

LESSON 2A

Tìm kiếm nhị phân hay thường gọi là *chặt nhị phân*, tên gọi theo Tiếng Anh là *Binary Search* là một thuật toán hiệu quả trong các bài toán tìm kiếm trên các hàm tuyến tính thay vì *tìm kiếm tuần tự*. Bài toán sau đây đặc tả bản chất của thuật toán tìm kiếm nhị phân: “**Cho dãy số nguyên A gồm N phần tử a_1, a_2, \dots, a_N có tính chất $a_1 < a_2 < \dots < a_N$. Có Q câu hỏi, mỗi câu hỏi là một giá trị x , với mỗi câu hỏi bạn cần in ra vị trí của giá trị x trên dãy A đã cho hoặc in ra 0 nếu như x không xuất hiện trên dãy A** ”. Ví dụ: Dãy A gồm 3 phần tử $a_1 = -4, a_2 = 3, a_3 = 7$ và một câu hỏi có giá trị $x = 3$ tức là ta sẽ in ra 2, nếu câu hỏi với giá trị $x = 5$ thì sẽ in ra 0. Để đưa ra vị trí của giá trị x , thông thường ta sẽ tìm kiếm tuần tự từ vị trí $1 \rightarrow N$ sẽ mất độ phức tạp tối đa là $O(N)$ trong khi vận dụng tính chất tăng dần của dãy, người ta chỉ cần thực hiện tối đa $O(\log_2(N))$ với mã giả như sau:

```
L = 1; R = N; Pos = 0;
while (L ≤ R){
    M = ⌊ $\frac{L+R}{2}$ ⌋;
    if (a[M] = x) → Pos = M và break.
    if (a[M] < x) → L = M + 1;
    Ngược lại → R = M - 1;
}
In ra kết quả là Pos.
```

Nhận thấy sau mỗi vòng lặp, đoạn $[L, R]$ sẽ giảm đi một nửa, như vậy có nghĩ là bắt đầu từ $[1, N]$ thì nó sẽ kết thúc sau tối đa $\log_2(N)$ lần, bạn hãy thực hiện bài toán trên theo ngôn ngữ C++ hoàn chỉnh thay vì mã giả.

Input:

- Dòng đầu tiên gồm duy nhất một số nguyên dương N ($N \leq 10^5$);
- Dòng thứ hai là một dãy số nguyên A tăng dần gồm N phần tử a_1, a_2, \dots, a_N ($|a_i| \leq 10^9$);
- Dòng thứ ba là một số nguyên dương Q ($Q \leq 10^5$) – số lượng câu hỏi;
- Q dòng tiếp theo, mỗi dòng là một số nguyên x ($|x| \leq 10^9$).

Output: Gồm Q dòng, mỗi dòng là vị trí số nguyên x tương ứng trên dãy A hoặc in ra 0 nếu x không nằm trên dãy.

Ví dụ:

| IBS1304A.INP | IBS1304A.OUT |
|--------------|--------------|
| 5 | 1 |
| -4 -1 7 9 12 | 0 |
| 3 | 4 |
| -4 | |
| 2 | |
| 9 | |