Nhập môn phân tích độ phức tạp thuật toán – 21TN.

BÀI TẬP LÝ THUYẾT LẦN 1

Câu 1. (*3 điểm*)

1) Tại sao có thể nói rằng hàm $f(n) = n^{\sqrt{n}}$ thì có độ lớn nằm giữa "lớp các hàm đa thức n^a (với $a > 0$)" và "lớp các hàm mũ c^n (với $c > 1$)"?

2) Đánh giá độ lớn của hàm $f(n) = \dfrac{\lfloor n^4 \log n - 2n^2 + 5 \rfloor}{\lceil n\sqrt{n^3+1} + 10 \rceil}$ dưới dạng $\Theta(n^c)$.

3) Xét các hàm số $f(x), g(x), h(x)$ xác định và nhận giá trị trên tập số dương, xét tính đúng/sai của mệnh đề sau: nếu $f(x)g(x) \in \Theta(h(x)^2)$ và $f(x) \in \Omega(h(x))$ thì có $g(x) \in O(h(x))$.

Câu 2. (*3 điểm*)

1) So sánh phí tính toán của hai đoạn code bên dưới:

```
//algorithm 1
for(int i = 1; i*i <= n; i++){
    for(int j = 1; j*j*j <= n - i*i; j++){
        res += j;
    }
}
//algorithm 2
for(int i = 1; i*i*i <= n; i++){
    for(int j = 1; j*j <= n - i*i*i; j++){
        res += j;
    }
}
```

2) Với n nguyên dương, đặt $T_n = \left(\frac{1}{\sqrt{1}} + \frac{1}{\sqrt{2}} + \cdots + \frac{1}{\sqrt{n}}\right)(\log 1 + \log 2 + \cdots + \log n)$. Ước lượng độ lớn của T_n theo ký hiệu Θ.

Bài 3. *(4 điểm)* Cho dãy số nguyên có n số $a_1, a_2, \ldots, a_n$ mà giá trị mỗi số thuộc tập hợp $\{0,1,2\}$. Cần đếm xem có tất cả mấy cách chọn ra cặp chỉ số (L, R) với $1 \le L < R \le n$ sao cho trong dãy con của a xét từ vị trí L đến vị trí R thì có một số nào đó xuất hiện từ 3 lần trở lên. Ví dụ: *nếu* $a = [0,1,2,2,2]$ *thì đáp số là* 3*, ứng với* 3 *cách chọn cặp chỉ số* $(L,R) = (3,5),\ (2,5),\ (1,5)$.

1) Hãy mô tả cách vét cạn naive cho bài toán trên (dùng mã giả hoặc code C++/Python) và đánh giá độ phức tạp tương ứng của thuật toán đó.

2) Bằng cách sử dụng thêm các dãy phụ x, y, z trong đó x_i cho biết số giá trị bằng 0 trong dãy từ a_1 đến a_i; tương tự y_i và z_i lần lượt cho biết số giá trị bằng 1 và bằng 2, hãy đề xuất cải tiến cách làm trên thành $O(n^2)$.

3) Bằng cách sử dụng nguyên lý Dirichlet, hãy tìm cách cải tiến cách làm trên thành $O(n)$.

HẾT