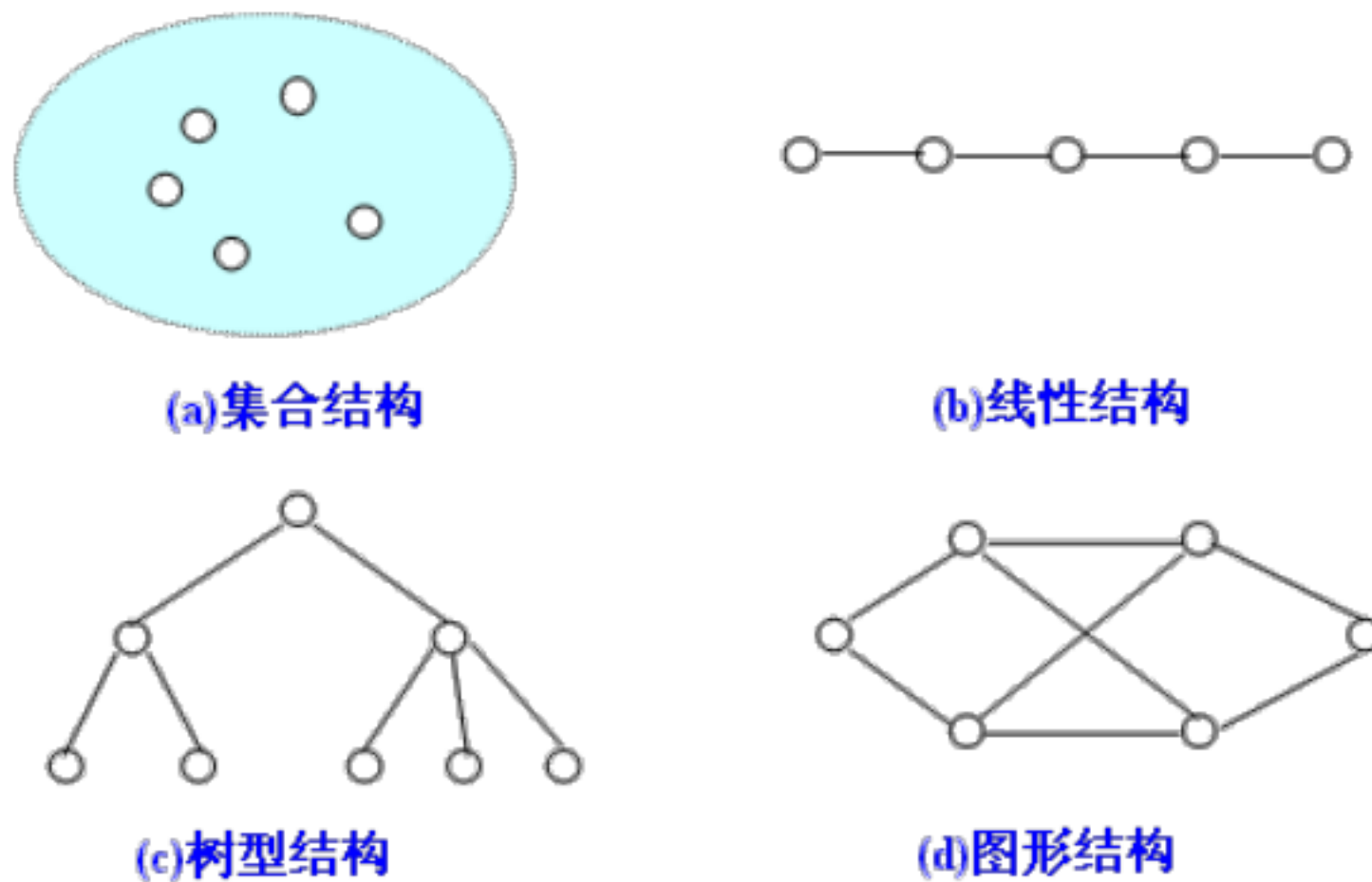


数据结构与算法

第一节 数据结构及算法概述

一、数据结构



图、 四类基本结构的示意图

【要点】

1. 数据元素是数据的基本单位。
2. 数据结构是相互之间存在一种或多种特定关系的数据元素的集合。
- 3.4 类基本的逻辑结构：集合、线性结构、树形结构和网状结构。
- 4.4 种数据存储方式：顺序、链式、索引和散列。

