No.
Date

算法分析

典型算法时间函数:

1° 1 执行1~几次的指令(常数次)

2° $\log N$ (对数次) 比N稍慢. 底数对其值的影响不大.

3° N (线性)

4° $N\log N$

5° N^2 (二次性) 该算法只对较小的问题实用.

6° N^3 (立方性) 该算法只对小问题实用.

7° 2^N (指数性) 蛮力算法.

注:①在计算科学中 $\log_2 N$ 常简写为 $\lg N$.

② $\lfloor X \rfloor$: 小于等于X的最大整数.

$\lceil X \rceil$: 大于等于X的最小整数.

No.

Date

基本递推式

1、 循环遍历输入删除某项:

$$C_N = C_{N-1} + N \quad (N \geq 2,\ C_1 = 1)$$

$$C_N = \frac{N(N+1)}{2} \approx \frac{N^2}{2}$$

2. 用一个步骤对分输入的递归程序:

$$C_N = C_{N/2} + 1 \quad ,\ (N \geq 2,\ C_1 = 1)$$

$$C_{2^n} = n+1 \quad ; \quad C_n \approx \lg N.$$

3. 对分输入,但也许必须检查输入中的每一项的递归程序:

$$C_N = C_{N/2} + N \quad (N \geq 2,\ C_1 = 0)$$

$$C_N \approx 2N$$

4. 在等分输入之前,之中,之后,必须线性遍历输入的递归程序:

$$C_N = 2C_{\frac{N}{2}} + N. \quad (N \geq 2,\ C_1 = 0)$$

$$C_N \approx N \lg N$$

5. 等分输入,然后执行其它定量工作的递归程序:

$$C_N = 2C_{N/2} + 1. \quad (N \geq 2,\ C_1 = 0)$$

$C_N \approx$ [illegible]N