

# TUBİTAK Ulusal Bilgisayar Olimpiyatları 2013

## II. Aşama Sınavı

### 2. Gün

TOBB Ekonomi ve Teknoloji Üniversitesi, Ankara

24 Kasım 2013

- Sınav süresi 5.5 saattir.
- Girdi-çıkıtı ve çalışma süresi/hafıza sınırları:

<i>Soru</i>	<i>Girdi</i>	<i>Çıkıtı</i>	<i>Süre sınırı</i>	<i>Hafıza sınırı</i>
Remzi	remzi.gir	remzi.cik	1 saniye	256 MB
Sirala	<i>interaktif</i>	<i>interaktif</i>	1 saniye	256 MB
Vialand	vialand.gir	vialand.cik	1 saniye	32 MB

#### Not:

Sınavda *partial-feedback* uygulaması vardır. Yani, sınav sırasında programınız önceden belirlenmiş çeşitli büyüklükte 4-5 örnek girdi-çıkıtı için denenip size bunlar üzerindeki puanınız söylenecektir. Bu uygulamanın amacı programınızdaki olası hataları daha kolay farketmenizdir. Gerçek girdiler üzerindeki puanınız örnekler üzerindeki puanınızdan farklı olabilecektir.

#### BAŞARILAR.

## KAŞIF REMZİ

Remzi bey yıllarca süren kooperatif ödemesinden kurtulup, yaşıda kemale erince emekli oldu. Şimdi tek bir kaygısı var o da dünyayı gezmek.

Bir takım takıntıları olan Remzi bey, 1 den M e kadar numaralanmış M ülkeyi kafasına göre 1 den N e kadar numaralanmış N bölgeye ayırdı ve dünyayı bölge bölge gezmeyi kafaya koydu. Remzi bey bölgeleri oluştururken kendince benzer olduğunu düşündüğü ülkeleri aynı bölgeye koyduğu için bir bölgedeki bir ülkeye dahi uğrasa kendisini bütün bölgeyi gezmiş sayıyor.

Gezmişken tam gezmek isteyen Remzi bey bazı bölgeleri birden fazla kez gezebileceği  $K \cdot P + 1$  (K ve P birer tamsayı) tane bölge numarası içeren bir liste hazırladı. Listede  $K+1$  adet sırabozucu vardır ve bunlar bulunduğu sıranın P moduna göre 1 değerine sahip olan elemanlarıdır. Yani, 1,  $P+1$ ,  $2P+1$ , ...,  $KP+1$  .nci sırada yer alan bölge numaraları sırabozucudur. Sırabozucular hariç geri kalan her  $P-1$  lik grup ise kendi içinde küçükten büyüğe sıralıdır. Yani (sınırlar dahil olmak üzere), 2 ile P,  $P+2$  ile  $2P$ ,  $2P+2$  ile  $3P$ , ...,  $(K-1)P+2$  ile  $KP$  sıraları arasında yer alan her  $P-1$  tane eleman kendi içinde küçükten büyüğe sıralıdır. Örneğin,  $K=2$  ve  $P=4$  için uygun bir liste şu şekildedir:

7 2 9 11 8 4 9 12 3

burada birinci grup (2 9 11) ve ikinci grup (4 9 12) kendi içinde küçükten büyüğe sıralıdır, 7 8 3 ise sırabozucu elemanlardır.

Remzi bey bu gezinin bitmesini hiç istememektedir, ama çok uzun sürsün de istememektedir. Yani bildiğiniz kafası karışıktır. Listede bir bölgeden sıradaki bölgeye olabildiğince hızlı gitmek istemektedir. Ama bu durum gezintisini erken bitireceği için de üzülmektedir. Remzi bey bu yüzden gezisini uzatabilmek için listesinde **sırabozucu elemanlar dışındakileri** yukarıda bahsedilen her bir grup için sıralama kuralını bozmadan sizin değiştirmenize izin vermiştir.

Sizden istediği ise şudur: Listeyi her bir bölgeden sıradakine en kısa yoldan geçildiğinde toplamda en uzun sürenin harcanacağı şekilde değiştirdiğinizde, gezinin ne kadar süreceğini bulmanız.

Son olarak bütün bu süre hesaplarına sırabozuculardan bir sonraki bölgeye geçme süresini saymayacaksınız. Yani sırabozuculardan bir sonraki bölgeye geçmek hiç zaman almıyor gibi kabul edebilirsiniz.

