1. Xếp "Cột trụ"

Cho một tập các hình trụ $V = \{V_1, V_2, \ldots, V_n\}$ trong đó:

- **i)** hình trụ V_i có chiều cao h_i, bán kính r_i là các số nguyên thỏa mãn $0 < h_i, r_i \leq 10^6$ ($i = 1, 2, \ldots, n$)
- **ii)** các hình trụ có bán kính đôi một khác nhau.

Các hình trụ có thể xếp chồng lên nhau để tạo thành một **"Cột trụ"** theo quy tắc: Chỉ hình trụ có bán kính nhỏ hơn có thể được xếp chồng lên hình trụ có bán kính lớn hơn, nghĩa là với mọi $V_i, V_j \in V$: V_i xếp chồng được lên V_j khi và chỉ khi $r_i < r_j$. Chiều cao của **"Cột trụ"** là tổng chiều cao của các hình trụ được xếp trong **"Cột trụ"** đó.

Yêu cầu: Đối với chiều cao h nguyên dương cho trước, tính số lượng hình trụ cần ít nhất để xếp thành **"Cột trụ"** có chiều cao không nhỏ hơn h.

Dữ liệu: Vào từ file văn bản VOL.INP chứa nhiều test:

- Dòng đầu tiên chứa chứa số t cho biết số lượng test. Các dòng tiếp theo chứa các test.
- Dòng đầu tiên của test chứa số n và h lần lượt là số lượng hình trụ và chiều cao giới hạn của **"Cột trụ"**
- n dòng tiếp theo trong test, mỗi dòng chứa hai số h_i và r_i cho biết chiều cao và bán kính của hình trụ thứ i trong test.

Kết quả: Ghi ra file văn bản VOL.OUT chứa nhiều dòng, mỗi dòng một số nguyên cho biết kết quả của mỗi test, nếu không có cách xếp trụ in ra -1.

Ví dụ:

VOL.INP	VOL.OUT
2 3 2 1 5 2 3 3 1 4 8 2 5 3 2 1 3 1 1	1 -1

2. XÂU CƠ SỞ

Lũy thừa nguyên bậc n của một xâu là việc lặp lại liên tiếp n lần xâu đó.

Ví dụ $(abc)^3 = abcabcabc$.

Xâu cơ sở của 2 xâu S và T là xâu q có độ dài lớn nhất sao cho $S = q^i$ và $T = q^j$ $(i, j \in N)$

Yêu cầu: Cho 2 xâu khác rỗng S và T có độ dài không quá 10^6 và chỉ chứa các ký tự latinh thường. Hãy xác định xâu cơ sở của S và T. Nếu không tồn tại xâu cơ sở thì đưa ra ***NO***.

Dữ liệu: Vào từ file văn bản **BASESTRING.INP**:

- Dòng thứ nhất chứa xâu S.
- Dòng thứ 2 chứa xâu T.

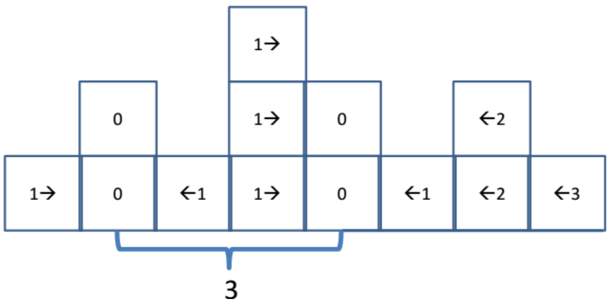
Kết quả: Đưa ra file văn bản **BASESTRING.OUT** xâu q hoặc thông báo ***NO***.

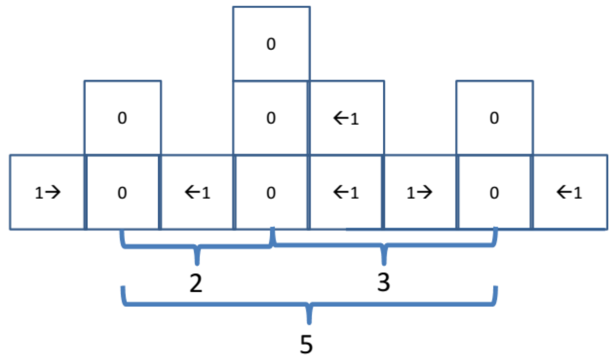
Ví dụ:

BASESTRING.INP	BASESTRING.OUT
aaa aa	a

3. Dồn đống

Có N cột liên tiếp nhau $1, 2, 3, \dots, N$. Cột thứ i gồm h_i khối hộp chồng lên nhau. Chi phí để di chuyển một khối sang một vị trí là 1 đơn vị. Người ta muốn dồn tất cả các khối hộp vào một số cột sao cho khoảng cách giữa 2 cột khác nhau bất kỳ còn khối hộp là một số nguyên tố.





Ví dụ, với dãy cột $1, 2, 1, 3, 2, 1, 2, 1$, hình bên trái thể hiện một cách di chuyển với chi phí là 13. Hình bên phải thể hiện một cách di chuyển với chi phí là 6.

Yêu cầu: Xác định chi phí nhỏ nhất để di chuyển các khối hộp thỏa mãn điều kiện trên.

Dữ liệu: vào từ file **MOUNTAIN.INP**

- Dòng đầu chứa hai số nguyên dương N ($N \le 30000$)
- Dòng thứ hai chứa hai N số nguyên $h_1, h_2, \dots, h_N$ $(0 \le h_i \le 10^3)$.

Kết quả: Ghi ra file **MOUNTAIN.OUT** một số nguyên duy nhất là chi phí nhỏ nhất tìm được.

MOUNTAIN.INP	MOUNTAIN.OUT
2 1 2	1
3 0 0 0	0
8 1 2 1 3 2 1 2 1	6

4. Hàm giao hoán

Hai hàm số f và g $(f, g: X \to X)$ được gọi là giao hoán nhau nếu $f\big(g(x)\big) = g\big(f(x)\big)$ $\forall x \in X$. Ví dụ hàm $f(x) = x + 1$ và $g(x) = x + 2$ là giao hoán. Hai hàm $f(x) = x + 1$ và $g(x) = 2x$ không giao hoán. Hàm số $f: X \to X$ được gọi là song ánh với $\forall y \in X$, tồn tại một và chỉ một giá trị $x \in X$ sao cho $y = f(x)$.

Yêu cầu: Cho hàm số $f: N_n \to N_n$ song ánh, với $N_n = \{1,2,3, \dots, n\}$. Xác định dãy $b_1, b_2, \dots, b_n$ có thứ tự từ điển nhỏ nhất với $b_i = g(i)$ và g là hàm số giao hoán của f.

Dữ liệu: vào từ file **COMMUTE.INP**

- Dòng đầu chứa số nguyên dương n $(n \leq 2.10^5)$.
- Dòng thứ hai chứ n số nguyên dương biểu diễn giá trị $f(1), f(2), \dots, f(n)$.

Kết quả: Ghi ra file **COMMUTE.OUT** n số nguyên $b_1, b_2, b_3, \dots, b_n$

COMMUTE.INP	COMMUTE.OUT
10 2 3 4 5 6 7 8 1 9 10	1 2 3 4 5 6 7 8 9 9

Ràng buộc:

- 30% số test $n \leq 10$