

Tìm kiếm nhị phân

Bài toán:

Thuật toán tìm kiếm nhị phân là một thuật toán tìm kiếm đơn giản nhưng có tốc độ tìm kiếm cực kỳ nhanh. Ý tưởng của thuật toán là: với một mảng N phần tử đã được sắp xếp (tăng dần/hoặc giảm dần tùy theo người ra đề) và số nguyên x cần tìm, ta sẽ thực hiện tìm kiếm ở miền có khả năng xuất hiện x sau mỗi lần lặp.

Yêu cầu: Hãy cài đặt thuật toán tìm kiếm nhị phân để tìm vị trí của phần tử x trong mảng có N phần tử và đếm số lần duyệt qua các phần tử.

Input :

- Dòng đầu tiên là số nguyên N dương ($0 < N < 32000$)
- Dòng tiếp theo chứa N số nguyên A_i là các phần tử của mảng đã được sắp xếp tăng dần và không trùng nhau
- Dòng cuối cùng là số nguyên x cần tìm

Output:

- Dòng đầu tiên là vị trí của x được tìm thấy trong mảng. Nếu không tìm thấy thì xuất ra -1
- Dòng tiếp theo là số lần duyệt qua các phần tử để tìm được x . Nếu không tìm thấy thì không cần xuất ra dòng này

Ví dụ:

INPUT	OUTPUT	Giải thích
8 1 3 4 5 10 12 15 20 4	2 3	- 4 được tìm thấy ở vị trí 2 - để tìm được 3 cần phải duyệt qua các phần tử ở vị trí 3, 1 và 2.
4 1 2 3 4 -9	-1	- không tìm thấy -9 trong mảng

Lưu ý: Số lần duyệt để tìm x trong bài sẽ dựa trên pseudo code sau đây:

```
function binary_search(A, n, T) is  
  L := 0  
  R := n - 1  
  while L ≤ R do  
    m := floor((L + R) / 2)  
    if A[m] < T then  
      L := m + 1  
    else if A[m] > T then  
      R := m - 1  
    else:  
      return m  
  return unsuccessful
```