

Biến đổi dãy số

Cô giáo dạy toán rất không ưa Steve và thường gọi Steve lên bảng giải những bài rất khó. Hôm nay cô giáo viết lên bảng dãy số nguyên không âm a_1, a_2, \dots, a_n và yêu cầu dùng ít phép biến đổi nhất để đưa về dãy số mới sao cho trong đó có h số liên tiếp nhau tạo thành một dãy tăng dần từ 1 đến h , tức là tồn tại i sao cho $a_i = 1, a_{i+1} = 2, \dots, a_{i+h-1} = h$. Nội dung mỗi phép biến đổi là xóa một số tùy chọn và thay nó bằng số mới lớn hơn số cũ một đơn vị.

Dĩ nhiên, anh bạn tội nghiệp Steve của chúng ta lại bị gọi lên bảng.

Hãy xác định số phép biến đổi tối thiểu Steve cần thực hiện hoặc đưa ra số -1 nếu không tồn tại cách nhận được dãy số mới theo yêu cầu.

Dữ liệu: Vào từ file văn bản SEQ.INP:

- ✚ Dòng đầu tiên chứa 2 số nguyên n và h ($1 \leq h \leq n \leq 200\,000$),
- ✚ Dòng thứ 2 chứa n số nguyên a_1, a_2, \dots, a_n ($0 \leq a_i \leq n, i = 1 \div n$).

Kết quả: Đưa ra file văn bản SEQ.OUT một số nguyên – kết quả tìm được.

Ví dụ:

SEQ.INP	SEQ.OUT
4 3 1 1 0 2	3

Giáo sư Chaos

Phòng thí nghiệm tuyệt mật của giáo sư Chaos nghiên cứu về một loại vi khuẩn cực độc. Đầu giờ ngày đầu tiên phòng thí nghiệm có a cá thể vi khuẩn nguy hiểm này.

Mỗi ngày thí nghiệm được tiến hành theo quy trình sau. Buổi sáng giáo sư lấy hộp chứa vi khuẩn, chuẩn tất cả chúng sang tủ ấm nhân giống. Ở đó chúng sẽ sinh sản, mỗi vi khuẩn sẽ tạo thành b vi khuẩn mới. Tiếp theo, người ta sẽ lấy ra khỏi tủ nhân giống c vi khuẩn để tiến hành các nghiên cứu khác nhau, sau đó khử trùng, diệt hết chúng. Nếu số lượng lấy ra không đủ c thì tất cả vi khuẩn sẽ được lấy ra thí nghiệm và nghiên cứu kết thúc. Số vi khuẩn còn lại cuối ngày sẽ được đưa vào hộp chứa để tiếp tục nghiên cứu trong những ngày tiếp theo. Nhưng hộp chứa chỉ có thể lưu giữ tối đa d vi khuẩn. Nếu còn thừa nhiều người ta cũng chỉ lưu lại d vi khuẩn, số vi khuẩn còn thừa, nếu có, sẽ bị diệt hết.

Để lên kế hoạch nghiên cứu giáo sư Chaos cần biết sau ngày thứ k sẽ có bao nhiêu vi khuẩn còn lại.

Hãy xác định số lượng vi khuẩn còn lại sau ngày thứ k .

Dữ liệu: Vào từ file văn bản CHAOS.INP gồm một dòng chứa 5 số nguyên a, b, c, d và k ($1 \leq a, b \leq 1000, 0 \leq c \leq 1000, a \leq d \leq 1000, 1 \leq k \leq 10^{18}$).

Kết quả: Đưa ra file văn bản CHAOS.OUT một số nguyên – số lượng vi khuẩn còn lại sau ngày thứ k.

Ví dụ:

CHAOS.INP
1 3 1 5 2

CHAOS.OUT
5

Các Cỗ Bài

Steve có nhiệm vụ lập trình minh họa cho một trò chơi mới với cỗ bài n quân. Đây là trò chơi một người. Trên mỗi quân bài có ghi một số nguyên trong phạm vi từ 1 đến m . Người chơi được chia k quân bài trên cùng của cỗ bài. Phần còn lại của cỗ bài được đặt trên bàn. Trong suốt quá trình chơi, không lúc nào người chơi không được cầm trên tay quá k quân bài. Ở mỗi thời điểm có thể thực hiện một trong 3 hành động sau:

- ✚ Bỏ một quân bài đang cầm trên tay xuống tập bài đập. Quân bài này sẽ không được dùng lại trong quá trình chơi tiếp theo,
- ✚ Nếu số bài cầm trên tay ít hơn k , người chơi có thể lấy một quân bài trên cùng ở cỗ bài trên bàn, bỏ sung vào số bài mình đang cầm,
- ✚ Hạ quân bài có số x xuống bàn nếu trước đó đã hạ các quân bài với các số $1, 2, \dots, x-1$ và quân bài số x chưa hạ.

Trò chơi kết thúc khi không thể thực hiện được hành động nào trong số kể trên. Mục tiêu của trò chơi là làm thế nào để hạ được nhiều quân bài nhất có thể.

Steve lập trình minh họa nên biết trước số ghi trên các quân bài ở cỗ bài trên bàn và biết các quân được chia cho người chơi. Hãy xác định số tối đa các quân bài có thể hạ xuống bàn.

Dữ liệu: Vào từ file văn bản CARDS.INP:

- ✚ Dòng đầu tiên chứa số nguyên t – số lượng tests trong file ($1 \leq t \leq 10^5$),
- ✚ Dữ liệu mỗi test ghi trên 2 dòng:
 - Dòng đầu tiên chứa 3 số nguyên n, m và k ($1 \leq n, m \leq 10^5, 1 \leq k \leq n$),
 - Dòng thứ 2 chứa n số nguyên a_1, a_2, \dots, a_n ($1 \leq a_i \leq m, i = 1 \div n$). Tổng các n trong file không vượt quá 10^5 .

Kết quả: Đưa ra file văn bản CARDS.OUT, kết quả mỗi test đưa ra trên một dòng dưới dạng số nguyên, xác định số quân bài tối đa có thể hạ.

Ví dụ:

CARDS.INP	CARDS.OUT
3	2
4 3 1	0
3 2 1 2	4
1 2 1	
2	
5 5 2	
4 2 1 4 3	