

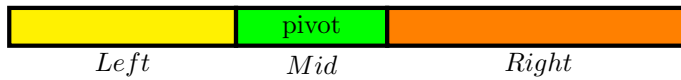
Võ Trần Đông Dương

PHẦN TRẢ LỜI CÂU HỎI CODING 3.4.2

Để tìm trung vị của mảng a gồm n phần tử mà không qua sắp xếp trực tiếp, ta có thể sử dụng thuật toán *quickselect*.

Ta sẽ quy bài toán này thành bài toán *Tìm phần tử lớn thứ k của mảng*. Hàm tìm phần tử thứ k trong mảng a sẽ là $find(a, k)$. Trong hàm $find$, ta thực hiện các bước sau:

1. Lựa chọn ngẫu nhiên một phần tử trong mảng đang xét làm *chốt* (*pivot*).
2. Chia mảng đang xét thành ba phần là *Left*, *Mid* và *Right*, với *Left* để lưu các phần tử có giá trị bé hơn *pivot*, *Right* để lưu các phần tử có giá trị lớn hơn *pivot*, còn *Mid* lưu các phần tử bằng với *pivot*.



3. Nếu $k < length(Left)$ thì phần tử thứ k sẽ nằm trong *Left*, ta sẽ tìm tiếp phần tử thứ k trong *Left* với hàm $find(Left, k)$.

Nếu $k < length(Left) + length(Mid)$ thì phần tử thứ k sẽ nằm trong *Mid*, may mắn thay, các phần tử trong *Mid* cũng chính là phần tử cần tìm, nên ta sẽ trả về $Mid[0]$.

Nếu k không rơi vào 2 trường hợp trên thì phần tử thứ k sẽ nằm trong *Right*, phần tử thứ k trong mảng a sẽ là phần tử thứ $k - length(Left) - length(Mid)$ trong *Right*, ta sẽ tìm tiếp với hàm

$$find(Right, k - length(Left) - length(Mid)).$$

Duy trì thực hiện đến khi đoạn mảng đang xét chỉ còn 1 phần tử, đó chính là phần tử cần tìm.

Từ đây, việc tìm trung vị của mảng a sẽ dựa vào thuật toán này.

Nếu n lẻ thì trung vị sẽ là $find(a, (n + 1)/2)$.

Nếu n chẵn thì trung vị sẽ là $\frac{1}{2} \cdot (find(a, n/2) + find(a, n/2 + 1))$.

Độ phức tạp của thuật toán trên là khoảng $O(n)$. Bởi vì mỗi lần đệ quy tìm kiếm, ta chỉ tìm trên một nửa của mảng đang xét, độ phức tạp trung bình sẽ là:

$$n + \frac{n}{2} + \frac{n}{4} + \frac{n}{8} + \dots = 2n \Rightarrow O(n).$$