

1. 算法不等于程序, 也不等于计算方法。程序的编制不可能优于算法的设计

2. 算法的特点: **可行性、确定性、有穷性、拥有足够的情报**

3. 算法复杂度主要包括**时间复杂度和空间复杂度**。

4. 时间复杂度指的是需要运算的次数

空间复杂度指的是运算所需要的内存空间

5. 数据结构包括数据的逻辑结构和数据的存储结构

6. 算法的执行效率与数据的存储结构有关

数据的存储结构有顺序、链接、索引等。

7. 数据结构中与所使用的计算机无关的是数据的逻辑结构

8. 数据结构应该包含的信息由: 数据元素的集合 d 和 数据的前后件关系 r

9. 数据结构中包括两种基本操作: 插入运算和删除运算

10. 根结点: 表示没有前件的结点 叶子结点: 表示没有后件的结点

11. 线性表表中的每一个数据元素, 除了第一个外, 有且只有一个前件, 除了最后一个外, 有且只有一个后件

12. 常见的线性结构有线性表、栈、队列和线性链表等。

13. 常见的非线性结构有树、二叉树和图等。

14. 线性表是一种存储结构, 它的存储方式: 顺序和链式

15. 栈是限定在一端进行插入与删除运算的线性表。

16. 栈是按照“先进后出”或“后进先出”的原则组织数据的

17. 队列是指允许在一端(队尾)进入插入, 而在另一端(队头)进行删除的线性表

18. 队列是“**先进先出**”或“**后进后出**”的线性表。
19. 线性链表：是一种物理存储单元上**非连续**、**非顺序**的存储结构，数据元素的逻辑顺序是通过链表中的指针链接来实现的
20. 每个结点由两部分组成：一部分用于存放数据元素的值，称为数据域；另一部分用于存放指针，称为指针域，用于指向该结点的前一个或后一个结点（即前件或后件）
21. 线性链表分为单链表、双向链表和循环链表三种类型。
22. 线性表的插入运算时需要移动元素，在等概率情况下，平均需要移动 $n/2$ 个元素。
23. 在树结构中，每一个结点只有一个前件，称为父结点。
24. 没有前件的结点只有一个，称为树的根结点，简称树的根。
25. 每一个结点可以有多个后件，称为该结点的子结点。没有后件的结点称为叶子结点。
26. 在树结构中，一个结点所拥有的后件的个数称为该结点的度，所有结点中最大的度称为树的度。
27. 树的最大层次称为树的深度。
28. 二叉树是一种很有用的非线性结构，它具有以下两个特点：1) 非空二叉树只有一个根结点；2) 每一个结点最多有两棵子树，且分别称为该结点的左子树与右子树。

二叉树的基本性质

性质 1 在二叉树的第 k 层上, 最多有 2^{k-1} ($k \geq 1$) 个结点。

性质 2 深度为 m 的二叉树最多有个 $2^m - 1$ 个结点。

$\log_2 n$ 性质 3 在任意一棵二叉树中, 度数为 0 的结点 (即叶子结点) 总比度为 2 的结点多一个。

性质 4 具有 n 个结点的二叉树, 其深度至少为 $\log_2 n + 1$, 其中 $\log_2 n$ 表示取 $\log_2 n$ 的整数部分。性质 5 具有 n 个结点的完全二叉树深度为 $\log_2 n$

29. 满二叉树: 每一层上的所有结点都有两个子结点。

30. 完全二叉树: 除最后一层外, 每一层上的结点数均达到最大值; 在最后一层上只缺少右边的若干结点。

31. 前序: 根左右 中序: 左根右 后序: 左右根

32. 顺序查找最坏的情况下需要比较 n 次顺序查找一个具有 n 个元素的线性表, 其平均复杂度为 $O(n)$ 。

33. 下列两种情况下只能采用顺序查找:

1) 如果线性表是无序表 (即表中的元素是无序的), 则不管是顺序存储结构还是链式存储结构, 都只能用顺序查找。

2) 即使是有序线性表, 如果采用链式存储结构, 也只能用顺序查找

34. 二分法查找最坏的情况下, 需要比较 $\log_2 n$ 次。

35. 二分法查找只适用于顺序存储的线性表, 且表中元素必须按关键字 有序 (升序), 进行二分法查找, 其时间复杂度为 $O(\log_2 n)$

36. 1、交换类排序法（方法：冒泡排序，快速排序）。

2、插入类排序法（方法：简单插入排序，希尔排序）。

3、选择类排序法（方法：简单选择排序，堆排序）。

类别	排序方法	时间复杂度
交换类	冒泡排序	$N(N-1)/2$
	快速排序	$O(n\log_2 n)$
插入类	简单插入排序	$N(N-1)/2$
	希尔排序	$O(n^{1.5})$
选择类	简单选择排序	$N(N-1)/2$
	堆排序	$O(n\log_2 n)$ 最小

