

CẤU TRÚC DỮ LIỆU & GIẢI THUẬT HOMEWORK 1

---oOo---

1. Tính độ phức tạp của các giải thuật sau:

Giải thuật a:

```
for (i = 0; i < n; i++)  
    for j = 0; j < n; j++)  
        b[i][j] += c;
```

Giải thuật b:

```
for (i = 0; i < n; i++)  
    for (j = i+1; j < n; j++)  
        b[i][j] -= c;
```

2. Tính độ phức tạp của các giải thuật sau trong trường hợp tốt nhất, trường hợp xấu nhất:

Giải thuật a:

```
for (i = 0; i < n; i++)  
    if (a[i] == k) return 1;  
return 0;
```

Giải thuật b:

```
void SelectionSort(int a[], int n)  
{  
    int min;  
    int tmp;  
    for (int i = 0; i < n; i++) {  
        min = i;  
        for (int j = i + 1; j < n; j++)  
            if (a[j] < a[min]) min = j;  
        if (a[min] < a[i]) {  
            tmp = a[i]; a[i] = a[min]; a[min] = tmp;  
        }  
    }  
}
```

3. Xác định giới hạn trên $O(g(n))$ của các hàm sau:

- a. $f(n) = 10n^2 - 3n + 20$
- b. $f(n) = \log n + 100$
- c. $f(n) = n \log n + \log n + 5$

4. Phát biểu nào là đúng, vì sao ?

- a. $2^{n+1} = O(2^n)$
- b. $2^{2n} = O(2^n)$

5. Tính giới hạn trên (big-O) của độ phức tạp của các giải thuật sau:

a. Giải thuật cộng ma trận:

```
for (i = 0; i < n; i++)  
    for (j = 0; j < n; j++)  
        a[i][j] = b[i][j] + c[i][j];
```

b. Giải thuật nhân ma trận:

```
for (i = 0; i < n; i++)  
    for (j = 0; j < n; j++)  
        for (k = 0; k < n; k++)  
            a[i][j] += b[i][k]*c[k][j];
```

c. Giải thuật chuyển vị ma trận:

```
for (i = 0; i < n-1; i++)  
    for (j = i+1; j < n; j++) {  
        tmp = a[i][j];  
        a[i][j] = a[j][i];  
        a[j][i] = tmp;  
    }
```

--- Hết ---