Bài 2: Đếm số phép toán gán, phép so sánh được thực thi và xác định độ phức tạp trong đoạn code sau:

sum = 0;

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

for (j = 0; j < i ; j++)

sum++;

giải:

\*Tính độ phức tạp

* Số phép gán cho for j:

i = 0 có 1 phép gán cho j; 0 phép cho sum.

i = 1 có 2 phép gán cho j; 1 phép cho sum.

i = 2 có 3 phép gán cho j; 2 phép cho sum.

Suy ra có i + 1 phép gán cho j và i phép gán cho sum => có 2i + 1 phép gán cho for j

* Số phép gán cho for i:

n = 0 có (2i + 1) phép gán for j \* 0 + 1;

n = 1 có (2i + 1) phép gán for j \* 1 + 2;

n = 2 có (2i + 1) phép gán for j \* 2 + 3;

Suy ra T(i,n)  = (2i + 1) \* n + n + 1 phép gán.

* sum = 0 lúc đầu nên có thêm 1 phép gán nữa

=> T(i,n)  = (2i + 1) \* n + n + 1 + 1

= 2\*i\*n + 2n + 2.

Mà i tối đa là n – 1.

=>T(n) = 2\*(n – 1)\*n + 2n +2

= 2n2 + 2. => T(O) = n2

\*số phép gán tối đa: 2n2 + 2 phép so sánh

\*số phép so sánh tối đa:

For i :

n = 0 thì có 1 phép so sánh vs i;

n = 1 thì có 2 phép so sánh vs i;

n = 3 thì có

=>có n phép so;

For j:

i = 0 thì có 2 phép so sánh vs i;

i = 1 thì có 3 phép so sánh vs i;

=> có i phép so sánh;

* có n + i phép so sánh tổng quát

mà (i = n) phép so sánh nên có 2n phép so sánh tối đa.