



# RRATING

Link submit: <https://www.codechef.com/problems/RRATING>

Solution:

C++	<a href="https://ideone.com/0NrMPV">https://ideone.com/0NrMPV</a>
Java	<a href="https://ideone.com/X2Svx5">https://ideone.com/X2Svx5</a>
Python	<a href="https://ideone.com/9LNRy7">https://ideone.com/9LNRy7</a>

Tóm tắt đề:

Một nhà hàng mới mở được các nhà phê bình đánh giá. Đầu bếp muốn gom nhiều đánh giá tích cực nhất có thể. Ông biết thực tế mọi người không xem qua hết tất cả đánh giá. Vì vậy, ông chọn ra các đánh giá tích cực và đăng nó trên trang web của nhà hàng. Một đánh giá được coi là tích cực nếu nó nằm trong top 1/3 tổng số đánh giá khi chúng được xếp theo xếp hạng. Ví dụ, giả sử xếp hạng gồm 8 đánh giá khác nhau như sau: 2 8 3 1 6 4 5 7. Thì những đánh giá trong top 1/3 là 8 và 7. Ta xem  $1/3$  là  $8/3 = 2$  đánh giá ở top. Để chính xác, số lượng đánh giá được hiển thị trên trang web sẽ bằng  $\text{floor}(n/3)$  với  $n$  là số lượng đánh giá hiện tại.

Cho các đánh giá của các nhà phê bình, đầu bếp muốn biết xếp hạng nhỏ nhất mà trang web sẽ hiển thị. Ở ví dụ trên, xếp hạng nhỏ nhất được hiển thị là 7. Các nhà phê bình khác nhau đánh giá nhà hàng liên tục. Vì vậy, những đánh giá mới được thêm vào và bảng xếp hạng được cập nhật liên tục. Tại thời điểm bất kỳ, đầu bếp muốn biết xếp hạng nhỏ nhất được hiển thị. Lưu ý là đánh giá có thể xuất hiện trên trang web trong một thời gian và bị loại vì có các đánh giá mới tốt hơn và ngược lại.

Input:

Dòng đầu tiên chứa một số nguyên  $N$  ( $1 \leq N \leq 250000$ ) – số lượng hành động.

$N$  dòng tiếp theo chứa một hành động trên mỗi dòng. Hành động gồm hai loại:

- 1 x: thêm một đánh giá với xếp hạng 'x' (x là số nguyên,  $1 \leq x \leq 1000000000$ ) vào danh sách đánh giá hiện có.
- 2: báo cáo xếp hạng nhỏ nhất hiện tại của trang web.

Output:

Mỗi test case, in một số nguyên cho mỗi hành động loại 2. Nếu không có đánh giá đủ điều kiện để xem là tích cực, in "No reviews yet".

### Ví dụ:

1 0	No reviews yet
1 1	9
1 7	9
2	
1 9	
1 21	
1 8	
1 5	
2	
1 9	
2	

### Giải thích ví dụ:

Trong ví dụ gồm 10 hành động. Hai hành động đầu thêm đánh giá 1 và 7 vào danh sách. Hành động thứ 3 loại 2, danh sách hiện tại có 2 đánh giá, top 1/3 là  $\text{floor}(2/3) = 0$ , in No reviews yet.

Hành động 4 – 7 thêm các đánh giá 9, 21, 8, 5 vào danh sách. Hành động 8 loại 2, danh sách hiện tại có 6 đánh giá, sắp xếp lại: 21 9 8 7 5 1, top 1/3 là  $\text{floor}(6/3) = 2$ , vậy xếp hạng nhỏ nhất hiện tại trong top 1/3 là 9.

Hành động 9 thêm đánh giá 9 vào. Hành động 10 loại 2, danh sách hiện tại có 7 đánh giá: 21 9 9 8 7 5 1, top 1/3 là  $\text{floor}(7/3) = 2$ , vậy xếp hạng nhỏ nhất hiện tại trong top 1/3 là 9.

### Hướng dẫn giải:

Ban đầu, ta thấy với mỗi type 2 ta chỉ cần đi sắp xếp lại mảng chứa danh sách đánh giá hiện tại giảm dần và in ra đánh giá thứ  $\text{floor}(n/3)$  (mảng bắt đầu từ 1). Cách này sẽ vượt quá thời gian quy định. Cứ mỗi lần sort mảng sẽ tốn nhiều thời gian, ta có thể thay bằng heap thì chỉ tốn thời gian khi thêm đánh giá mới, chứ không phải sort cả mảng. Nhưng dùng heap thì chỉ có thể lấy top, muốn biết đánh giá thứ  $\text{floor}(n/3)$  thì phải pop ra từng cái, và sau đó phải push vô heap lại.

Ta có một nhận xét khác, khi có một đánh giá tốt hơn so với top 1/3 thì cái mới thêm vào top, còn cái thấp nhất sẽ bị loại khỏi top và nằm trong phần còn lại của danh sách đánh giá. Đồng thời, top được tính là  $n/3$ , nên khi số đánh giá ít hơn 3 thì không có đánh giá nào trong top, khi bằng 3 4 5 có 1 đánh giá trong top, 6 7 8 có 2 đánh giá trong top, cứ tiếp tục như vậy. Có thể thấy khi số đánh giá tăng đến khi chia hết cho 3 thì số đánh giá trong top tăng lên 1. Vậy dùng min-heap để lưu đánh giá top 1/3 – top của min-heap là đánh giá nhỏ nhất hiện tại, max-heap lưu phần còn lại. Khi có một đánh giá mới x, nếu x lớn hơn top của min-heap thì đem top đó bỏ qua max-heap và đẩy x vào min-heap. Ngược lại, đẩy x vào max-heap. Sau đó, nếu số đánh giá hiện tại chia hết cho 3 thì lấy top của max-heap đem qua min-heap để tăng số trong top lên 1.

**Độ phức tạp:**  $O(N \log N)$  với N là số lượng thao tác.