***BÀI TẬP CHƯƠNG 1***

**Bài 1: Đếm số phép toán gán, phép so sánh được thực thi và xác định độ phức tạp trong đoạn code sau:**

**for (i = 0; i < n; i++)**

**for (j = 0; j < m; j++)**

**if (a[ i ][ j ] = = x)**

**return 1;**

**return -1;**

**Phép so sánh**

- Phép so sánh lệnh for của m:

+ m = 0 có 1 phép so sánh

+ m = 1 có 3 phép so sánh

+ m = 2 có 5 phép so sánh

+…

+ m = k có 2k+1

- Phép so sánh lệnh for của n:

+ n = 0 có 1 phép so sánh và 0 vòng lặp for của m

+ n = 1 có 2 phép so sánh và 1 vòng lặp for của m

+ n = 2 có 3 phép so sánh và 2 vòng lặp for của m

+…

+ n = k có k+1 phép so sánh và k vòng lặp for của m

*=>* ***Tổng số phép so sánh là : k + 1 + k\*(2k + 1)***

**Phép gán**

-Phép gán lệnh for của m:

+ m = 0 có 1 phép gán

+ m = 1 có 2 phép gán

+ m = 2 có 3 phép gán

+…

+ m = k có k+1 phép gán

-Phép gán lệnh for của n:

+ n = 0 có 1 phép gán và 0 vòng lập for của m

+ n = 1 có 2 phép gán và 1 vòng lập for của m

+ n = 2 có 3 phép gán và 2 vòng lập for của

+…

+ n = k có k+1 phép gán và k vòng lập for của m

* ***Tổng số phép gán là : k+1 + k\*(k+1) +1 = k2 + 2k + 2***

Độ phức tạp của thuật toán: Chọn phép gán để tính độ phức tạp

T(n) = k2 + 2k + 2

Ta thấy T(n) thuộc lớp O(n2)

**Kết luận : độ phức tạp của thuật toán thuộc lớp O(n2)**