

哈工大计算机考研全套视频和资料，真题、考点、典型题、命题规律独家视频讲解！
详见：网学天地（www.e-studysky.com）；咨询QQ：2696670126

数据结构与算法	第二章 算法及其性能度量	2—1
<h2>第2章 算法评价和复杂性分析</h2> <p>教学目的和内容：</p> <ul style="list-style-type: none">了解分析和评价算法的一些基本准则掌握算法时间复杂性分析方法 <h3>2.1 算法及其性能评价准则</h3> <h3>2.2 算法时间复杂性分析方法</h3> <p>哈尔滨工业大学计算机科学与技术学院 张岩</p>		

数据结构与算法	第二章 算法及其性能度量	2—2
<h3>2.1 算法及其性能评价准则</h3> <p>一、算法、算法的特征和算法描述</p> <p>算法（Algorithm）：是对特定问题求解步骤的一种描述，它是指令（规则）的有限序列，其中每一条指令表示一个或多个操作。</p> <p>算法的特征：</p> <ul style="list-style-type: none">①有穷性、②确定性、③能行性、④输入、⑤输出 <p>算法描述：</p> <ul style="list-style-type: none">①自然语言；②程序设计语言；③类语言*； <p>常用的算法设计方法：</p> <ul style="list-style-type: none">①递归法（Recursion）、②分治法（Divide-and Conquer）、③贪心法（Greedy）、④动态规划（Dynamic Programming）、⑤搜索与遍历、⑥回溯（Backtracking）、⑦解空间局部搜索⑧近似算法（Approximation）、⑨在线算法（On-Line）等 <p>哈尔滨工业大学计算机科学与技术学院 张岩</p>		

数据结构与算法	第二章 算法及其性能度量	2—3
<h3>2.1 算法及其性能评价准则</h3> <p>二、一个好的算法的标准</p> <h4>1. 正确性 (Correctness)</h4> <ul style="list-style-type: none">●正确性的含义：是指对于一切合法的输入数据经有限时间或有限步后均可得到正确（满足规格说明要求）的结果；●算法包括两方面的内容：① 解决问题的方法；② 实现这一方法的一系列指令（语句、步骤）●算法的正确性证明：①需要一组相关的引理和定理，确认一个算法所使用的方法和公式的正确性；②在证明一系列的语句确实做了符合规定的动作。●算法的正确性的严格的形式化证明还未取得突破，还是一项令人生畏的工作。只有那些比较简单的算法，其正确性才能被形式化证明。 <p>哈尔滨工业大学计算机科学与技术学院 张岩</p>		

数据结构与算法	第二章 算法及其性能度量	2—4
<h3>2.1 算法及其性能评价准则</h3> <p>二、一个好的算法的标准</p> <h4>2. 时间复杂性 (Time Complexity)</h4> <ul style="list-style-type: none">●如何计算和比较算法的执行时间：(1) 实验测量法（实际执行时间、执行指令的条数）<ul style="list-style-type: none">优点：精确缺点：必须先运行根据算法编制的程序；所得的时间统计量依赖于计算机的硬件、软件等环境因素，容易掩盖算法本身的优劣。 <p>哈尔滨工业大学计算机科学与技术学院 张岩</p>		

哈工大计算机考研全套视频和资料，真题、考点、典型题、命题规律独家视频讲解！
详见：网学天地（www.e-studysky.com）；咨询QQ：2696670126

数据结构与算法	第二章 算法及其性能度量	2—5
2.1 算法及其性能评价准则		
二、好的算法的标准		
(2)事前分析(估计)法		
高级语言编写的程序在计算机上运行时间取决于下列因素：		
①依据的算法本身选择何种策略		
②问题的规模（输入的规模，输入的大小）。如求素数问题		
③程序设计语言：算法的实现语言级别越高，执行效率越低		
④编译程序产生的机器代码的质量		
⑤及其执行指令的速度		
③④⑤表明用绝对的时间单位衡量一个算法的效率是不适合的		
●算法的时间复杂性		
①认为一个特定的算法执行时间(运行工作量)，只依赖于问题的规模，即算法执行时间是问题规模的函数		
哈尔滨工业大学计算机科学与技术学院 张岩		

