



International Olympiad in Informatics 2016

12-19th August 2016

Kazan, Russia

day1_2

railroad

Country: TKM

Anna bir disney parkta çalışıyor ve o yeni bir roller coaster oyunu için demiryolu inşa etmekle görevli. O çoktan n özel bölmeden oluşan (0 -dan $n - 1$ kadar numaralandırılmış) roller coaster tren yolu dizayn etmiş; tepeleri, fren kısımlar ve diğer kısımlar. O bunları biraraya getirip roller coaster tren yolunun son halini dizayn etmesi lazım.

0 ve $n - 1$ kapalı aralığı her i , i özel bölmesinin iki özelliği var;

- bölmeye girildiğinde bir hız limiti var; trenin hızı **en fazla** s_i km/h olması lazım.
- o bölmeden çıkarken girerkenki hızı ne olursa olsun trenin hızı **tam olarak** t_i km/h.

Tren yolu bittiğinde o n bölmenin her birisi sadece bir kere kullanılmış olması lazım. Dahası, arka arkaya gelecek her iki bölmenin arasında bir bağlantı kısım olacak. Anna n özel bölmelerin diziliş sırasını seçmesi ve bağlantı kısımlarının her birisinin uzunluğunun kaç metre olacağına karar vermesi lazım. Bir bağlantı kısmının uzunluğu metreyle ölçülüyor ve negatif tamsayı olmaması lazım. (sıfır da olabilir),

İki bölmenin arasındaki bağlantının her bir metresi treni 1 km/h yavaşlatıyor. En başta, tren ilk bölmeye 1 km/h hızla giriyor.

Dizaynın son hali sıradaki şartları sağlaması lazım;

- tren her bölmeye girerken hiç hız limiti ihlal etmemesi lazım;
- trenin hızı en son bölmenin sonuna gelene kadar her an pozitif olmalıdır.

subtask 3 hariç diğer tüm subtasklerde yapmanız gereken, n özel bölmenin sırasını ve her bölme arasındaki bağlantıların uzunluğunu bulmak, şöyleki bağlantıların toplam uzunluğu mümkün mertebe az olmalı. Subtask 3-de sizin yapmanız gereken, bütün bağlantı uzunlukları sıfır olan bir tren yolu olup olamayacağına bakmalısınız.

Implementation details

You should implement the following function (method):

- `int64 plan_roller_coaster(int[] s, int[] t)`
 - s : n uzunluğun sırası, izin verilen maksimum giriş hızlar.
 - t : n uzunluğun sırası, çıkış hızlar.
 - Fonksiyon bölmeler arası bağlantı uzunluklarının toplamının olabilecek en küçük değerini çıkarması lazım. (subtask 3-de eğer cevap sıfır değilse herhangi bir tamsayı yazabilirsiniz, detaylar için subtasks bölümüne bakın.

For the C language the function signature is slightly different:

- `int64 plan_roller_coaster(int n, int[] s, int[] t)`

- n : s ve t -deki eleman sayısı (yani özel bölme sayısı),
- diğer parametreler aynı

Example

`int64 plan_roller_coaster([1, 4, 5, 6], [7, 3, 8, 6])`

Bu örnekte dört özel bölme var. En iyi çözüm onları **0,3,1,2**, şeklinde dizmek ve bağlantı uzunluklarının da sırasıyla **1,2,0** şeklinde olması. Bu trenin nasıl ilerlediğini gösteriyor;

- Trenin başlama hızı **1** km/h.
- Tren yolculuğuna **0** özel bölmesine girerek başlıyor.
- Tren **0** bölmesinden **7** km/h hızla çıkıyor.
- Sonra **1** m uzunluğunda bir bağlantı var. Bağlantının sonuna vardığında hızı **6** km/h oluyor.
- Tren **3** özel bölmesine **6** km/h hızla giriyor ve aynı hızla ayrılıyor.
- **3** bölmeden çıkınca **2** m uzunluğunda bir bağlantıdan geçiyor. Hızı da **4** km/h oluyor.
- Tren **1** bölmeye **4** km/h hızla giriyor ve **3** km/h hızla çıkıyor.
- **1** bölmeden çıkınca hemen **2** bölmeye giriyor.
- Tren **2** bölmeden çıkıyor. Son hızı **8** km/h.

Fonksiyon bağlantıların toplam uzunluğunu $1 + 2 + 0 = 3$ ieklinde çıkarması lazım.

Subtasks

In all subtasks $1 \leq s_i \leq 10^9$ and $1 \leq t_i \leq 10^9$.

1. (11 points): $2 \leq n \leq 8$,
2. (23 points): $2 \leq n \leq 16$,
3. (30 points): $2 \leq n \leq 200\,000$. Bu subtaskde programın sıfır cevabını çıkarıp çıkarmadığına bakmamız lazım. Eğer cevap sıfır değilse, herhangi bir pozitif tamsayı doğru kabul edilecek.
4. (36 points): $2 \leq n \leq 200\,000$.

Sample grader

The sample grader reads the input in the following format:

- line 1: integer n .
- line $2 + i$, for i between 0 and $n - 1$: integers s_i and t_i .