

2011年计算机二级公共基础知识教程（1）PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/646/2021\\_2022\\_2011\\_E5\\_B9\\_B4\\_E8\\_AE\\_A1\\_c97\\_646470.htm](https://www.100test.com/kao_ti2020/646/2021_2022_2011_E5_B9_B4_E8_AE_A1_c97_646470.htm)

导读：算法是指解题方案的准确而完整的描述。即是一组严谨地定义运算顺序的规则，并且每一个规则都是有效的，且是明确的，没有二义性，同时该规则将在有限次运算后可终止。本课程的学习，要求认真看书，对书中的内容进行归纳和总结，将所有的知识穿成一条线。在看书的过程中，要仔细阅读，对书中重要的内容、概念要记住，因为本课程的考试是采用标准化的考试方式，单选和填空两种题型，因此要求考生对知识的掌握要准确，不能模棱两可。反复地看书，做题，因为本课程主要是一些理论的知识，要求记忆的内容很多，因此，必须多做题，多看书，在做题的过程中检验自己对知识的理解和掌握情况是否到位、正确。自己总结课程的内容，也是帮助理解和记忆的好方法。

### 第一章 数据结构与算法 一、学习目标与要求

1. 了解算法的基本概念和一些常用的算法，学会计算算法的时间复杂度；
2. 掌握数据结构的基本概念，并了解数据的逻辑结构和存储结构，学会利用图形的方式表示数据结构；
3. 了解线性表的基本概念，并掌握线性表的顺序存储结构以及顺序存储的线性表的基本运算；
4. 了解栈和队列的基本概念，并掌握它们的基本运算；
5. 了解线性链表的基本概念，并掌握线性链表的基本运算，同时，了解循环链表的基本概念和基本操作；
6. 理解树的概念，尤其是二叉树的基本概念和相关性质，掌握二叉树的存储结构和遍历技术；
7. 掌握查找技术，学会利用顺序查找和二分查找在数列中查

找指定的数据；8. 学会利用相关的排序技术实现无序数列的排序操作。

## 二、内容要点

### (一) 算法

#### 1. 算法的基本概念

算法是指解题方案的准确而完整的描述。即是一组严谨地定义运算顺序的规则，并且每一个规则都是有效的，且是明确的，没有二义性，同时该规则将在有限次运算后可终止。

#### 1) 算法的基本特征

(1) 可行性 由于算法的设计是为了在某一个特定的计算工具上解决某一个实际的问题而设计的，因此，它总是受到计算工具的限制，使执行产生偏差。如：计算机的数值有效位是有限的，当大数和小数进行运算时，往往会因为有效位数的影响而使小数丢失，因此，在算法设计时，应该考虑到这一点。

(2) 确定性 算法的设计必须是每一个步骤都有明确的定义，不允许有模糊的解释，也不能有多义性。例如，一个实际的问题，小宝和萍萍共有12个苹果，小宝比萍萍多4个，请问小宝和萍萍各有几个苹果？这个问题，我们可以立一个方程来求解，要求 $x$ 和 $y$ 的值，公式是正确的，但如何让计算能够进行计算，我们的算法不能把公式直接输进去，而应该设计出解题的步骤和过程。即设计的算法是计算工具所能够正常解决问题的过程。

(3) 有穷性 算法的有穷性，即在一定的时间内是能够完成的，即算法应该在计算有限个步骤后能够正常结束。例如，在数学中的无穷级数，在计算机中只能求有限项，即计算的过程是有穷的。

(4) 拥有足够的情报 算法的执行与输入的数据和提供的初始条件相关，不同的输入或初始条件会有不同的输出结果，提供准确的初始条件和数据，才能使算法正确执行。

#### 2) 算法的基本要素

一是数据对象的运算和操作，二是算法的控制结构。

(1) 算法中对数据的运算和操作 算法实际上是按解

题要求从环境能进行的所有操作中选择合适的操作所组成的一组指令序列。即算法是计算机所能够处理的操作所组成的指令序列。

(2) 算法的控制结构 算法的功能不仅取决于所选用的操作，而且还与各操作之间的顺序有关。在算法中，操作的执行顺序又称算法的控制结构，一般的算法控制结构有三种：顺序结构、选择结构和循环结构。在算法描述是，有相关的工具对这三种结构进行描述，常用的描述工具有：流程图、N-S结构图和算法描述语言等。

