

等级考试公共基础考点分析之数据结构与算法(2) PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/133/2021_2022__E7_AD_89_E7_BA_A7_E8_80_83_E8_c97_133619.htm

考点2 算法的复杂度

1 算法的时间复杂度 算法的时间复杂度，是指执行算法所需要的计算工作量。同一个算法用不同的语言实现，或者用不同的编译程序进行编译，或者在不同的计算机上运行，效率均不同。这表明使用绝对的时间单位衡量算法的效率是不合适的。撇开这些与计算机硬件、软件有关的因素，可以认为一个特定算法“运行工作量”的大小，只依赖于问题的规模(通常用整数 n 表示)，它是问题的规模函数。即算法的工作量 $=f(n)$ 例如，在 $N \times N$ 矩阵相乘的算法中，整个算法的执行时间与该基本操作(乘法)重复执行的次数 n^3 成正比，也就是时间复杂度为 n^3 ，即 $f(n)=O(n^3)$ 在有的情况下，算法中的基本操作重复执行的次数还随问题的输入数据集不同而不同。

例如在起泡排序的算法中，当要排序的数组 a 初始序列为自小至大有序时，基本操作的执行次数为 n ；当初始序列为自大至小有序时，基本操作的执行次数为 $n(n-1)/2$ 。对这类算法的分析，可以采用以下两种方法来分析。(1)平均性态(Average Behavior) 所谓平均性态是指各种特定输入下的基本运算次数的加权平均值来度量算法的工作量。设 x 是所有可能输入中的某个特定输入， $p(x)$ 是 x 出现的概率(即输入为 x 的概率)

， $t(x)$ 是算法在输入为 x 时所执行的基本运算次数，则算法的平均性态定义为 $\sum_{x \in D_n} p(x) \cdot t(x)$ 其中 D_n 表示当规模为 n 时，算法执行的所有可能输入的集合。(2)最坏情况复杂性(Worst-case Complexity) 所谓最坏情况分析，是指在规模为 n 时，算法所执行的基本运

算的最大次数。 2算法的空间复杂度 算法的空间复杂度是指执行这个算法所需要的内存空间。 一个算法所占用的存储空间包括算法程序所占的空间、输入的初始数据所占的存储空间以及算法执行中所需要的额外空间。 其中额外空间包括算法程序执行过程中的工作单元以及某种数据结构所需要的附加存储空间。 如果额外空间量相对于问题规模来说是常数，则称该算法是原地(in place)工作的。 在许多实际问题中，为了减少算法所占的存储空间，通常采用压缩存储技术，以便尽量减少不必要的额外空间。 100Test 下载频道开通，各类考试题目直接下载。 详细请访问 www.100test.com