【例题·单选题】（ 2009 年 × 省信用社招聘考试真题 ） 下列说法不正确的是（ ）。

- A. 数据元素是数据的基本单位 B. 数据项是数据中不可分割的最小标志单位
- C. 数据可由若干个数据元素构成 D. 数据项可由若干个数据元素构成

『正确答案』 D

『答案解析』 数据元素是数据的基本单位， 在计算机程序中通常被作为一个整体进行考虑和处理。一个数据元素可由若干个数据项组成。数据项是不可分割的、含有独立意义的最小数据单位。因此 D 选项不正确。

二、算法

算法是对特定问题求解步骤的一种描述，它是指令的有限序列，其中每条指令表示一个或多个操作。

算法的特性：有穷性、确定性、可行性、输入和输出。

【要点】

评价算法优劣标准：正确性、可读性、健壮性、高效率与低存储量需求。

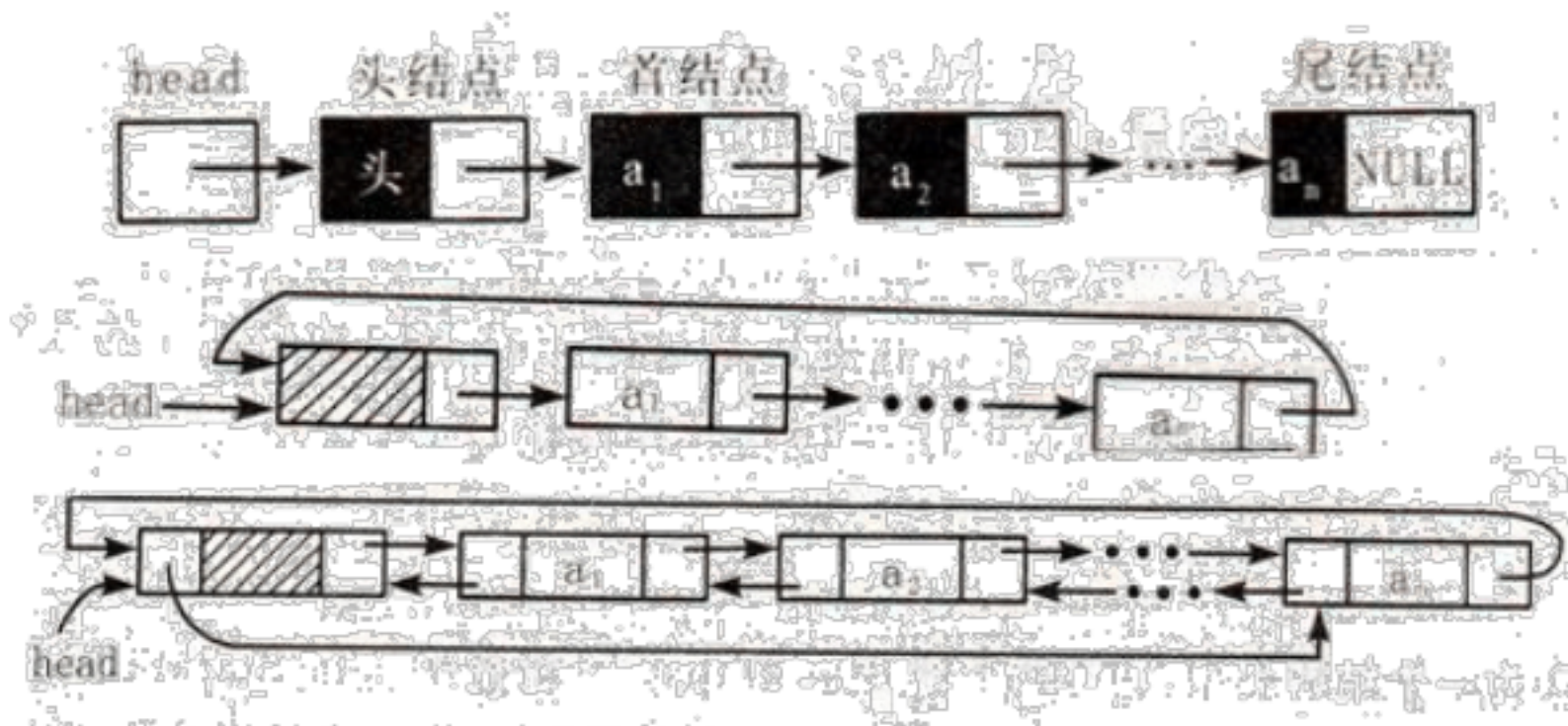
