Bài 1: Đếm phép gán và Phép so sánh.

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

{

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

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

return 1;

}

return -1;

}

* Phép gán:

Lần n = 0 thực hiện 1 lần

Lần n = 1 thực hiện 3 + 2\*m lần

Lần n = 2 thực hiện 5 + 4\*m lần

……………………………..

Lần n = n thực hiện (n + 2)+ (n+1) \* m

* T(n) = O(n2)
* Phép so sánh

Lần n = 0 thực hiện 1 lần

Lần n = 1 thực hiện 2 + 2\*m lần

Lần n = 2 thực hiện 3 + 4\*m lần

……………………………..

Lần n = n thực hiện (n + 1) + (n \* 2) \* m

* T(n) = O(n2)

Bài 2: Đếm phép gán và Phép so sánh.

sum = 0;

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

{

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

{

sum++;

}

}

* Phép gán:

Lần n = 0 thực hiện 2 lần

Lần n = 1 thực hiện 4 + 3\*m lần

Lần n = 2 thực hiện 6 + 5\*m lần

……………………………..

Lần n = n thực hiện (n + 1) \* 2 + n\*m

* T(n) = O(n2)
* Phép so sánh

Lần n = 0 thực hiện 1 lần

Lần n = 1 thực hiện 2 + 2\*m lần

Lần n = 2 thực hiện 3 + 4\*m lần

……………………………..

Lần n = n thực hiện (n + 1) + (n \* 2) \* m

* T(n) = O(n2)

Bài 3: Đếm phép gán và Phép so sánh.

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

sum1+=i; }

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

sum2+=i;}

* Phép gán:

Lần n = 0 thực hiện 1 + 1 lần

Lần n = 1 thực hiện 3 + 3 lần

Lần n = 2 thực hiện 7 + 14 lần

……………………………..

Lần n = n thực hiện (n + 1) + (2\*n + 1)

* T(n) = O(n)
* Phép so sánh

Lần n = 0 thực hiện 1 + 1 lần

Lần n = 1 thực hiện 2 + 3 lần

Lần n = 2 thực hiện 3 + 5 lần

……………………………..

Lần n = n thực hiện (n + 1) + (2\*n +1)

* T(n) = O(n)

Bài 4: Đánh giá độ giải thuật của đoạn code sau

int GT (int n)

{

if (n == 1)

return 1;

return n \* GT(n-1);

}

* T(n) = n.

Bài 5:

int Fibo (int n)

{

if (n <=1) {

return n; }

return Fibo(n-1) + Fibo(n-2);

}

* T(n) = n + n – 1 = O(n)