数据结构与算法	第二章 算法及其性能度量	2—6
2.1 算法及其性能评价准则		
二、好的算法的标准		
②由于算法的执行时间由组成算法的控制结构（顺序、分支和循环）和基本操作的综合效果，因此从算法中选择对于研究问题来说是基本的操作（基本运算），把算法的基本操作的重复执行的次数作为算法的时间度量		
③通常，我们并不关心 $T(n)$ 的具体值，而是关心它的增长率，即 $T(n)$ 随问题规模 n （是一个整数）的增大的变化趋势（界限）		
④算法的复杂性定义		
算法中基本操作重复执行的次数是问题规模 n 的某个函数 $f(n)$ ，算法的时间度量记作 $T(n) = O(f(n))$ 。它表示随问题规模 n 的增大，算法执行时间的增长率不会超过 $f(n)$ ，称为算法的渐进时间复杂性，简称时间复杂性。		
哈尔滨工业大学计算机科学与技术学院 张岩		

数据结构与算法	第二章 算法及其性能度量	2—7
2.1 算法及其性能评价准则		
二、好的算法的标准		
3.空间复杂性(Space Complexity)		
●算法的空间复杂性是指算法在执行过程中的存储量需求		
●一个算法的存储量需求除了存放算法本身所有的指令、常数、变量和输入数据外，还包括对数据进行操作的工作单元和存储实现计算所需信息的辅助空间		
●算法的存储量需求与输入的规模、表示方式、算法采用的数据结构、算法的设计以及输入数据的性质有关		
●算法的执行的不同时刻，其空间需求可能是不同的		
●算法的空间复杂性是指算法在执行过程中的最大存储量需求		
●空间复杂性的渐进表示----空间复杂度		
$T(n) = O(f(n))$ 其中， n 为问题的输入规模		
哈尔滨工业大学计算机科学与技术学院 张岩		

数据结构与算法	第二章 算法及其性能度量	2—8
2.1 算法及其性能评价准则		
二、好的算法的标准		
4.可读性(Readability)		
●可读性好的算法有助于设计者和他人阅读、理解、修改和重用		
●晦涩难懂的算法不容易隐藏错误，而且还增加了阅读...难度		
●可读性好的算法，常常也具有简单性		
5.健壮性(Robustness)		
●当输入数据非法时，能作出适当的反应（如对输入数据进行语法检查，提出修改输入建议并提供重新输入的机会），避免异常出错		
6.一个好的算法还应该具有灵活性（Flexibility）、可重用性（Reuseable）和自适应性（Adaptability）等		
哈尔滨工业大学计算机科学与技术学院 张岩		

哈工大计算机考研全套视频和资料，真题、考点、典型题、命题规律独家视频讲解！
详见：网学天地（www.e-studysky.com）；咨询QQ：2696670126

数据结构与算法	第二章 算法及其性能度量	2—9
2.2 算法时间复杂性分析方法		
一、算法的时间复杂性		
<p>定义2.1 设一个领域问题的输入的规模为n，D_n是该领域问题的所有输入集合，任意一个输入$I \in D_n$是$P(I)$是I出现的概率，$\sum_{I \in D_n} P(I) = 1$，$T(I)$是（解决该领域问题的）算法在输入$I$下所执行的基本运算的次数。</p> <p>称</p> $E(n) = \sum_{I \in D_n} \{P(I) * T(I)\}$ <p>为该算法的期望复杂性（平均时间复杂性）</p> <p>称</p> $W(n) = \max_{I \in D_n} \{T(I)\}$ <p>为该算法的最坏时间复杂性</p> <p>举例</p>		
哈尔滨工业大学计算机科学与技术学院 张岩		

数据结构与算法	第二章 算法及其性能度量	2—10
2.2 算法时间复杂性分析方法		
一、算法的时间复杂性		
<p>例2.1 A是一个由n个不同元素的实数数组，给出求其最大和最小元素的算法和时间复杂性</p> <pre>void SM(double A[], int n, double max, double min) { double max, min; max=min=A[0]; for (k=1; k<=n-1; k++) { if (A[k]>max) max=A[k]; if (A[k]<min) min=A[k]; } }</pre> <p>容易看出，算法SM对大小为n的数组需要$2(n-1)$比较，若算法SM的基本运算是元素比较，则复杂性为$2(n-1)$</p>		
哈尔滨工业大学计算机科学与技术学院 张岩		