第二节 线性表

线性表是 n ($n \geq 0$) 个数据元素 a_1, a_2, \dots, a_n 组成的有限序列， $n=0$ 时称为空表。

非空的线性表，有以下特征：

1. 有且仅有一个开始结点 a_1 ，没有直接前趋，有且仅有一个直接后继 a_2 。
2. 有且仅有一个终结结点 a_n ，没有直接后继，有且仅有一个直接前趋 a_{n-1} 。
3. 其余的内部结点 a_i ($2 \leq i \leq n-1$) 都有且仅有一个直接前趋 a_{i-1} 和一个直接后继 a_{i+1} 。

线性表的链式存储包括单链表、循环链表和双链表。



【注意】

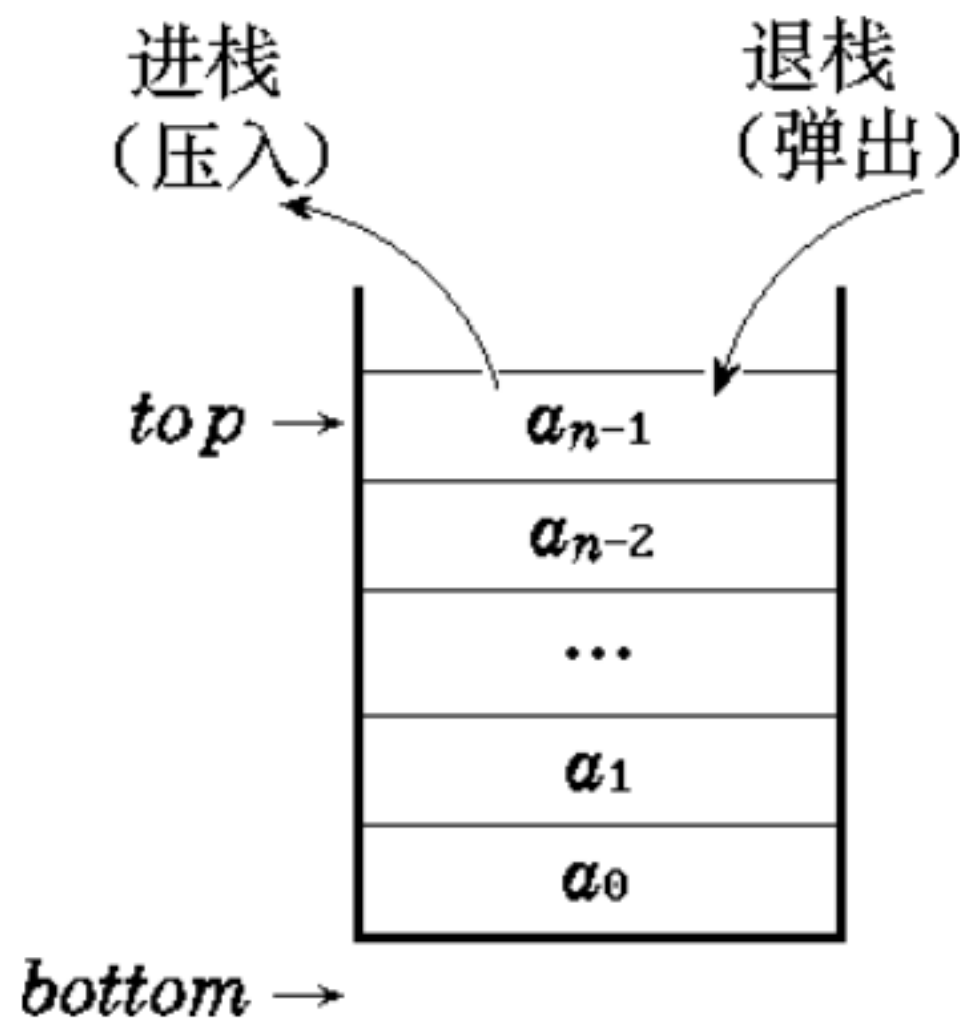
与单链表的插入和删除操作不同的是，在双链表中插入和删除须同时修改两个方向上的指针。

