



Minimum Loss

Link submit: <https://www.hackerrank.com/contests/womens-codesprint-2/challenges/minimum-loss>

Solution:

C++	http://ideone.com/Z0dA7V
Java	https://ideone.com/BAyQhh
Python	https://ideone.com/bk6Fln

Tóm tắt đề:

Có một ngôi nhà và giá của ngôi nhà được thay đổi qua từng năm. Cụ thể năm thứ i sẽ có giá là p_i . Bạn cần phải chọn ra một thời điểm x để mua ngôi nhà, rồi chọn một thời điểm y để bán lại ngôi nhà ($y > x$, $p_y < p_x$) sao cho hiệu $p_x - p_y$ là nhỏ nhất có thể.

Input:

Dòng đầu tiên gồm một số nguyên dương n là số lượng năm mà ngôi nhà có thể bán hoặc mua được ($1 \leq n \leq 2 \cdot 10^5$).

Dòng thứ hai gồm n số nguyên dương cách nhau bởi một dấu khoảng trắng, số nguyên dương thứ i thể hiện giá tiền của căn nhà ở năm thứ i ($1 \leq p_i \leq 10^{16}$).

Output:

Yêu cầu in ra hiệu nhỏ nhất có thể khi bạn mua rồi bán lại trong năm sau.

Ví dụ:

3 5 10 3	2
-------------	---

Giải thích ví dụ:

Bạn mua ngôi nhà vào năm thứ 1, bạn được 5 đồng. Sau đó vào năm thứ 3, bạn bán lại ngôi nhà với giá 3 đồng. Như vậy bạn chịu lỗ $5 - 3 = 2$ đồng.

Hướng dẫn giải:

Với mỗi năm, gọi p là giá tiền của ngôi nhà lớn hơn giá ở năm thứ i ($a[i]$) nhưng là giá nhỏ nhất trong các năm từ năm 1 đến năm $i - 1$. Nếu tìm được p thì cập nhật lại $res = \min(res, p - a[i])$.

Để có thể thực hiện được việc tìm p , bạn bỏ các phần tử từ phần tử 1 đến phần tử $i - 1$ vào một cây nhị phân tìm kiếm. Sau đó, với mỗi phần tử $a[i]$, sử dụng hàm tìm phần tử nhỏ nhất lớn hơn $a[i]$ để tìm ra được p .

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

Big-O Coding