

TÌM SỐ

Cho một số nguyên dương n . Bạn hãy tìm một số nguyên dương X lớn nhất sao cho:

- ✿ X là số chính phương.
- ✿ Phân tích được X thành tích các thừa số nguyên dương đôi một khác nhau và không vượt quá n :
$$X = a_1 \times a_2 \dots \times a_k \text{ sao cho } 0 < a_1 < a_2 < \dots < a_k \leq n$$

Dữ liệu: Vào từ file văn bản SQRNUM.INP gồm nhiều test

- ✿ Dòng đầu ghi số t là số lượng test ($t \leq 2000$)
- ✿ t dòng tiếp theo, mỗi dòng ghi một số nguyên dương n ($n \leq 10^7$)

Kết quả: Ghi ra file văn bản SQRNUM.OUT tương ứng với mỗi test trong file input, ghi số X tìm được. Vì kết quả có thể rất lớn nên bạn chỉ cần đưa ra phần dư của X khi chia cho $10^9 + 7$.

Ví dụ:

SQRNUM.INP	SQRNUM.OUT
4	1
1	518400
10	144
7	13653622
32	

TẬP DẪY CON

Một dãy con của một chuỗi S là chuỗi tạo thành từ S bằng cách giữ nguyên chuỗi S hoặc xóa đi một vài ký tự và giữ nguyên thứ tự các ký tự khác. Ví dụ với S là "abaca", những chuỗi "abaca", "aba", "aaa", "a" và "" là những dãy con của S nhưng những chuỗi "aabaca", "cb", "bcaa" không phải là dãy con của S .

Bạn được cho chuỗi S và tập Q rỗng. Bạn được lấy bất kỳ một dãy con t của S để thêm vào tập Q nếu nó chưa có trong Q . Chi phí để thêm chuỗi t vào tập Q bằng $n - |t|$, ở đây $|t|$ là chiều dài của chuỗi t .

Nhiệm vụ của bạn là tìm chi phí ít nhất để xây dựng tập Q chứa đúng k chuỗi.

STRSET.INP

Dòng 1 chứa hai số nguyên dương $n \leq 100$; $k < 10^{12}$

Dòng 2 chứa chuỗi S chỉ gồm n chữ cái latin thường

STRSET.OUT

Một số nguyên duy nhất là chi phí xây dựng tập Q hoặc -1 nếu không thể xây dựng tập Q

Ví dụ

STRSET.INP	STRSET.OUT
4 5 asdf	4
5 6 aaaaa	15
5 7 aaaaa	-1
10 100 ajihushda	233

PHÉP TOÁN BITWISE AND

Cho dãy số nguyên dương $A = (a_1, a_2, \dots, a_n)$ và một số nguyên z . Với mỗi số nguyên không âm $x \in [0; 2^z - 1]$, hãy tìm phần tử nhỏ nhất a_i trong dãy A thỏa mãn: $a_i \& x == 0$.

Dữ liệu: Vào từ file văn bản BITAND.INP

- ✿ Dòng 1 chứa hai số nguyên n, z ($1 \leq n \leq 10^6; 0 \leq z \leq 20$)
- ✿ Dòng 2 chứa n số nguyên dương a_1, a_2, \dots, a_n (với: $1 \leq a_i < 2^z$)

Kết quả: Ghi ra file văn bản BITAND.OUT 2^z số nguyên trên một dòng là các kết quả ứng với $x = 0, 1, \dots, 2^z - 1$

Ví dụ

BITAND.INP	BITAND.OUT
5 3	1 2 1 4 1 2 1 -1
2 4 1 7 6	

ĐƯA THƯ

Tại vương quốc X thời xưa, việc chuyển phát thư tín được thực hiện bằng ngựa. Kinh đô nằm ở đầu con đường chính đi dọc vương quốc, dọc trên con đường có n trạm ngựa đánh số từ 1 tới n . Trong mỗi trạm có một con ngựa làm nhiệm vụ đưa thư.

Khi một người cần đưa thư giữa hai trạm, anh ta sẽ lên con ngựa tại trạm xuất phát. Mỗi khi đi tới một trạm trên đường đi, anh ta có thể đi tiếp (bằng con ngựa đang dùng) hoặc đổi sang con ngựa của trạm mới tới và đi tiếp, thời gian đổi ngựa cố định là Δ giây mỗi lần đổi.

Biết rằng với mỗi trạm i ($i = 1, 2, \dots, n$), vị trí của trạm nằm cách kinh đô l_i km, con ngựa của trạm đi mỗi km mất a_i giây, người đưa thư được dùng con ngựa này tối đa 1 lần và con ngựa không thể đi được nhiều hơn k_i km nếu được sử dụng.

Yêu cầu: Người đưa thư cần chuyển một công lệnh khẩn từ trạm s tới trạm t . Hãy giúp anh ta tìm cách chuyển với thời gian nhanh nhất.

Dữ liệu: Vào từ file văn bản DELIVER.INP

✿ Dòng đầu tiên chứa số nguyên $C \leq 100$ là số bộ dữ liệu, tiếp theo là C nhóm dòng, mỗi nhóm dòng là một bộ dữ liệu, trong mỗi nhóm:

✿ Dòng 1 chứa 4 số nguyên dương $n \leq 10^5$; $\Delta \leq 10^9$; s, t ($1 \leq s, t \leq n$; $s \neq t$)

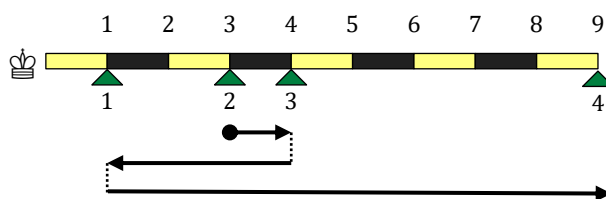
✿ n dòng tiếp theo, dòng thứ i chứa 3 số nguyên dương l_i, a_i, k_i ($l_i, a_i, k_i \leq 10^9$)

Các số trên một dòng của dữ liệu vào được ghi cách nhau bởi dấu cách.

Kết quả: Với mỗi bộ dữ liệu, ghi ra file văn bản DELIVER.OUT một số nguyên duy nhất trên một dòng là thời gian (tính bằng giây) để đưa thư từ s tới t theo phương án tìm được. Ghi ra số -1 nếu không tồn tại cách chuyển thư từ s tới t .

Ví dụ

DELIVER.INP	DELIVER.OUT
3	64
4 1 2 4	40
1 3 9	28
3 20 9	
4 6 4	
9 9 9	
4 2 1 4	
1 16 4	
2 12 4	
3 8 4	
4 4 4	
4 2 4 1	
1 16 1	
2 12 1	
3 8 1	
4 4 1	



Giải thích về ví dụ cho bộ dữ liệu 1:

Dùng ngựa trạm 2 chạy tới trạm 3: $20 \times 1 = 20$ giây

Đổi sang ngựa trạm 3: 1 giây

Dùng ngựa trạm 3 chạy về trạm 1: $6 \times 3 = 18$ giây

Đổi sang ngựa trạm 1: 1 giây

Dùng ngựa trạm 1 chạy tới trạm 4: $3 \times 8 = 24$ giây

Tổng cộng: 64 giây