## Determining Dominant Terms:

Steps :

1.) replace constants with \_\_\_

2.) Compare terms (can use limits)

3.) Remove \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

If more than one variable (M, N) \_\_\_\_\_\_\_ replace one variable with another

1.) M \_\_\_\_ N \_\_\_\_

2.) M \_\_\_\_ N \_\_\_\_

3.) M \_\_\_\_ N \_\_\_\_

Ex:

TC = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ——> O (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

TC = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ——> O (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

TC = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ——> O (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

TC = \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ——> O (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

## If Statements:

Compute TC for \_\_\_\_\_\_\_\_ branch.

Best case \_\_\_\_\_\_\_\_\_\_\_\_\_, worst case \_\_\_\_\_\_\_\_\_\_\_ average case = (\_\_\_\_\_ + \_\_\_\_) / \_\_

For General case give the \_\_\_\_\_\_\_\_\_ case

|  |  |
| --- | --- |
| If ( \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ) {  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  }  else {  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  } | Work: |

Best: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Worst: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ General: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

–-----------------------------------------------------------------------------------------------------

|  |  |
| --- | --- |
| if ( \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ) {  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  }  else {  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  } | Work: |

Best: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Worst: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ General: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## Function Calls:

* The TC for a function call is based on the time complexity of the \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ and the \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ when called.
* TC for a function definition can only depend on \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.
* Alternative variables used to describe the TC must first be \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

void funct1(int X) has TC: O(X2) —> funct1(Y) TC: O( \_\_\_\_\_\_\_ )

—> funct1(A+B) TC: O( \_\_\_\_\_\_\_ )

float funct2(int K, int V) has TC: O(V) —> funct2(S, R) TC: O( \_\_\_\_\_\_\_ )

—> funct2(V, K) TC: O( \_\_\_\_\_\_\_ )

—> funct2(V, 100) TC: O( \_\_\_\_\_\_\_ )

void funct3(int N, int P, int Q) has TC: O(P + NQ) —> funct3(D, M, L) TC: O( \_\_\_\_\_\_\_\_\_\_\_ )

—> funct3(D, T\*T, L) TC: O( \_\_\_\_\_\_\_\_\_\_ )

int funct4(int B, int C, int D) has TC: O(D) —> funct2(S, R, 6) TC: O( \_\_\_\_\_\_\_ )