A Time Complexity Question – By GeeksForGeeks

What is the time complexity of following function fun()? Assume that log(x) returns log value in base 2.

```
void fun()
{
   int i, j;
   for (i=1; i<=n; i++)
      for (j=1; j<=log(i); j++)
         printf("GeeksforGeeks");
}
```

Time Complexity of the above function can be written as Θ(log 1) + Θ(log 2) + Θ(log 3) + ... + Θ(log n) which is Θ (log n!)

Order of growth of ‘log n!’ and ‘n log n’ is same for large values of n, i.e., Θ (log n!) = Θ(n log n). So time complexity of fun() is Θ(n log n).

The expression Θ(log n!) = Θ(n log n) can be easily derived from following Stirling’s approximation (or Stirling’s formula).

```
log n! = n log n - n + O(log(n))
```

Sources:
http://en.wikipedia.org/wiki/Stirling%27s_approximation