3) 算法设计的基本方法 为用计算机解决实际问题而设计的算法，即是计算机算法。通常的算法设计有如下几种：

(1) 列举法 列举法的基本思想是，根据提出的问题，列举出所有可能的情况，并用问题中给定的条件检验哪些是满足条件的，哪些是不满足条件的。列举法通常用于解决“是否存在”或“有哪些可能”等问题。例如，我国古代的趣味数学题：“百钱买百鸡”、“鸡兔同笼”等，均可采用列举法进行解决。使用列举法时，要对问题进行详细的分析，将与问题有关的知识条理化、完备化、系统化，从中找出规律。

(2) 归纳法 归纳法的基本思想是，通过列举少量的特殊情况，经过分析，最后找出一般的关系。归纳是一种抽象，即从特殊现象中找出一般规律。但由于在归纳法中不可能对所有的情况进行列举，因此，该方法得到的结论只是一种猜测，还需要进行证明。

(3) 递推 递推，即是从已知的初始条件出发，逐次推出所要求的各个中间环节和最后结果。其中初始条件或问题本身已经给定，或是通过对问题的分析与化简而确定。递推的本质也是一种归纳，递推关系式通常是归纳的结果。例如，斐波那契数列，是采用递推的方法解决问题的。

(4) 递归 在解决

一些复杂问题时，为了降低问题的复杂程度，通常是将问题逐层分解，最后归结为一些最简单的问题。这种将问题逐层分解的过程，并没有对问题进行求解，而只是当解决了最后的问题那些最简单的问题后，再沿着原来分解的逆过程逐步进行综合，这就是递归的方法。递归分为直接递归和间接递归两种方法。如果一个算法直接调用自己，称为直接递归调用；如果一个算法A调用另一个算法B，而算法B又调用算法A，则此种递归称为间接递归调用。

(5) 减半递推技术 减半递推即将问题的规模减半，然后，重复相同的递推操作。例如，一元二次方程的求解。

(6) 回溯法 有些实际的问题很难归纳出一组简单的递推公式或直观的求解步骤，也不能使用无限的列举。对于这类问题，只能采用试探的方法，通过对问题的分析，找出解决问题的线索，然后沿着这个线索进行试探，如果试探成功，就得到问题的解，如果不成功，再逐步回退，换别的路线进行试探。这种方法，即称为回溯法。如人工智能中的机器人下棋。

## 2. 算法复杂度

算法的复杂度包括时间复杂度和空间复杂度。

1) 时间复杂度 即实现该算法需要的计算工作量。算法的工作量用算法所执行的基本运算次数来计算。同一个问题规模下，如果算法执行所需要的基本次数取决于某一特定输入时，可以用以下两种方法来分析算法的工作量：

算法工作量 =  $f(n)$

(1) 平均性态 用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  $x$  是某个可能输入中的某个特定输入， $p(x)$  是  $x$  出现的概率， $t(x)$  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：

$D_n$  表示当规模为  $n$  时，算法执行时所有可能输入的集合。

(2) 最坏情况复杂度 指在规模为  $n$  时，

算法所执行的基本运算的最大次数。它定义为：公共基础知识笔记：2011年计算机二级公共基础知识辅导讲义汇总 2011年全国计算机等级考试二级公共基础知识要点汇总 2011年全国计算机等级考试二级公共基础知识总结汇总 公共基础知识试题：2011年全国计算机等级考试二级公共基础知识复习题 全国计算机等级二级公共基础知识考前练习汇总 全国计算机二级考试公共基础知识习题 全国计算机等级考试二级公共基础知识考前练习16套 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](http://www.100test.com)