

## Giải thuật tìm kiếm nhị phân (Binary Search)

⌂ Trang trước (../cau-truc-du-lieu-va-giai-thuat/giai-thuat-tim-kiem-tuyen-tinh.jsp)

Trang sau ➔ (../cau-truc-du-lieu-va-giai-thuat/giai-thuat-tim-kiem-noi-suy.jsp)

### Giải thuật tìm kiếm nhị phân (Binary Search) là gì ?

Binary Search (Tìm kiếm nhị phân) là một giải thuật tìm kiếm nhanh với độ phức tạp thời gian chạy là  $O(\log n)$ . Giải thuật tìm kiếm nhị phân làm việc dựa trên nguyên tắc chia để trị (Divide and Conquer). Để giải thuật này có thể làm việc một cách chính xác thì tập dữ liệu nên ở trong dạng đã được sắp xếp.

Binary Search tìm kiếm một phần tử cụ thể bằng cách so sánh phần tử tại vị trí giữa nhất của tập dữ liệu. Nếu tìm thấy kết nối thì chỉ mục của phần tử được trả về. Nếu phần tử cần tìm là lớn hơn giá trị phần tử giữa thì phần tử cần tìm được tìm trong mảng con nằm ở bên phải phần tử giữa; nếu không thì sẽ tìm ở trong mảng con nằm ở bên trái phần tử giữa. Tiến trình sẽ tiếp tục như vậy trên mảng con cho tới khi tìm hết mọi phần tử trên mảng con này.

### Cách Binary Search làm việc

Để Binary Search làm việc thì mảng phải cần được sắp xếp. Để tiện cho việc theo dõi, mình sẽ cung cấp thêm các hình minh họa tương ứng với mỗi bước.

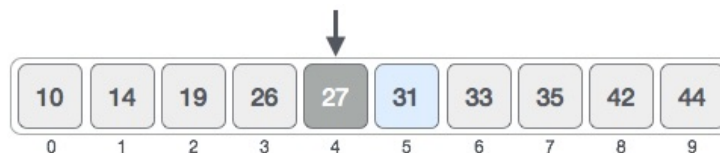
Giả sử chúng ta cần tìm vị trí của giá trị 31 trong một mảng bao gồm các giá trị như hình dưới đây bởi sử dụng Binary Search:



Đầu tiên, chúng ta chia mảng thành hai nửa theo phép toán sau:

$$\text{chỉ-mục-giữa} = \text{ban-đầu} + (\text{cuối} + \text{ban-đầu}) / 2$$

Với ví dụ trên là  $0 + (9 - 0) / 2 = 4$  (giá trị là 4.5). Do đó 4 là chỉ mục giữa của mảng.

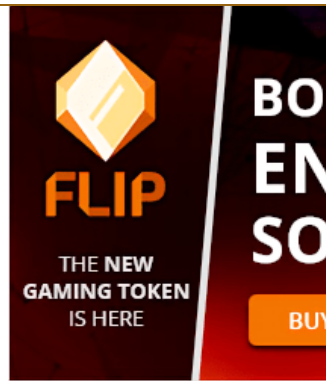


Bây giờ chúng ta so sánh giá trị phần tử giữa với phần tử cần tìm. Giá trị phần tử giữa là 27 và phần tử cần tìm là 31, do đó là không kết nối. Bởi vì giá trị cần tìm là lớn hơn nên phần tử cần tìm sẽ nằm ở mảng con bên phải phần tử giữa.



Chúng ta thay đổi giá trị ban-đầu thành chỉ-mục-giữa + 1 và lại tiếp tục tìm kiếm giá trị chỉ-mục-giữa.

$$\begin{aligned} \text{ban-đầu} &= \text{chỉ-mục-giữa} + 1 \\ \text{chỉ-mục-giữa} &= \text{ban-đầu} + (\text{cuối} + \text{ban-đầu}) / 2 \end{aligned}$$



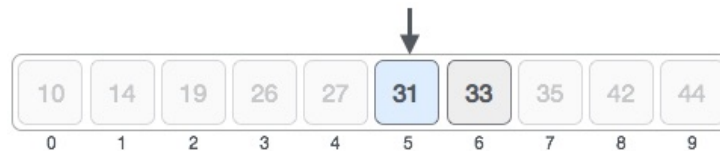
Bây giờ chỉ mục giữa của chúng ta là 7. Chúng ta so sánh giá trị tại chỉ mục này với giá trị cần tìm.



