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++;

**Time complexity: N – both the loops run logN times**

**Code-2:**

int sum = 0;  
        for (int I = 1 I < N; I \*= 2)  
            for (int j = 0; j < I; j++)  
                sum++;

**Time complexity: N – both the loops run logN times**

**Code-3:**

int sum = 0;  
        for (int I = 1 I < N; I \*= 2)  
            for (int j = 0; j < N; j++)  
                Sum++;

**Time complexity: NlogN – inner loop runs N times and outer loop runs logN times**