数据结构与算法	第二章 算法及其性能度量	2—11
2.2 算法时间复杂性分析方法		
一、算法的时间复杂性		
<p>例2.2 A是一个由n个不同元素的实数数组，给出确定给定实数K是否在A中的算法及其时间复杂性</p> <pre>int SK(double A[], int n, double K) { int j=1; while (j<=n){ if (A[j]==K) break; else j++; } return j; } //若j<=n, 则K在A中, 否则(j=n+1) K不在A中</pre>		
哈尔滨工业大学计算机科学与技术学院 张岩		

数据结构与算法	第二章 算法及其性能度量	2—12
2.2 算法时间复杂性分析方法		
一、算法的时间复杂性		
<ul style="list-style-type: none">•算法SK的基本运算是元素与K的比较•假定q表示K在A中的概率，有假设K等于A的每个元素有相同的概率。令$D_n = \{I_1, I_2, \dots, I_n, I_{n+1}\}$，其中$I_j (1 \leq j \leq n)$表示$K=A[j]$的任意输入，$I_n$表示K不是A中元素的任意一个输入，•由上述说明有：当$1 \leq j \leq n$时，$P(I_j) = q/n$，$T(I_j) = j$，$P(I_{n+1}) = 1-q$，$T(I_{n+1}) = n$•算法SK的期望复杂性和最坏复杂性为 $E(n) = \sum_{j \in D_n} \{P(I_j) * T(I_j)\}$ $= \sum_{j=1, \dots, n+1} [(q/n)*j] + (1-q)*n$ $= 1/2(n+1)q + (1-q)n = 1/2(n+1) (K在A中, q=1)$ $W(n) = \max \{T(I_j) 1 \leq j \leq n\} = n$		
哈尔滨工业大学计算机科学与技术学院 张岩		

数据结构与算法 第二章 算法及其性能度量 2—13

2.2 算法时间复杂性分析方法

二、函数阶的比较

定义2.2 设 $f(n)$ 、 $T(n)$ 是整数集到实数集上的函数，称函数 $f(n)$ 是 $T(n)$ 增长率的上界，当且仅当存在一个正常数 C 和整数 n_0 ，使得对任意的 $n \geq n_0$ 时，有

$$T(n) \leq C f(n)$$

记作：

$$T(n) = O(f(n))$$

此时也称 $T(n)$ 的阶至多为 $f(n)$ ）

*一个算法的时间复杂性为 $O(f(n))$ ，表明它的基本操作次数至多是 $f(n)$ 的某个常数倍

例2.1 设函数 $T(n) = 3n^5 + 4n^2 + 1$ ，证明： $T(n) = O(n^5)$

证明： $f(n) = n^5$ 。取 $n_0 = 0, C = 8$ ，则当 $n \geq n_0$ 时有

$$T(n) = 3n^5 + 4n^2 + 1 \leq 8n^5 = C f(n) \quad \text{证毕}$$

哈尔滨工业大学计算机科学与技术学院 张岩

数据结构与算法 第二章 算法及其性能度量 2—14

2.2 算法时间复杂性分析方法

二、函数阶的比较

定义2.3 设 $f(n)$ 、 $T(n)$ 是整数集到实数集上的函数，称函数 $f(n)$ 是 $T(n)$ 增长率的下界，当且仅当存在一个正常数 C 和整数 n_0 ，使得对任意的 $n \geq n_0$ 时，有

