



Mantarları Sayma (mushrooms)

Mantar uzmanı Andrew sadece Singapurda yetişen mantarlar konusunda bir araştırma yapmaktadır.

Araştırması kapsamında, 0'dan $n - 1$ 'e numaralandırılmış n tane mantar toplar. Her bir mantar A ve B olarak adlandırılmış iki mantar türünden birisindendir.

Andrew **0 no'lu mantarın A türünden olduğunu bilmektedir**, ama her iki tür de tamamen aynı gözüktüğünden, 1'den $n - 1$ 'e numaralandırılmış olan mantarların türlerini bilmemektedir.

Neyse ki, labında, Andrew'ya bu konuda yardımcı olabilecek bir makinesi vardır. Bu makineyi kullanmak için, iki ya da daha fazla mantar herhangi bir sırada tek sıra olarak makineye yerleştirilir ve makine çalıştırılır. Daha sonra makine farklı türlerden kaç tane mantar ikilisinin bu sıralamada **ardışık** olduğunu hesaplar. Örnek vermek gerekirse, eğer makineye şu sırada $[A, B, B, A]$ mantarlarını koyarsanız sonuç 2 olacaktır.

Ne yazık ki bu makineyi çalıştırmak çok pahalıdır ve bu nedenle makine kısıtlı sayıda kullanılabilir. Buna ek olarak makineye bütün çalıştırmalarda konulabilecek toplam mantar sayısı da 100 000'i aşmamalıdır. Bu makineyi kullanarak Andrew'ya A türünden olan mantar sayısını bulmasında yardımcı olunuz.

Implementasyon detayları

Aşağıdaki fonksiyonu implement etmelesiniz:

```
int count_mushrooms(int n)
```

- n : Andrew'nun topladığı mantar sayısı.
- Bu fonksiyon tam olarak bir kez çağrılır ve A türünden olan mantarların sayısını dönmelidir.

Bu fonksiyon aşağıdaki fonksiyonu çağırabilir:

```
int use_machine(int[] x)
```

- x : sınırlar dahil 2 ve n arasında bir uzunluğa sahip bir dizi. Bu dizi, dizideki sıra ile makineye konulacak mantarların etiketlerini gösterir.
- x dizisinin elemanları, sınırlar dahil 0'dan $n - 1$ 'e **birbirinden farklı** tamsayılardır.
- x dizisinin uzunluğu d olsun. Bu durumda bu fonksiyon şu şartı sağlayan tüm farklı j indekslerinin sayısını döner: $0 \leq j \leq d - 2$, $x[j]$ ve $x[j + 1]$ farklı türden mantarlardır.
- Bu fonksiyon en fazla 20 000 kez çağrılabilir.

- `use_machine` fonksiyonuna gönderilen x 'lerin toplam uzunluğu 100 000'i geçemez.

Örnekler

Örnek 1

Sıra ile $[A, B, B]$ türlerinde 3 tane mantar toplanmış olsun. `count_mushrooms` fonksiyonu aşağıdaki gibi çağrılır:

```
count_mushrooms(3)
```

Bu fonksiyon `use_machine([0, 1, 2])` çağrısını yapabilir ve bu örnekte bu çağrı 1 döner. Daha sonra `use_machine([2, 1])` çağrısını yapabilir ve bu da 0 döner.

Bu noktada A türünden 1 tane mantar olduğunu anlayacak bilgiye sahip oluruz ve bu nedenle `count_mushrooms` 1 dönmelidir.

Örnek 2

Sıra ile $[A, B, A, A]$ türlerinde 4 tane mantar toplanmış olsun.

`count_mushrooms` fonksiyonu aşağıdaki gibi çağrılır:

```
count_mushrooms(4)
```

Bu fonksiyon `use_machine([0, 2, 1, 3])` çağrısını yapabilir ve bu çağrı 2 döner. Daha sonra `use_machine([1, 2])` çağrısını yapabilir ve bu da 1 döner.

Bu noktada A türünden 3 tane mantar olduğunu anlayacak bilgiye sahip oluruz ve bu nedenle `count_mushrooms` 3 dönmelidir.

Kısıtlar

- $2 \leq n \leq 20\,000$

Puanlama

Eğer herhangi bir test case'de `use_machine`'e yapılan çağrılar yukarıdaki kurallara uymazsa ya da `count_mushrooms` fonksiyonunun döndüğü değer yanlışsa puanınız 0 olur. Diğer durumlarda, Q bütün test case'lerde `use_machine` fonksiyonuna yapılan çağrı sayılarının maksimumu olsun. Bu durumda puanınız aşağıdaki tabloya göre hesaplanacaktır:

Durum	Puan
$20\,000 < Q$	0
$10\,010 < Q \leq 20\,000$	10
$904 < Q \leq 10\,010$	25
$226 < Q \leq 904$	$\frac{226}{Q} \cdot 100$
$Q \leq 226$	100

Bazı test case'lerde grader'ın davranışı duruma göre değişiklik gösterebilir (adaptive). Bu şu demektir: bazı test case'lerde önceden belirlenmiş bir mantar türü dizisi yoktur. Onun yerine grader'ın verdiği cevaplar `use_machine`'e daha önce yapılmış çağrılara göre değişebilir. Ama dikkatinizi çekmek isteriz ki, her bir etkileşimden sonra grader'ın vereceği cevabın şimdiye kadarki verilen tüm cevaplarla da tutarlı olacağı ve bütün cevaplarla tutarlı en az bir mantar dizisi olacağı garanti edilmektedir.

Örnek grader

Örnek grader mantar türlerini gösteren bir s tamsayısı dizisi okur. Her bir $0 \leq i \leq n - 1$ için, $s[i] = 0$ olması i . mantarın A türünden olduğunu, $s[i] = 1$ olması da i . mantarın B türünden olduğunu göstermektedir. Örnek grader girdiyi aşağıdaki formatta okur:

- satır 1: n
- satır 2: $s[0] \ s[1] \ \dots \ s[n - 1]$

Örnek grader'ın çıktısı aşağıdaki formattadır:

- satır 1: `count_mushrooms` fonksiyonunun döndüğü değer.
- satır 2: `use_machine` fonksiyonuna yapılan çağrı sayısı.

Örnek grader'ın adaptive olmadığına dikkatinizi çekmek isteriz.