

BÀI TẬP CHỦ ĐỀ 4: SẮP XẾP

Phần trắc nghiệm (chọn 1 câu trả lời đúng nhất)

Câu 1. Ưu điểm của selection sort so với những thuật toán sort khác là gì?

- A. Không cần thêm vùng nhớ phụ
- B. Thuật toán chạy nhanh hơn các thuật toán sắp xếp khác
- C. Với mảng sắp thứ tự sẵn, selection sort chạy tốt nhất
- D. Tất cả các câu trên

Câu 2. Thuật toán sắp xếp nào sau đây có thời gian tìm kiếm xấu nhất ít hơn $O(n^2)$?

- A. Insertion sort
- B. Heap sort
- C. Quick sort
- D. Bubble sort

Câu 3. Cho mảng cần sắp xếp có n phần tử. Chi phí để sắp xếp mảng này với thuật toán Bubble sort trong trường hợp xấu nhất là bao nhiêu?

- A. $O(1)$
- B. $O(\log_2 n)$
- C. $O(n)$
- D. $O(n^2)$

Câu 4: Cần sắp xếp một mảng L được sắp theo thứ tự tăng dần nhưng có 1 số ít phần tử đứng không đúng chỗ (ví dụ $L = \{ 1, 3, 5, 9, 7, 10, 19, 50, 45, 59\}$), thuật toán sắp xếp nào sẽ thích hợp nhất để sắp xếp mảng L ?

- A. Heapsort
- B. Selection sort
- C. Quick sort
- D. Insertion sort

Câu 5: Thuật toán sắp xếp có thời gian chạy trung bình nhanh nhất là:

- A. Quick sort
- B. Heap sort
- C. Insertion sort
- D. Selection sort

Câu 6: Gọi P là thuật toán quick sort sắp xếp mảng tăng dần với phần tử lĩnh canh (pivot) là phần tử đầu tiên của mảng cần sắp thứ tự. Gọi t_1, t_2 là số phép so sánh thực hiện bởi P trên 2 mảng $A1 = \{1, 2, 3, 4, 5\}$ và $A2 = \{4, 1, 5, 3, 2\}$. Phát biểu nào sau đây đúng?

- A. $t_1 = 5$
- B. $t_1 < t_2$
- C. $t_1 > t_2$
- D. $t_1 = t_2$

Câu 7: Thuật toán sắp xếp nào có thời gian chạy tốt nhất và thời gian chạy xấu nhất bằng nhau?

- A. Heap sort
- B. Insertion sort
- C. Quick sort
- D. Selection sort

Câu 8: Thuật toán Quick sort cài đặt theo thiết kế kỹ thuật nào sau đây?

- A. Tham lam
- B. Quy hoạch động
- C. Chia để trị
- D. Quay lui

Câu 9: Trường hợp xấu nhất, thuật toán Quick sort có thời gian chạy là $O(n^2)$. Đó là trường hợp nào?

- A. Mảng sắp tăng dần
- B. Mảng sắp giảm dần
- C. Phần tử lĩnh canh là phần tử trung bình của mảng
- D. Phần tử lĩnh canh là phần tử lớn nhất hoặc nhỏ nhất trong mảng

Câu 10: Các thuật toán nào sau đây có thời gian chạy xấu nhất là $O(n^2)$?

- A. Insertion sort
- B. Bubble sort
- C. Selection sort
- D. Shaker sort
- E. Tất cả các thuật toán trên

Phần tự luận

Câu 1. Chứng minh rằng Heap Sort luôn có độ phức tạp là $O(n \log_2 n)$

Câu 2. Hãy chỉnh sửa thuật toán Quick Sort để sắp xếp dãy số nguyên theo thứ tự giảm dần

Câu 3. Thời gian chạy của thuật toán Quick Sort là bao nhiêu nếu tất cả các phần tử của mảng cần sắp xếp có giá trị bằng nhau.

Câu 4. Xét bản cài đặt thao tác phân hoạch (Partition) trong thuật toán Quick Sort sau đây:

```
i = 0;
j = n-1;
x = a[n/2];
do
{
    while (a[i] < x) i++;
    while (a[j] > x) j--;
    Hoanvi(a[i], a[j]);
}while (i <= j);
```

Có dãy $a[0], a[1], \dots, a[n-1]$ nào làm đoạn chương trình trên sai hay không? Cho ví dụ minh họa.

Câu 4. Hãy cài đặt các thuật toán: Insertion sort, Heapsort, Quicksort và báo cáo kết quả thời gian chạy của thuật toán cho các thí nghiệm sau:

- Thí nghiệm 1: Sắp xếp mảng tăng dần với mảng có 1,000 phần tử khởi tạo ngẫu nhiên.
- Thí nghiệm 2: Sắp xếp mảng tăng dần với mảng có 10,000 phần tử khởi tạo ngẫu nhiên.
- Thí nghiệm 3: Sắp xếp mảng tăng dần với mảng có 100,000 phần tử khởi tạo ngẫu nhiên.
- Thí nghiệm 4: Sắp xếp mảng tăng dần với mảng có 1,000,000 phần tử khởi tạo ngẫu nhiên.

Thí nghiệm/Thuật toán	Insertion sort	Heapsort	Quicksort
TN1: $n=1,000$	Ví dụ: 5ms
TN2: $n=10,000$	Ví dụ: 10ms
TN3: $n=100,000$
TN4: $n=1,000,000$

Vẽ đồ thị so sánh thời gian chạy của 3 thuật toán với n tăng dần theo bảng thống kê trên. Bạn có nhận xét gì khi quan sát bảng thống kê và đồ thị vừa vẽ?

Câu 5*. Chứng minh rằng $O(n \log_2 n)$ là ngưỡng chặn dưới của các thuật toán sắp xếp có sử dụng phép so sánh ($<$, $>$, $=$) trong cài đặt.