

Phân tích các cấu trúc dữ liệu phổ biến trong C#

Việc sử dụng cấu trúc dữ liệu với đúng công việc cũng là một kỹ năng quan trọng trong lập trình. Việc lựa chọn sai có thể khiến cho ứng dụng của bạn chậm đi đáng kể.

Tóm tắt

Kiểu dữ liệu	Truy cập (index/key)	Tìm kiếm	Thêm (cuối)	Thêm (giữa)	Xóa (giữa)
Array	O(1)	O(n)	O(1) (chưa đầy) O(n) (đầy)	O(n)	O(n)
List<T>	O(1)	O(n)	O(1)	O(n)	O(n)
Dictionary<TKey, TValue>	O(1)	O(1) (theo key) O(n) (theo value)	O(1)	x	O(1)
HashSet<T>	x	O(1)	O(1)	x	O(1)
Queue<T>	x	O(n)	O(1)	x	O(1)
Stack<T>	x	O(n)	O(1)	x	O(1)

Lưu ý khi sử dụng

- Sử dụng Array, List khi cần truy cập nhanh theo chỉ số.
 - Array nhẹ hơn do không có chi phí quản lý
 - List nhiều tiện ích (thêm, sửa, xóa, kiểm tra...)
- Sử dụng Dictionary khi cần tìm kiếm nhanh theo khoá.
- Sử dụng Queue, Stack khi cần thêm/xóa ở đầu hoặc cuối.
- Sử dụng HashSet cho danh sách không trùng lặp, không cần truy cập từng phần tử riêng lẻ.