37. 程序设计的风格主要强调：“清晰第一，效率第二”。

38. 结构化程序设计方法的主要原则可以概括为：自顶向下，逐步求精，模块化，限制使用 goto 语句。

39. 结构化程序的基本结构：顺序结构，选择结构，重复结构。

40. 面向对象方法的主要优点：（1）与人类习惯的思维方法一致；（2）稳定性好；（3）可重用性好；（4）易于开发大型软件产品；（5）可维护性好。

41. 面向对象的程序设计主要考虑的是提高软件的可重用性。

42. 一个对象由对象名、属性和操作三部分组成。

43. 对象的基本特点：标识惟一性，分类性，多态性，封装性，模块独立性好。

44. 信息隐蔽是通过对象的封装性来实现的。

45. 计算机软件是包括程序、数据及相关文档的完整集合。

46. 软件工程源自软件危机

47. 软件危机可以归结为成本、质量、生产率等问题。

48. 软件工程三要素：方法、工具和过程。

49. 软件生命周期分为软件定义、软件开发及软件运行维护三个阶段

50. 软件定义阶段：包括制定计划和需求分析。

51. 软件开发阶段：

软件设计：分为概要设计和详细设计两个部分。

软件实现：把软件设计转换成计算机可以接受的程序代码。

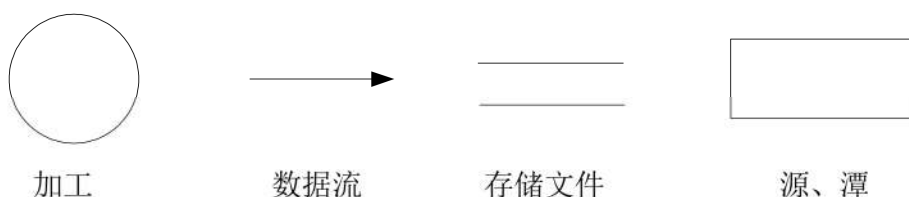
软件测试：发现软件存在的错误。

52. 软件运行维护阶段：软件投入运行，并在使用中不断地维护，进行必要的扩充和删改。

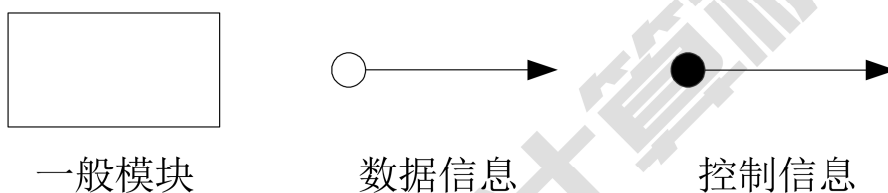
53. 软件生命周期中所花费最多的阶段是软件运行维护阶段。

54. 结构化分析的常用工具：1) 数据流图 (DFD); 2) 数据字典 (DD); 3) 判定树; 4) 判定表 (需求分析)

55. 画数据流图的基本步骤：自外向内，自顶向下，逐层细化，完善求精 下图为数据流图的基本元素



56. 数据字典的作用是对数据流图中出现的被命名的图形元素的确切解释。
57. 数据字典是结构化分析方法的核心。
58. 模块分解的主要指导思想是信息隐蔽和模块独立性
59. 模块的耦合性和内聚性是衡量软件的模块独立性的两个定性指标
60. 一个设计良好的软件系统应具有高内聚、低耦合的特征。
61. 常用的软件结构设计工具是结构图，也称程序结构图。程序结构图的基本图符：



62. 数据流的类型：大体可以分为两种类型，变换型和事务型。
63. 常用的过程设计（即详细设计）工具有以下几种：
- 图形工具：程序流程图、N-S（方盒图）、PAD（问题分析图）和 HIPO（层次图+输入/处理/输出图）。
- 表格工具：判定表。
- 语言工具：PDL（伪码）

64. 软件测试的关键是设计测试用例，一个好的测试用例能找到迄今为止尚未发现的错误

65. 软件测试方法：静态测试和动态测试。

静态测试: 包括代码检查、静态结构分析、代码质量度量。不实际运行软件, 主要通过人工进行。

动态测试: 是基于计算机的测试, 主要包括白盒测试方法和黑盒测试方法。

66. 白盒测试方法也称为**结构测试或逻辑驱动测试**。

67. 白盒测试法的测试用例是根据程序的内部逻辑来设计的, 主要用软件的单元测试, 主要方法有逻辑覆盖、基本路径测试等。

68. 黑盒测试方法也称为**功能测试或数据驱动测试**。黑盒测试是对软件已经实现的功能**是否满足需求进行测试和验证**。

69. 黑盒测试主要方法有等价类划分法、边界值分析法、错误推测法等, 主要用软件的确认测试。

70. 软件测试过程一般按 4 个步骤进行: 单元测试、集成测试、确认测试和系统测试。

71. 单元测试的内容包括: **模块接口测试**、**局部数据结构测试**、**错误处理测试**和**边界测试**。

72. 程序调试的任务是**诊断和改正**程序中的错误, 主要在开发阶段进行, 调试程序应该由编制源程序的程序员来完成。

73. 程序调试的基本步骤: (1) 错误定位; (2) 纠正错误; (3) 回归测试。

74. **软件的调试后要进行回归测试, 防止引进新的错误。**

75. 原因排除法、回溯法、强行排错法

76. E-R 模型的图示法:

加小黑老师 QQ: 82375141, 让你顺利通关计算机二级!

- 1) 实体集: 用矩形表示。
- 2) 属性: 用椭圆形表示。
- 3) 联系: 用菱形表示。
- 4) 实体集与属性间的联接关系: 用无向线段表示。
- 5) 实体集与联系间的联接关系: 用无向线段表示。



微信扫码参加计算机二级考前直播

直播总时长 **12 小时**, 全程免费, 干货满满!

还有 **3000 元** 奖学金等你来拿!!!

一定要推荐给身边考二级的同学!
小黑课堂团队感谢你的支持!

关注公众号**小黑课堂计算机二级**, 领取保过秘籍手册!
