

# Bài 10.6: Một số hàm tìm kiếm trong thư viện C++

---

- ✓ Tổng quan
- ✓ Một số hàm thông dụng
- ✓ Ví dụ minh họa
- ✓ Bài tập thực hành

# Tổng quan

- Thư viện `<algorithm>` của C++ đã hỗ trợ rất nhiều các hàm xây dựng sẵn trong đó có cả các hàm hỗ trợ tìm kiếm nhanh.
- Để sử dụng các hàm đó bạn include thư viện `<algorithm>` vào đầu file chương trình.
- Tất cả các hàm trong thư viện này đều là hàm template. Do đó nó tương thích với mọi kiểu dữ liệu hợp lệ bạn muốn sử dụng.

# Một số hàm thông dụng

Tên hàm	Mô tả
<code>bool binary_search(ForwardIt first, ForwardIt last, const T&amp; value);</code> (Tới C++ 20)	Kiểm tra xem phần tử tương đương value có xuất hiện trong phạm vi [first, last) hay không. Sử dụng toán tử < để so sánh
<code>constexpr bool binary_search(ForwardIt first, ForwardIt last, const T&amp; value);</code> (C++ 20)	
<code>bool binary_search(ForwardIt first, ForwardIt last, const T&amp; value, Comp comp);</code> (Tới C++ 20)	Kiểm tra xem phần tử tương đương value có xuất hiện trong phạm vi [first, last) hay không. Sử dụng hàm so sánh comp cho trước để so sánh
<code>constexpr bool binary_search(ForwardIt first, ForwardIt last, const T&amp; value, Comp comp);</code> (C++ 20)	
<code>constexpr bool binary_search(l first, S last, const T&amp; value, Comp comp = {}, Proj proj = {});</code> (C++ 20)	Kiểm tra xem phần tử dự kiến tương đương với value có xuất hiện trong khoảng [first, last) hay không.
<code>constexpr bool binary_search(R&amp;&amp; r, const T&amp; value, Comp comp = {}, Proj proj = {});</code> (C++ 20)	Tương tự như trên nhưng sử dụng khoảng phần tử cho trong tham số r. Tức là nó sử dụng ranges::begin(r) làm first và ranges::end(r) làm last.

# Nội dung tiếp theo

**Các thuật toán trên đồ thị**