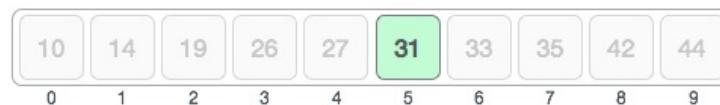
Giá trị tại chỉ mục 7 là không kết nối, và ngoài ra giá trị cần tìm là nhỏ hơn giá trị tại chỉ mục 7 do đó chúng ta cần tìm trong mảng con bên trái của chỉ mục giữa này.



Tiếp tục tìm chỉ-mục-giữa lần nữa. Lần này nó có giá trị là 5.



So sánh giá trị tại chỉ mục 5 với giá trị cần tìm và thấy rằng nó kết nối.



Do đó chúng ta kết luận rằng giá trị cần tìm 31 được lưu giữ tại vị trí chỉ mục 5.

Binary Search chia đôi lượng phần tử cần tìm và do đó giảm số lượng phép so sánh cần thực hiện nên giải thuật tìm kiếm này được thực hiện khá nhanh.

## Giải thuật mẫu cho Binary Search

Dưới đây là code mẫu cho giải thuật tìm kiếm nhị phân:

Giải thuật tìm kiếm nhị phân (Binary Search)

A ← một mảng đã được sắp xếp

n ← kích cỡ mảng

x ← giá trị để tìm kiếm trong mảng

gán lowerBound = 1

gán upperBound = n

while x not found

if upperBound < lowerBound

EXIT: x không tồn tại.

gán midPoint = lowerBound + ( upperBound - lowerBound ) / 2

if A[midPoint] < x

gán lowerBound = midPoint + 1

if A[midPoint] > x

gán upperBound = midPoint - 1

if A[midPoint] = x

EXIT: x được tìm thấy tại midPoint

kết thúc while

kết thúc giải thuật

Để tìm hiểu sự triển khai code đầy đủ của giải thuật tìm kiếm nhị phân trong ngôn ngữ C, mời bạn click chuột vào chương: **Tìm kiếm nhị phân (Binary Search) trong C** ([../cau-truc-du-lieu-va-giai-thuat/tim-kiem-nhi-phan-trong-c.jsp](#)) .

Loạt bài hướng dẫn **Cấu trúc dữ liệu và giải thuật** của chúng tôi dựa trên nguồn tài liệu của trang: Tutorialspoint

Follow fanpage của team <https://www.facebook.com/vietjackteam/> (<https://www.facebook.com/vietjackteam/>) hoặc facebook cá nhân Nguyễn Thanh Tuyền <https://www.facebook.com/tuyen.vietjack> (<https://www.facebook.com/tuyen.vietjack>) để tiếp tục theo dõi các loạt bài mới nhất về Java,C,C++,Javascript,HTML,Python,Database,Mobile.... mới nhất của chúng tôi.

**Bài học Cấu trúc dữ liệu và giải thuật phổ biến tại vietjack.com:**

Giải thuật tiệm cận - Asymptotic Algorithms (<http://vietjack.com/cau-truc-du-lieu-va-giai-thuat/phan-tich-tiem-can.jsp>)

Cấu trúc dữ liệu mảng (Array) (<http://vietjack.com/cau-truc-du-lieu-va-giai-thuat/cau-truc-du-lieu-mang.jsp>)

Danh sách liên kết - Linked List (<http://vietjack.com/cau-truc-du-lieu-va-giai-thuat/cau-truc-du-lieu-danh-sach-lien-ket.jsp>)

Cấu trúc dữ liệu ngăn xếp - Stack (<http://vietjack.com/cau-truc-du-lieu-va-giai-thuat/cau-truc-du-lieu-ngan-xep.jsp>)

Cấu trúc dữ liệu hàng đợi - Queue (<http://vietjack.com/cau-truc-du-lieu-va-giai-thuat/cau-truc-du-lieu-hang-doi.jsp>)

Tìm kiếm tuyến tính - Linear Search (<http://vietjack.com/cau-truc-du-lieu-va-giai-thuat/giai-thuat-tim-kiem-tuyen-tinh.jsp>)

Tìm kiếm nhị phân - Binary Search (<http://vietjack.com/cau-truc-du-lieu-va-giai-thuat/giai-thuat-tim-kiem-nhi-phan.jsp>)