$$T(n) \geq C f(n)$$

记作：

$$T(n) = \Omega(f(n))$$

此时也称 $T(n)$ 的阶至少为 $f(n)$ ）

*一个算法的时间复杂性为 $\Omega(f(n))$ ，表明它的基本操作次数至少是 $f(n)$ 的某个常数倍

例2.3 设函数 $T(n) = 3n^5 + 4n^2 + 1$ ，证明： $T(n) = \Omega(n^5)$

证明： $f(n) = n^5$ 。取 $n_0 = 0, C = 1$ ，则当 $n \geq n_0$ 时有

$$T(n) = 3n^5 + 4n^2 + 1 \geq 1n^5 = C f(n) \quad \text{证毕}$$

哈尔滨工业大学计算机科学与技术学院 张岩

数据结构与算法 第二章 算法及其性能度量 2—15

2.2 算法时间复杂性分析方法

定理2.1 若

$$A(n) = a_m n^m + \dots + a_1 n + a_0$$

是关于 n 的 m 次多项式，则

$$A(n) = O(n^m)$$

*此定理说明，时间复杂性仅取决于多项式的最高次幂，而与最高次幂的系数和其他低次项无关

- $O(1)$ 表示时间复杂性为常数
- 常见的时间复杂性及其比较

$$O(1) < O(\log \log n) < O(\log n) < O(n) < O(n \log n) < O(n^2) < O(n^3) < O(2^n)$$

哈尔滨工业大学计算机科学与技术学院 张岩

数据结构与算法 第二章 算法及其性能度量 2—16

2.2 算法时间复杂性分析方法

程序运行时间比较 $T(n) = O(f(n))$

哈尔滨工业大学计算机科学与技术学院 张岩

哈工大计算机考研全套视频和资料，真题、考点、典型题、命题规律独家视频讲解！
 详见：网学天地（www.e-studysky.com）；咨询QQ：2696670126

数据结构与算法 第二章 算法及其性能度量 2—17

2.2 算法时间复杂性分析方法

三、时间复杂性的运算法则

设 $T_1(n)=O(f(n))$, $T_2(n)=O(g(n))$, 则

- 加法规则: $T_1(n)+T_2(n)=O(\max\{f(n), g(n)\})$
- 乘法规则: $T_1(n) \cdot T_2(n)=O(f(n) \cdot g(n))$

哈尔滨工业大学计算机科学与技术学院 张岩

数据结构与算法 第二章 算法及其性能度量 2—18

2.2 算法时间复杂性分析方法

例2.4:

```

① s = 0;
   → f(n) = 1; T1(n) = O(f(n)) = O(1)           常量阶
② for ( i=1; i <= n; ++i ) { ++x; s += x; }
   → f(n) = 3n+1; T1(n) = O(f(n)) = O(n)         线性阶
③ for ( i=1; i <= n; ++i )
    for ( j=1; j <= n; ++j ) { ++x; s += x; }
   → f(n) = 3n2+2n+1; T3(n) = O(f(n)) = O(n2)     平方阶
④ for ( i=1; i <= n; ++i )
    for ( j=1; j <= n; ++j )
        for ( k=1; k <= n; ++k )
            c[i][j] += a[i][k] * b[k][j]; }
   → f(n) = 2n3+3n2+2n+1; T4(n) = O(f(n)) = O(n3)   立方阶
    
```

哈尔滨工业大学计算机科学与技术学院 张岩

数据结构与算法 第二章 算法及其性能度量 2—19

2.2 算法时间复杂性分析方法

例2.5:

```

⑤ Long fact ( int n )
{ if ( n==0 ) || ( n==1 )
  return( 1 );
else
  return( n * fact( n - 1 ) );
}
    
```

$T(n) = \begin{cases} C & \text{当 } n=0, n=1 \\ G + T(n-1) & \text{当 } n > 1 \end{cases}$

取 $f(n) = n$

$\therefore T(n) = O(f(n)) = O(n)$

$T(n) = G + f(n-1)$
 $T(n-1) = G + f(n-2)$
 $T(n-2) = G + f(n-3)$
 \dots
 $T(2) = G + f(1)$
 $+ T(1) = C$
 $T(n) = G \cdot (n-1) + C$

哈尔滨工业大学计算机科学与技术学院 张岩

数据结构与算法 第二章 算法及其性能度量 2—20

2.2 算法时间复杂性分析方法

作业:

1.p28 例2.1.5

2. A是一个由n个不同元素的实数数组，给出求其最大和最小元素的递归算法和时间复杂性

$\ln \log \leq \geq \neq O \Omega \theta$

哈尔滨工业大学计算机科学与技术学院 张岩