirebilir. 2. ve 4. sorgular için tarlanın istenilen şekle getirilemeyeceği kanıtlanabilir.

Size ayrıca ek dosyalar içerisinde 3 örnek girdi daha verilmiştir. 1. ve 3. girdiler 6. alt görevin koşulunu sağlamaktadır. 2. ve 4. girdiler 5. alt görevin koşulunu sağlamaktadır. Ayrıca 2. girdi 2. alt görevin koşullarını sağlamaktadır.

Ağaçtaki Eşsizlik

Zaman kısıtı: 4 sn

Memory kısıtı: 256 MB

Her ağaç eşsizdir, fakat ağaçlardaki değerler daha da eşsizdir.

Yine bir gün arkadaşınızla kim algoritmalar konusunda daha iyi diye tartışırken arkadaşınıza zor olduğunu düşünerek bir problem sordunuz. Sorduğunuz problemde her birinde bir değer olan N düğümden oluşan kökü düğüm 1 olan bir ağaç veriliyor ve her düğümün alt ağacındaki eşsiz değer sayısı isteniyor.

Ne yazık ki siz tam arkadaşınızı alt ettiğinizi düşünürken o çözümü hemen buluverdi. Üstüne üstlük bu problemin çok kolay olduğunu söyledi ve size daha zor bir versiyonunu sormak için probleme Q adet sorgu ekledi. Her sorgu bir düğümdeki değeri verilen yeni değerle değiştiriyor ve o düğümün alt ağacındaki eşsiz değer sayısını istiyor.

Arkadaşınız problemin bu versiyonunu çözemeyeceğinizden emin konuşuyor fakat tabii ki siz bunun altında kalmak istemiyorsunuz. Onun problemini çözüp, üstüne üstlük kendinizi kanıtlamak için bir de ayrıca eşsiz değer sayısı yerine eşsiz değerlerin toplamını da bulmak istiyorsunuz.

Girdi

İlk satırda iki adet tam sayı N ve F . Düğüm sayısı ve problem versiyonunu belirten değişken. (0 ise eşsiz değerlerin sayısı, 1 ise eşsiz değerlerin toplamı)

Takip eden $N - 1$ satırda iki adet tam sayı A ve B . Ağaçtaki kenarlar.

Sonraki satırda N adet tam sayı içeren dizi C . Ağaçta başlangıçta bulunan değerler.

Sonraki satırda bir adet tam sayı Q . Sorgu sayısı.

Takip eden Q satırda iki adet tam sayı V ve X . Sorguda verilen düğüm ve yeni değer.

Çıktı

$1 \leq i \leq Q$ için i . satırda i . sorgunun cevabı.

Kısıtlar

- $1 \leq N, Q \leq 2 \times 10^5$
- $0 \leq F \leq 1$
- $1 \leq A_i, B_i, C_i, V_i, X_i \leq N$

Alt Görevler

Alt Görev 1 [7 puan]

- $N, Q \leq 10^3$

Alt Görev 2 [19 puan]

- $N, Q \leq 5 \times 10^4$
- $F = 0$

Alt Görev 3 [21 puan]

- $N, Q \leq 5 \times 10^4$
- $F = 1$

Alt Görev 4 [24 puan]

- $N, Q \leq 2 \times 10^5$
- $F = 0$

Alt Görev 5 [29 puan]

- $N, Q \leq 2 \times 10^5$
- $F = 1$

Örnek Girdi 1

```
7 0
1 2
1 3
2 4
2 5
3 6
3 7
1 1 2 2 2 1 1
5
1 3
2 2
3 3
7 4
1 5
```

Örnek Çıktı 1

```
3
1
2
1
5
```

Örnek Açıklama 1

- İlk sorgudan sonra 1 düğümünün alt ağacındaki eşsiz değerler: 1, 2, 3
- İkinci sorgudan sonra 2 düğümünün alt ağacındaki eşsiz değerler: 2
- Üçüncü sorgudan sonra 3 düğümünün alt ağacındaki eşsiz değerler: 1, 3
- Dördüncü sorgudan sonra 7 düğümünün alt ağacındaki eşsiz değerler: 4
- Beşinci sorgudan sonra 1 düğümünün alt ağacındaki eşsiz değerler: 1, 2, 3, 4, 5

