**Acitivity3**

Submit in a word document or a text file.

Give tilde approximations for the following quantities:

1. N + 1

ANS: N

1. 1 + 1/N

ANS :1

1. (1 + 1/N )(1 + 2/N )

ANS: 2/N^2

1. 2N^3 - 15 N^2 + N

ANS: 2N^3

1. lg(2N )/lg N

ANS: lg(2N)

1. lg(N^2 + 1) / lg N

ANS: lgN^2

**Activity4**

Submit in a word document or in a text file

Give the order of growth (as a function of N ) of the running times of each of the following code fragments:

**Note:** use the following Notations for writing Answers: N, 1, log(N), N^2, N\*log(N), 2^N, N^3, log(N)

**Code-1:**     int count = 0;

            for(int i = 0; i < N; i++){

                count++;

            }

ANS: N

**Code-2:**int sum = 0;

            if(sum == 0){

                sum++;

}

ANS: 1

**Code-3:** for(int i = N; i > 0; i < N/2){

            int sum = 0;

}

ANS: lgN

**Code-4:** for(int i = 0; i < N; i++){

                for(int j = 0; j < N; j++){

                    System.out.println(“Hello”);

                }

            }

ANS: N^2

**Code-5:** for(int i = 0; i < N; i++){

                for(int j = 0; j < N; j = j \* 2){

                    System.out.println(“Hello”);

                }

            }

ANS: NlgN

**Code-6:** public int fibonacci(int number) {

if (number <= 1) {

                     return number;

                } else {

                         return fibonacci(number - 1) + fibonacci(number - 2);

                }

}

ANS: N^2

**Activity5**

Submit in a word document or in a text file

Give the order of growth (as a function of N ) of the running times of each of the following code fragments:

**Note:** use the following Notations for writing Answers: N, 1, log(N), N^2, N\*log(N), 2^N, N^3, log(N)

**Code-1:**     int sum = 0;  
        for (int n = N; n > 0; n /= 2)  
            for(int i = 0; i < n; i++)  
                Sum++;

  ANS: N

**Code-2:** int sum = 0;  
        for (int i = 1 i < N; i \*= 2)  
            for (int j = 0; j < i; j++)  
                sum++;

ANS: N

**Code-3:** int sum = 0;  
        for (int i = 1 i < N; i \*= 2)  
            for (int j = 0; j < N; j++)  
                Sum++;

ANS: NlgN

Top of Form

Bottom of Form

Top of Form

Bottom of Form

Top of Form

Bottom of Form