

SẮP XẾP NHANH



Mục tiêu

- Nắm rõ tư tưởng của các thuật toán sắp xếp nhanh
- Cài đặt được các thuật toán sắp xếp nhanh trên ngôn ngữ lập trình C++
- Ứng dụng của các thuật toán sắp xếp nhanh vào các bài toán thực tế

Nội dung

- Sắp xếp Quick Sort
- Sắp xếp Heap Sort
- Sắp xếp Merge Sort

Sắp xếp Quicksort

Cho một mảng gồm n phần tử (ví dụ là các số nguyên):

- Nếu mảng chỉ có 1 phần tử thì dừng
- Ngược lại:
 - Lấy một phần tử làm khoá.
 - Phân hoạch các phần tử trong mảng vào trong 2 mảng con:
 - Các phần tử nhỏ hơn hoặc bằng khoá
 - Các phần tử lớn hơn khoá
 - Sắp xếp trên 2 mảng con
 - Trả về kết quả

Sắp xếp Quicksort

```
QuickSort(A, l,r){  
    If ( $l < r$ ) {  
        Pos:= Split(l,r);  
        QuickSort(l,pos-1);  
        QuickSort(pos+1,r);  
    }
```

Sắp xếp Quicksort

```
Split(A, low, high) {  
    Item:=A[low];  Left:=Low; Right:=high;  
    While (left <right) {  
        While A[right]> item {  
            Right:=right -1;  
            While (left <right) and A[left]<=item){  
                Left:=left +1;  
                If (left <right) {  
                    Permute (X[left],x[right])  
                    Pos:=Right;  
                }  
            }  
        }  
    }  
}
```

Sắp xếp Heapsort

- Sử dụng cây thư tự bộ phận
- Sử dụng maxheap để sắp xếp tăng dần

Sắp xếp MergeSort

- Xem dãy cần sắp xếp bao gồm hai dãy con đã được sắp, ta xây dựng dãy phần tử từ hai dãy trên sao cho dãy này cũng được sắp
- Trộn nhị phân
- Trộn tự nhiên

Tổng kết

- Các phương có thời gian thực hiện trung bình $< O(n^2)$ phù hợp với dữ liệu lớn.

Tiếp theo ...

