Bài 13

|  |  |
| --- | --- |
| 1. Các loại thuật toán tìm kiếm ? | * Một số thuật toán tìm kiếm cơ bản:   + Tìm kiếm tuyến tính (Linear Search)  + Tìm kiếm nhị phân (Binary Search)  + Tìm kiếm nội suy (Interpolation Search) — ít dùng, khi dữ liệu phân bố đều  +Tìm kiếm băm (Hash Search) — sử dụng bảng băm  +Tìm kiếm theo cấu trúc dữ liệu (như tìm kiếm trên cây tìm kiếm nhị phân - BST). |
| 1. Ý tưởng thuật toán tìm kiếm tuyến tính? Độ phức tạp? | * Ý tưởng tìm kiếm tuyến tính   + Duyệt từng phần tử trong mảng từ trái sang  phải.  + So sánh từng phần tử với giá trị cần tìm.  + Nếu khớp → trả về vị trí; nếu không → trả  về -1.   * Độ phức tạp   + Tốt nhất: O(1) (nếu phần tử ở đầu)  + Trung bình & Xấu nhất: O(n) (phải duyệt hết mảng ) |
| 1. Ý tưởng thuật toán tìm kiếm nhị phân? Độ phức tạp? | * Ý tưởng tìm kiếm nhị phân   + Áp dụng cho mảng đã sắp xếp.  + So sánh phần tử giữa (mid) với giá trị cần tìm (x):  + Nếu bằng → trả về vị trí.  + Nếu x < mid → tìm trong nửa trái.  + Nếu x > mid → tìm trong nửa phải.  + Lặp lại cho đến khi tìm thấy hoặc không còn phần tử.   * Độ phức tạp   + Tốt nhất: O(1) (phần tử ở giữa ngay từ đầu)  + Trung bình & Xấu nhất: O(log n) (do chia đôi  liên tục) |
| 1. So sánh tìm kiếm tuyến tính và nhị phân | * So sánh tìm kiếm tuyến tính và tìm kiếm nhị  phân * Điều kiện sử dụng:   + Tuyến tính: Áp dụng cho mọi mảng, không cần sắp xếp.  + Nhị phân: Chỉ áp dụng cho mảng đã được sắp xếp.   * Cách hoạt động:   + Tuyến tính: Duyệt từng phần tử từ đầu đến  cuối, so sánh từng phần tử.  + Nhị phân: Chia đôi mảng, so sánh phần tử giữa, thu hẹp phạm vi tìm kiếm còn lại.   * Độ phức tạp:   + Tuyến tính:  + Tốt nhất: O(1)  + Trung bình & xấu nhất: O(n)   * Nhị phân:   + Tốt nhất: O(1)  + Trung bình & xấu nhất: O(log n)   * Ưu điểm:   + Tuyến tính: Dễ cài đặt, không cần sắp xếp mảng.  + Nhị phân: Nhanh và hiệu quả trên mảng lớn đã sắp xếp.   * Nhược điểm:   + Tuyến tính: Chậm khi mảng lớn.  + Nhị phân: Phải sắp xếp mảng trước khi tìm. |