第三节 栈和队列

一、栈

栈是一种“特殊的”线性表，这种线性表中的插入和删除运算限定在表的某一端进行。不含任何数据元素的栈称为空栈。

栈的基本运算：构造一个空栈 $\text{InitStack} (S)$ 、判栈空 $\text{StackEmpty} (S)$ 、判栈满 $\text{StackFull} (S)$ 、进栈 $\text{Push}(S, x)$ 、退栈 $\text{Pop}(S)$ 、取栈顶元素 $\text{StackTop}(S)$ 。



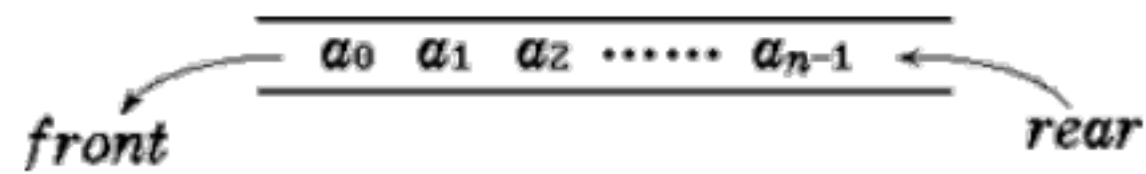
【要点】

栈的修改是按后进先出的原则进行。

二、队列

队列是只允许在一端进行插入，而在另一端进行删除的运算受限的线性表。当队列中没有元素时称为空队列。

栈和队列一般有两种存储结构：顺序存储结构和链式存储结构。



【要点】

队列的修改是按照先进先出的原则进行的。

【例题·单选题】（ 2007 年 × 省信用社招聘考试真题 ） 以下（ ）不是栈的基本运算？

- A. 删除栈顶元素 B. 删除栈底元素 C. 判断栈是否为空 D. 将栈置为空栈

『正确答案』 B