### Girdi (remzi.gir)

- İlk satırda sırayla N, M, P, K, Q sayıları
- Sonraki Q satırın her birinde ülkeler arasındaki yolları belirten a, b ve c tamsayıları (a ve b şehirleri arasında c uzunluğunda çift yönlü bir yol var, c negatif olamaz).
- Sonraki N satırın her birinde sıradaki bölge için, o bölgedeki ülke sayısı s ve sonrasında s tane ülke numarası. (Bir ülke iki bölgede birden bulunamaz ve her bölgede en az bir ülke bulunacaktır.)
- Sonraki satırda  $K+1$  tane bölge numarası (sırabozucular).

### Çıktı (remzi.cik)

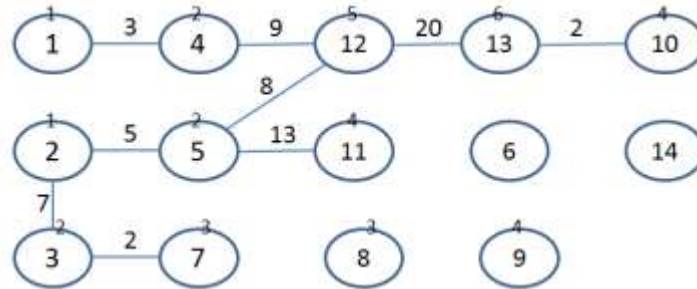
- Tek satırda bir tamsayı: Gezinin ne kadar süreceği.

Listede sıradaki bölgeye olabilecek en kısa sürede ulaşmak için başka bölgeler ve ülkeler üzerinden geçebilirsiniz, buraları dolaşma süreniz toplam süreye sayılmasına rağmen o ülkeleri/bölgeleri gezmiş sayılmazsınız.

### Örnek Girdi/Çıktı

remzi.gir

```
6 14 3 2 9
1 4 3
2 5 5
2 3 7
3 7 2
4 12 9
12 13 20
5 11 13
5 12 8
10 13 2
2 1 2
3 3 4 5
2 7 8
3 9 10 11
1 12
1 13
1 6 3
```



Ülke numaraları düğümlerin içine  
bölge numaraları ise üzerine  
yazılmıştır.

remzi.cik

78

### Örnek girdinin çözümü

1 (sirabozucu)  
4  
5  
6 (sirabozucu)  
5  
6  
3 (sirabozucu)

### Çözümün açıklaması

1.bölgeden 4.bölgeye **0** ile gideriz.( Sırabozuculardan bir sonraki bölgeye geçme süresini saymayacaksınız) **+0**

4.bölgeden 5.bölgeye gitmek için önce 4.bölgeden 2.bölgeye **13** uzunlugundaki(5-11 nolu ülkeler arası) yolu kullanarak sonra da 2.bölgeden 5. bölgeye **8** uzunluğundaki(5-12 nolu ülkeler arası) yolu kullanarak gideriz. **+21**

5.bölgeden 6.bölgeye **20** uzunluğundaki(12-13 nolu ülkeler arası) yolu kullanarak gideriz. **+20**

6.bölgeden 5.bölgeye **0** ile gideriz.( Sırabozuculardan bir sonraki bölgeye geçme süresini saymayacaksınız) **+0**

5.bölgeden 6.bölgeye **20** uzunluğundaki(12-13 nolu ülkeler arası) yolu kullanarak gideriz. **+20**

6.bölgeden 3.bölgeye, önce 6.bölgeden 4.bölgeye **2** uzunluğundaki(13-10 nolu ülkeler arası) yolu kullanarak gideriz, sonra 4.bölgeden 2.bölgeye **13** uzunluğundaki(11-5 nolu ülkeler arası) yolu kullanarak gideriz, sonra 2.bölgeden 3.bölgeye **2** uzunluğundaki(3-7 nolu ülkeler arası) yolu kullanarak gideriz. **+17**

**Toplam → 0+21+20+0+20+17= 78**

### **Notlandırma**

#### **Altgörev 1 (30 puan)**

$N \leq 10, M \leq 50, K * P + 1 \leq 13, Q \leq 100$

#### **Altgörev 2 (20 puan)**

$N \leq 1000, M \leq 5000, P = 2, K \leq 1000, Q \leq 10000$

#### **Altgörev 3 (20 puan)**

$N \leq 1000, M \leq 5000, P \leq 100, K = 1, Q \leq 10000$

#### **Altgörev 4 (30 puan)**

$N \leq 1000, M \leq 5000, P \leq 100, K \leq 1000, Q \leq 10000$

## SIRALA BAKALIM

Ali'nin arkadaşı Ömer'in elinde  $N$  ( $10 \leq N \leq 1.000$ ) tane kutu vardır. Ömer her kutuyu belli bir miktar para koymuştur ve hiçbir iki kutudaki para miktarı aynı değildir. Ömer kutuların para değerleri sıralamasını bilmektedir ve Ali ile bir oyun oynamak istemektedir. **Kutular 0 ile  $N-1$  arasında numaralanmıştır.**

