Big O is a complexity of a function

1. Critical operation
2. How many times critical operation is performed

What is a critical operation?

1. Deal with what the function is computing.
2. Number of times it occurs needs to vary with size of input.

Int search (int [] arr, int target){

For(int i= 0; i<arr.length; i++){

If(arr[i] == target){

return target;

}

Throw new Exception (“No such element”);

}

O(n) = n

Int search (int [] arr, int target){

For(int i= 0; i<arr.length; i++){

For(int j = 0; j < arr.length; j++){

If(arr[i] + arr{j} == target){

Return target;

}

}

Throw new Exception (“No such element”);

}

Critical operation: the retrieving of arrays

O(n) = n2

Int binary-search(u >= 1){

int mid = (

n/2k  =1

k = log2n

Recursive Big o

T(n) = f(n) + T(n-1)