***CÂU HỎI CHƯƠNG 3***

***Câu 1:***

Quick Sort là phương pháp tối ưu nhất:

* Độ phức tạp tốt nhất: O(nlogn).
* Độ phức tạp xấu nhất: O(n2).

Bubble Sort là phương pháp kém tối ưu nhất:

* Độ phức tạp tốt nhất: n(n – 1) / 2.
* Độ phức tạp xấu nhất: n(n – 1) / 2.

***Câu 2:***

Trong các phương pháp, Selection Sort và Bubble Sort đều như nhau vì có độ phức tạp tốt nhất và xấu nhất là n(n – 1) / 2.

***Câu 3***

*Merge Sort* là một trong những thuật toán có độ phức tạp ở mức trung bình và cùng sử dùng phương pháp chia để trị giống thuật toán sắp xếp nhanh Quick Sort. Thuật toán này không chỉ áp dụng trong sắp xếp mà còn ở nhiều bài toán khác.

Giống như Quick Sort, Merge Sort là một thuật toán chia để trị. Thuật toán này chia mảng cần sắp xếp thành 2 nửa. Tiếp tục lặp lại việc này ở các nửa mảng đã chia. Sau cùng gộp các nửa đó thành mảng đã sắp xếp. Hàm merge() được sử dụng để gộp hai nửa mảng. Hàm merge(arr, l, m, r) là tiến trình quan trọng nhất sẽ gộp hai nửa mảng thành 1 mảng sắp xếp, các nửa mảng là arr[l…m] và arr[m+1…r] sau khi gộp sẽ thành một mảng duy nhất đã sắp xếp.

Độ phức tạp thuật toán:

* Trường hợp tốt: O(nlogn).
* Trung bình: O(nlogn).
* Trường hợp xấu: O(nlogn).
* Không gian bộ nhớ sử dụng: O(n).

*Bucket sort*

Đặt các phần tử của mảng input vào các xô thích hợp, sau khi đặt hết tất cả các phần tử vào trong các xô thì trong mỗi xô chúng ta sắp xếp các phần tử trong xô theo thứ tự. Cuối cùng là liên kết các xô lại trở thành dãy các phần tử đã được sắp xếp theo thứ tự.

Độ phức tạp của thuật toán là O(n log n).