Ali sırayla tam olarak  $K$  ( $\sqrt{N} \leq K \leq N$ ) tane kutuyu seçip, bu  $K$  tane kutunun kendi aralarındaki küçükten büyüğe sıralamasını Ömer'den öğrenecektir. Bütün kutuları olabildiğince en az hamleyle küçükten büyüğe sıralayınca oyun bitecektir ve hamle sayısının azlığına göre puan alacaktır.

Ali bu oyunu en iyi olduğu ispatlanmış bir strateji ile oynamaktadır. Olabilecek en az sorguda sıralamayı bulmakta ve tam puan almaktadır. Sizden istenen Ali'nin yerine bu oyunu Ömer ile oynayacak interaktif bir program yazmanız ve olabildiğince çok puan (mümkünse Ali gibi tam puan) almanız.

### Örnek Oyun

Farz edelim ki elimizde 6 tane kutu olsun ve Ali her seferinde 3 kutu sorabilsin. Örnek bir oyun şu şekilde olabilir:

Ali'nin sorguları	Ömer'in kutuları sıralamış hali
0 1 2	2 1 0
3 4 5	3 4 5
2 3 4	3 2 4
1 0 5	1 5 0
1 4 2	2 4 1

Ali beş tane sorgu yaparak kutuların küçükten büyüğe sırasının 3 2 4 1 5 0 olduğunu bulmuştur.

### Notlandırma

Kodunuz en iyi şekilde oynayan bir Ömer programı ile oynayacaktır. Beş tane  $N$  ve  $K$  girdi paketi olacaktır. Her girdi paketinde 20 tane rastgele  $N$ 'in permütasyonu olan bir dizilim için kodunuz oyunu oynayacaktır. Her paketteki en uzun süren sorgu sayısı, Ali'nin yaptığı sorguyla eşit sayıda ise 20 puan alacaksınız. Eğer daha fazla sorgu yaparsanız  $\left\lfloor 16 - \left\lfloor \frac{(\text{Sizin sorgu sayınız} - \text{Ali'nin sorgu sayısı})}{(\text{Ali'nin sorgu sayısı})} \right\rfloor \right\rfloor$  puan alacaksınız.

- Yanlış bir dizilim vermişseniz veya Ömer'in hala farklı bir dizilimi tutma ihtimali varsa,
- Yanlış bir sorgu vermişseniz (sorgunun uzunluğu  $K$ 'ya eşit değilse, sorguda tekrar eden sayı varsa)
- 20.000 sorgudan fazla sorgu yaparsanız 0 puan alacaksınız. Paketlerin  $N$  ve  $K$  değerleri aşağıda verilmiştir.

	N değeri	K değeri
1.Paket:	20	10
2.Paket:	50	10
3.Paket:	100	10
4.Paket:	200	16
5.Paket:	1000	70

### Kodlama Detayları

- Kodunuzun başında `#include "omer.h"` yazarak Ömer'i dahil etmeniz gerekiyor.

- `int sizeN();` ve `int sizeK();` fonksiyonlarını çağırarak N'i ve K'yı öğrenmeniz gerekiyor. Önce, `sizeN()` sonra `sizeK()` fonksiyonlarını çağırmanız gerekiyor.

- Sorgu yapmak için `void sortSet(int *)`; fonksiyonuna K boyutlarındaki bir dizinin adresini göndermeniz gerekmektedir. Sayılar 0 ve N-1 arasında olması, tekrar etmemesi gerekmektedir ve tam olarak K tane sayı içermelidir. Bu fonksiyon dizinizi sıralayacaktır. Diziniz 0'ıncı indisten başlayıp K-1'de bitmelidir.

- Tüm sayıları sıraladığınızda emin olduktan sonra `void solution(int *)` fonksiyonuna N boyutlarında bir dizinin adresini vermelisiniz. Bu fonksiyonu kodun sonunda ve tam olarak bir kere çağırmalısınız. Diziniz 0'ıncı indisten başlayıp, N-1'inci indiste bitmeli.

-Süre sınırı her paketteki her bir oyun için 1sn'dir.