Sắp xếp nổi bọt - Bubble Sort (<http://vietjack.com/cau-truc-du-lieu-va-giai-thuat/giai-thuat-sap-xep-noi-bot.jsp>)

Sắp xếp chèn - Insertion Sort (<http://vietjack.com/cau-truc-du-lieu-va-giai-thuat/giai-thuat-sap-xep-chen.jsp>)

⌂ Trang trước ([../cau-truc-du-lieu-va-giai-thuat/giai-thuat-tim-kiem-tuyen-tinh.jsp](#))

Trang sau ➡ ([../cau-truc-du-lieu-va-giai-thuat/giai-thuat-tim-kiem-noi-suy.jsp](#))

1 bình luận

Sắp xếp theo

Hàng đầu



Thêm bình luận...

Plugin bình luận của Facebook

**Các loạt bài khác:**

160 bài học ngữ pháp tiếng Anh hay nhất (<http://vietjack.com/ngu-phap-tieng-anh/index.jsp>)

155 bài học Java tiếng Việt hay nhất (<http://vietjack.com/java/index.jsp>)

100 bài học Android tiếng Việt hay nhất (<http://vietjack.com/android/index.jsp>)

247 bài học CSS tiếng Việt hay nhất (<http://vietjack.com/css/index.jsp>)

197 thẻ HTML cơ bản (<http://vietjack.com/html/index.jsp>)

297 bài học PHP (<http://vietjack.com/php/index.jsp>)

85 bài học C# hay nhất (<http://vietjack.com/csharp/index.jsp>)

101 bài học C++ hay nhất (<http://vietjack.com/cplusplus/index.jsp>)

97 bài tập C++ có giải hay nhất ([http://vietjack.com/bai\\_tap\\_cplusplus\\_co\\_giai/index.jsp](http://vietjack.com/bai_tap_cplusplus_co_giai/index.jsp))

208 bài học Javascript có giải hay nhất (<http://vietjack.com/javascript/index.jsp>)

Và còn rất nhiều loạt bài học khác....

**Học tiếng Anh tại vietjack.com:**

160 bài học ngữ pháp tiếng Anh hay nhất (<http://vietjack.com/ngu-phap-tieng-anh/index.jsp>)

160 bài tập ngữ pháp tiếng Anh hay nhất (<http://vietjack.com/bai-tap-ngu-phap-tieng-anh/index.jsp>)

72 bài ngữ pháp thực hành (<http://vietjack.com/ngu-phap-tieng-anh-co-ban/index.jsp>)

50 tình huống tiếng Anh thông dụng (<http://vietjack.com/tinh-huong-tieng-anh-thong-dung/index.jsp>)

120 bí kíp luyện phần V TOEIC (<http://vietjack.com/part-5-toeic/index.jsp>)



**VietJack.com** - Chia sẻ...  
95.744 lượt thích

Đã thích Đăng kí ngay Chia sẻ

Bạn và 12 người bạn khác thích nội dung này




Boosts your productivity  
with PhpStorm —  
the most powerful PHP IDE

**PhpStorm**

Try FREE now

**J .com** (index.jsp)

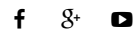
Trang web chia sẻ nội dung miễn phí dành cho người Việt.

Lớp (../series/lop-3.jsp)	3	Lớp (../series/lop-4.jsp)	4	Lớp (../series/lop-5.jsp)	5	Lớp (../series/lop-6.jsp)	6	Lớp (../series/lop-7.jsp)	7	Lớp (../series/lop-8.jsp)	8
Lớp (../series/lop-9.jsp)	9	Lớp (../series/lop-10.jsp)	10	Lớp (../series/lop-11.jsp)	11	Lớp (../series/lop-12.jsp)	12	IT - Lập trình (../series/it-lap-trinh.jsp)		Tiếng Anh (../series/mon-tieng-anh.jsp)	

Liên hệ với chúng tôi

📍 66 Kim Hoa, Đống Đa, Hà Nội  
 📞 Phone: 01689933602  
 ✉ Email: [vietjackteam@gmail.com](mailto:vietjackteam@gmail.com)  
 (mailto:vietjackteam@gmail.com)

Tuyển dụng (<http://vietjack.com/lien-he/tuyen-dung.jsp>)



Về chúng tôi (<http://vietjack.com/lien-he/>)