PHÂN TÍCH ĐỘ PHỨC TẠP
CỦA THUẬT TOÁN KHÔNG ĐỆ QUY

Nhóm 9

❖ **Bài 1:**

```
ALGORITHM    Mystery(n)
    //Input: A nonnegative integer n
    S ← 0
    for i ← 1 to n do
        S ← S + i ∗ i
    return S
```

a. Output của thuật toán này là gì?

Tính tổng $S(n) = \sum_{i=1}^{n} i^2$.

b. Basic operation của thuật toán này là gì?

Basic operation của thuật toán là phép nhân.

c. Tính số lần thực thi basic operation? (Tính $C(n)$)

$$C(n) = \sum_{i=1}^{n} 1 = n.$$

d. Lớp hiệu năng của thuật toán?

$C(n) \in \Theta(n)$.

e. Cải thiện hoặc đề xuất một thuật toán tốt hơn và xác định lớp hiệu năng? (Nếu có)

Sử dụng công thức $\sum_{i=1}^{n} i^2 = \frac{n(n+1)(2n+1)}{6}$. Khi đó $C(n) \in \Theta(1)$.

❖ **Bài 2:**

```
ALGORITHM  Secret(A[0..n − 1])
    //Input: An array A[0..n − 1] of n real numbers
    minval ← A[0]; maxval ← A[0]
    for i ← 1 to n − 1 do
        if A[i] < minval
            minval ← A[i]
        if A[i] > maxval
            maxval ← A[i]
    return maxval − minval
```

a. Output của thuật toán này là gì?

Tính hiệu của phần tử lớn nhất và phần tử nhỏ nhất trong mảng.

b. Basic operation của thuật toán này là gì?

Basic operation của thuật toán là phép so sánh.

c. Tính số lần thực thi basic operation? (Tính $\mathrm{C}(\mathrm{n})$)

$$\mathrm{C}(\mathrm{n}) = \sum_{i=1}^{n-1} 2 = 2(n-1)$$

d. Lớp hiệu năng của thuật toán?

$\mathrm{C}(\mathrm{n}) \in \Theta(\mathrm{n})$.

e. Cải thiện hoặc đề xuất một thuật toán tốt hơn và xác định lớp hiệu năng? (Nếu có)

Không.

❖ Bài 3:

ALGORITHM *Enigma*$(A[0..n-1, 0..n-1])$

```
//Input: A matrix A[0..n − 1, 0..n − 1] of real numbers
for i ← 0 to n − 2 do
    for j ← i + 1 to n − 1 do
        if A[i, j] ≠ A[j, i]
            return false
return true
```

a. Output của thuật toán này là gì?

Thuật toán trả về true nếu ma trận A đối xứng và false nếu ngược lại.

b. Basic operation của thuật toán này là gì?

Basic operation của thuật toán là phép so sánh.

c. Tính số lần thực thi basic operation? (Tính $\mathrm{C(n)}$)

$$\mathrm{C(n)} = \sum_{i=0}^{n-2} \sum_{j=i+1}^{n-1} 1 = \sum_{i=0}^{n-1} n - 1 - i = \frac{n(n-1)}{2}$$

d. Lớp hiệu năng của thuật toán?

$\mathrm{C(n)} \in \Theta(n^2)$.

e. Cải thiện hoặc đề xuất một thuật toán tốt hơn và xác định lớp hiệu năng? (Nếu có)

Không.

❖ **Bài 4:**

ALGORITHM $GE(A[0..n-1, 0..n])$

//Input: An $n \times (n+1)$ matrix $A[0..n-1, 0..n]$ of real numbers

for $i \leftarrow 0$ **to** $n-2$ **do**

  for $j \leftarrow i+1$ **to** $n-1$ **do**

    for $k \leftarrow i$ **to** n **do**

      $A[j, k] \leftarrow A[j, k] - A[i, k] * A[j, i] \,/\, A[i, i]$

a. Xác định lớp hiệu năng thời gian của thuật toán này?

$C(n) = \sum_{i=0}^{n-2}\sum_{j=i+1}^{n-1}\sum_{k=i}^{n} 1 = \sum_{i=0}^{n-2}\sum_{j=i+1}^{n-1} n-i+1 = \sum_{i=0}^{n-2}(n-i+1)(n-i-1)$

$= \sum_{j=1}^{n-1} j(j+2) = \sum_{j=1}^{n-1} j^2 + \sum_{j=1}^{n-1} 2j = \frac{n(n-1)(2n-1)}{2} + n(n-1)$

$= \frac{n(n-1)(2n+5)}{6} \in \Theta(n^3)$.

b. Xác định sự kém hiệu quả trong thuật toán này và làm sao để tăng tốc độ của thuật toán?

Phép chia A[*j, i*] / A[*i, i*] nằm trong vòng lặp của biến *k* nhưng không phụ thuộc vào biến *k*. Ta có thể dùng một biến tạm để lưu giá trị này trước vòng lặp của biến *k*, từ đó giảm độ phức tạp thời gian và tăng tốc độ thuật toán.