Örnek Girdi 2

```
7 1
1 2
1 3
2 4
2 5
3 6
3 7
1 1 2 2 2 1 1
5
1 3
2 2
3 3
7 4
1 5
```

Örnek Çıktı 2

```
6
2
4
4
15
```

Yorucu Çoraplar

Zaman kısıtı: 2 sn

Memory kısıtı: 256 MB

Büyük bir çorap temizleme fabrikasında çalışıyorsunuz. Kocaman bir makinede bütün çoraplar yıkanıyor, başka kocaman bir makinede de kurutuluyor. Sizin göreviniz ise kurutulan çorapları eşlemek.

Bugün N çorap çifti yıkandı. Çoraplar size birer birer verilecek. Kurutma makinesi matematiksel bir başyapıt olduğu için her bir çorap permütasyonunun verilme olasılığı eşittir. Önünüzde toplam K adet çorap tutabilen bir masa var. Elinizdeki çorabı soldan sağa sırayla önünüzdeki çoraplarla karşılaştırmanız gerekiyor. Her bir çorap karşılaştırması 1 kalori harcamanıza sebep oluyor. Eğer incelediğiniz çorap elinizdekiyle eşleşirse o çifti ayırıyorsunuz ve sağda geriye kalan çoraplar mekanik masa yardımıyla otomatik olarak bir sola kayarak boşluğu dolduruyor. O andan itibaren incelemeyi bırakıp sonraki çorabı alıyorsunuz. Eğer elinizdeki çorap masadaki hiçbir çorapla eşleşmiyorsa da o çorabı sağdaki ilk boşluğa koyuyorsunuz. Eğer masada boş yer yoksa da isyan edip istifa ediyorsunuz.

Çorapları eşleme işlemi oldukça yorucu olduğu için işe başlamadan önce yeterince yemek yemek istiyorsunuz fakat maaşınız az olduğu için gereğinden fazla yemek de istemiyorsunuz. Bunun için N çorap çifti ve K çorap tutabilen masa için gereken beklenen kalori değerini hesaplamak istiyorsunuz. Ayrıca her gün yıkanan çorap sayısı ve masanız değişebileceği için bunu T farklı gün için yapmak istiyorsunuz. T gün içinde toplam yıkanacak çorap çifti sayısının S olduğunu da unutmayorsunuz.

Bilgisayarlar ondalık sayıları tam olarak temsil edemediği için cevaplarınızı $\text{mod}(10^9 + 7)$ altında vermeniz gerekiyor. $\frac{1}{p} \text{mod}(10^9 + 7) = P^{10^9+5} \text{mod}(10^9 + 7)$ özelliği de asla unutmadığınız bir özellik olduğu için sorun yaşayacağınızı düşünmüyorsunuz.

Girdi

İlk satırda bir adet tam sayı T , gün sayısı.

Takip eden T satırda iki adet tam sayı N ve K , çorap çifti sayısı ve masanın tutabildiği çorap sayısı.

Çıktı

$1 \leq i \leq T$ için i . satırda i . gün için gereken beklenen kalori değeri.

Kısıtlar

- $1 \leq N, T \leq 2 \times 10^3$
- $T \leq S \leq 10^4$
- $1 \leq K \leq N$

Alt Görevler

Alt Görev 1 [4 puan]

- $N \leq 3$
- $S \leq 6$

Alt Görev 2 [7 puan]

- $N, S \leq 6$

Alt Görev 3 [9 puan]

- $N \leq 500$
- $S \leq 1000$
- $K = N$

Alt Görev 4 [21 puan]

- $N \leq 500$
- $S \leq 1000$

Alt Görev 5 [11 puan]

- $K = N$

Alt Görev 6 [48 puan]