### Değerlendirme

Çözümünüzün olduğu klasöre `omer.h` ve `omer.o` dosyalarını indirip derleme ve çalıştırma komutlarını şu sırayla izleyiniz.

*Derleme:* `g++ cozum.cpp omer.o -o cozum`

**Not:** Çözümünüz kesinlikle `.cpp` uzantılı olmalıdır.

*Çalıştırma:* `./cozum`

**Not:** Program ekrandan okuyup ekrana yazacaktır.

**Not2:** Çalıştırdığınız programa sırasıyla;

- İlk satıra N ve K değerlerini
- İkinci satırda N tane 0 ile N-1 arasındaki dizinin en son halini belirten sayıları
- Son satırda ise sizden bulunan en iyi çözüm beklenmektedir

O en iyi çözüm ve sizin çözüm kodunuzun çözümüne göre puanınızın 20 ye bölünmüş hali çıktı olarak size verilecektir.

*Örnek Çalıştırma:*

6 3  
3 2 4 1 5 0  
5

Yukarıda verilen örneğin girdi biçimine uygun halidir. Bu girdiyi denedikten sonar çıktıda  $[0,1]$  aralığında bir sayı olacaktır. 0'dan 1'e doğru gittikçe çözümünüzün daha doğru olduğunu anlayabilirsiniz.

## VİALAND

Vialand günümüz dünyasına tepki duyarak kaçan birkaç aktivist tarafından çok çok uzakta kurulmuş yeni bir gezegen ve aslında yeni, küçük ve tertemiz bir dünyadır. Hatta öyle ki varlığın da yokluğun da paylaşıldığı; kinin, nefretin olmadığı; her şeyin adaletle paylaşıldığı; sevginin esas olduğu bir dünyadır Vialand. Günümüz ataerkil ve keşmekeş dünya toplumuna inat Vialand tek bir ülke halinde ve bir prenses tarafından yönetilmektedir. Prensese Sena, Vialand'ın en büyük yöneticisi olarak görülmektedir ve Prensese halkı tarafından çok sevilmektedir. Vialand **N** adet şehirden oluşmakta ve bu **N** şehir **N-1** adet yolla birbirine bağlıdır (yani her iki şehir arasında en az bir yol var) .

Prensese Sena ülkesinin günümüz dünyasına inat yemyeşil ve tertemiz tutmaya çalışmaktadır. Hatta öyle ki bu durumu yakın tarihte ziyarete gelecek olan DB (Dünyalılar Birliği) başkanına da göstermek istemektedir. Ve bu gösteriyi ufak bir gezintiyle yapmak isteyen Prensese Sena ülkesindeki en güzel (yeşillik, temiz) yolu bulup DB başkanını o güzergâhtan gezdirmek istemektedir.

Prensese Sena için, seçilen bir güzergâhın **güzellik derecesi güzergâh üzerindeki ağaçlı yolların sayısının ağaçsız yolların sayısından ne kadar fazla** olduğudur. Ancak istediği güzellik derecesi formatında en güzel yolu bulmakta zorlanan Prensese bu konuda sizden yardım istemektedir. Sizin göreviniz Prensese Sena'nın isteğine göre seçilebilecek en güzel yolun güzellik derecesini ona söylemektir.

### Girdi (vialand.gir)

- İlk satırda **N** sayısı
- Sıradaki **N-1** satırda 3 adet tamsayı **a, b, c** ( $2 \leq a, b \leq N$  ve  $0 \leq c \leq 1$ ) **a** ile **b** arasında yol olduğunu ve **c** sayısının durumuna göre bu yolda ağaç olup olmadığını belirtir. Yani, **c=1** ise o yol ağaçlı ve **c=0** ise o yol ağaçsızdır. En az bir tane ağaçlı yol olacağı garanti edilmektedir.

### Çıktı (vialand.cik)

- Tek satırda bir tamsayı: Maksimum güzellik derecesine sahip yolun güzellik derecesi.

### Örnek Girdi/Çıktı

**vialand.gir**

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

**vialand.cik**

```
3
```



## **Notlandırma**

### **Altgörev 1 (10 puan)**

$$2 \leq N \leq 300$$

### **Altgörev 2 (25 puan)**

$$2 \leq N \leq 5000$$

### **Altgörev 3 (20 puan)**

$2 \leq N \leq 100000$  ve Vialand'da tüm yollar ağaçlıdır (ağaçsız hiçbir yol yok)

### **Altgörev 4 (45 puan)**

$$2 \leq N \leq 100000$$