- Ek Kısıt Yoktur

Örnek Girdi 1

```
3
1 1
2 1
2 2
```

Örnek Çıktı 1

```
1
333333337
3
```

Örnek Açıklama 1

1 çorap çifti için olası tek permütasyon 1 1 ve bu permütasyon için sadece 1 kalori harcamanız gerekiyor. $\frac{1}{1} \bmod (10^9 + 7) = 1$

2 çorap çifti için olası 6 adet permütasyon var:

- 1 1 2 2: $k = 1$ için 2 kalori, $k = 2$ için 2 kalori
- 1 2 1 2: $k = 1$ için 1 kalori, $k = 2$ için 3 kalori
- 1 2 2 1: $k = 1$ için 1 kalori, $k = 2$ için 4 kalori
 - $k = 1$ için örnek şu şekilde:
 - İlk çorapta masa boş, hiç kalori harcamadan ilk boşluğa koydunuz.
 - İkinci çorabı masadaki ilk çorapla karşılaştırarak 1 kalori harcadınız, sonrasında sağdaki boşluğa koymak istediniz fakat boş yer olmadığı için sinirlenip istifa ettiniz.
 - $k = 2$ için örnek şu şekilde:
 - İlk çorapta masa boş, hiç kalori harcamadan ilk boşluğa koydunuz.
 - İkinci çorabı masadaki ilk çorapla karşılaştırarak 1 kalori harcadınız, sonrasında sağdaki boşluğa koydunuz.
 - Üçüncü çorabı masadaki ilk çorapla karşılaştırmak için 1 kalori harcadınız, eşleşmedi, ikinci çorapla karşılaştırmak için tekrar 1 kalori harcadınız ve eşleştiği için o çifti ayırdınız.
 - Dördüncü çorabı masadaki ilk çorapla karşılaştırmak için 1 kalori harcadınız ve eşleştiği için o çifti ayırdınız.
- 2 1 1 2: $k = 1$ için 1 kalori, $k = 2$ için 4 kalori
- 2 1 2 1: $k = 1$ için 1 kalori, $k = 2$ için 3 kalori

- 2 2 1 1: $k = 1$ için 2 kalori, $k = 2$ için 2 kalori

Dolayısıyla $k = 1$ için beklenen deęer $\frac{8}{6} \bmod (10^9 + 7) = 333333337$ ve $k = 2$ için beklenen deęer $\frac{18}{6} \bmod (10^9 + 7) = 3$.

Örnek Girdi 2

```
1
5 3
```

Örnek Çıktı 2

```
961904778
```

Taş Köprü

Zaman kısıtı: 2 sn

Hafıza kısıtı: 128 MB

Büyük ve eski bir Antheia krallığında **N** tane şehir bulunmaktadır ve bu şehirler, her biri 1 birim uzunluğunda olan **N-1** tane eski taş köprüyle iki yönlü olarak birbirine bağlanmıştır. Krallıkta yaşayan insanlar bu köprüleri kullanarak herhangi bir şehirden diğerine ulaşabilmektedir.

Krallığın hükümdarı Kraliçe Emilyya, bu eski taş köprülerden birinin yıkılması durumunda ne olacağını düşünerek endişelenmektedir. Böyle bir durumda, krallık iki izole bölgeye ayrılacak ve insanlar her bir bölge içinde seyahat edebilecek, ancak bölgeler arasında geçiş yapamayacaktır. Bu olasılığa karşı önlem olarak, Kraliçe Emilyya **M** adet yeni ve dayanıklı iki yönlü demir köprü inşa ettirmiştir. Normalde, Antheia'da yaşayan insanlar şehirler arasında geçiş için sadece eski taş köprüleri kullanır, ancak bir köprü yıkıldığında, iki bölgeyi yeniden bağlamak için krallık yeni yapılan demir köprülerden birini kullanacaktır.

Daha açık olarak, eğer eski taş köprülerden biri yıkılırsa, krallık ikiye bölünmüş olur. Kraliçe Emilyya, iki izole bölgeyi yeniden birleştirebilecek en kısa yeni demir köprüyü seçmek istemektedir, böylece tüm şehirler arasında tekrar ulaşım sağlanabilecektir.

Antheia krallığında yaşayan ancak zamanının çok ilerisinde bir bilgisayar olimpiyatçısı olarak Kraliçe Emilyya'ya her bir eski taş köprünün çökmesi durumunda iki bölgeyi birleştirebilecek en kısa demir köprüyü bulmasında yardımcı olmanız istenmektedir.

Girdi Formatı

Girdi verisinin ilk satırı **N** ve **M** değerlerini içerir.

Sonraki **N-1** satırın her biri, iki tam sayı **p** ve **q** kullanarak krallıktaki orijinal bir taş köprüyü tanımlar; burada **p** \neq **q** olup köprü **p** ile **q** şehri arasında iki yönlüdür.

Takip eden **M** satırın her biri, üç tam sayı **p**, **q** ve **w** ile yeni yapılan demir köprüyü tanımlar; burada **w**, demir köprünün uzunluğudur. Herhangi bir şehir çifti arasında en fazla bir yol bulunur.

Çıktı Formatı

Girdideki sıraya uygun bir şekilde her bir **N-1** taş köprü için, bu köprü yıkılması durumunda krallığı yeniden bağlayacak en kısa demir köprünün uzunluğunu standart çıktıya yazın. Eğer uygun bir demir köprü yoksa, -1 yazdırın.

Limitler

- $2 \leq N \leq 50.000$
- $1 \leq M \leq 50.000$
- $0 < w \leq 10^9$
- $1 \leq p, q \leq N$

Alt Görevler

Alt Görev 1 (17 Puan)

- $1 \leq N, M \leq 5000$

Alt Görev 2 (34 Puan)

- $2 \leq N, M \leq 25.000$

Alt Görev 3 (49 Puan)

- Ek kısıt yoktur.

Örnek Girdi

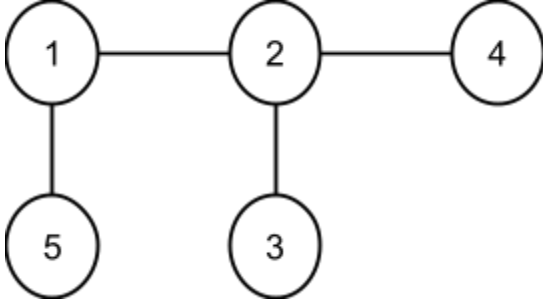
5 3
1 2
2 3
2 4
1 5
2 5 3
5 3 6
3 4 2

Örnek Çıktı

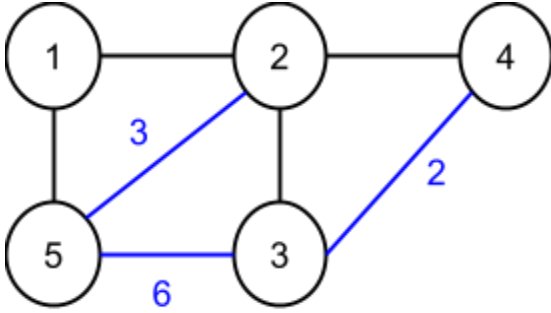
3
2
2
3

Örnek Açıklaması

Şehirler ve bunları birbirine bağlayan orijinal taş köprüler şu şekildedir:



Yeni eklenen demir köprüler ile birlikte krallık aşağıdaki şekildedir. Demir köprüler mavi ile gösterilmiştir.



Buna göre girdideki sıraya göre

- 1-2 arasındaki köprü yıkıldığında bunun için 3 ağırlığındaki demir köprü,
- 2-3 arasındaki köprü yıkıldığında bunun için 2 ağırlığındaki demir köprü,
- 2-4 arasındaki köprü yıkıldığında bunun için 2 ağırlığındaki demir köprü,
- 1-5 arasındaki köprü yıkıldığında bunun için 3 ağırlığındaki demir köprü

Seçilmelidir. Yani çıktı şu şekilde olmalıdır

3

2

2

3