『答案解析』 栈是一种特殊的线性表，这种线性表上的插入和删除运算限定在表的某一端进行。允许进行插入和删除的这一端称为栈顶，另一端称为栈底。处于栈顶位置

的数据元素称为栈顶元素。栈的修改是按后进先出的原则进行。栈的运算在栈顶进行。因此，删除栈底元素不属于栈的基本运算。

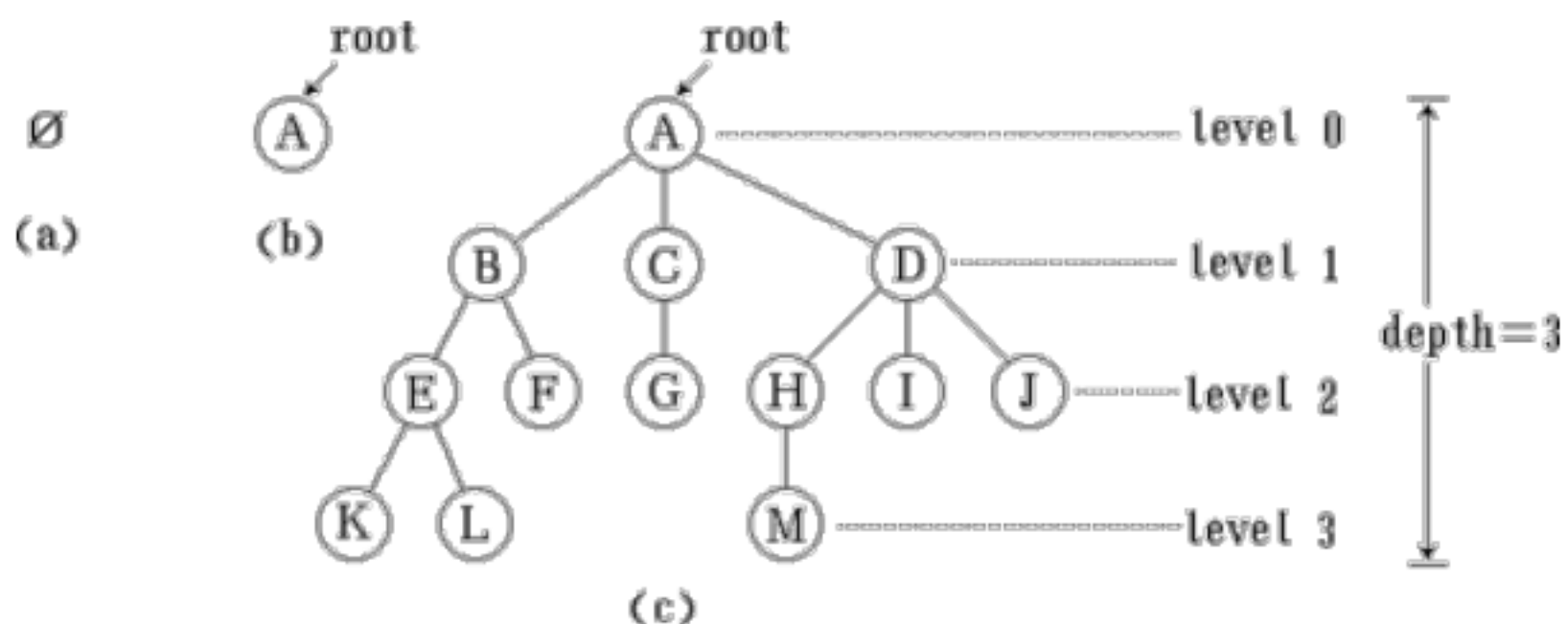
第四节 数组

数组：把具有相同类型的若干变量按有序的形式组织起来的同类数据元素的集合。

第五节 树

一、树

(一) 基本概念



树是 n ($n > 0$) 个结点的有穷集合，满足以下条件：

- (1) 有且仅有一个称为根的结点。
- (2) 其余结点分为 m ($m \geq 0$) 个互不相交的非空集合 T_1, T_2, \dots, T_m , 这些集合中的每一个都是一棵树，称为根的子树。

森林是 m ($m \geq 0$) 棵互不相交的树的集合。

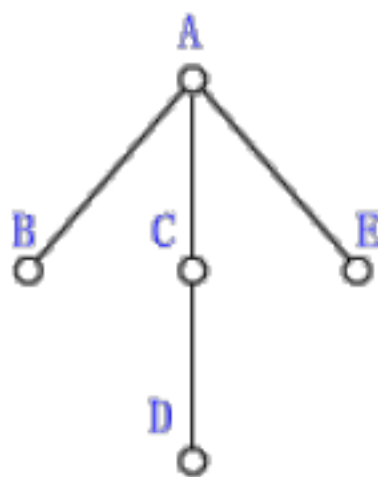
树中的一个结点拥有的子树数称为该结点的度。一棵树的度是指该树中结点的最大度数。

树中某个结点的子树的根称为该结点的孩子或儿子。

树中结点的最大层数称为树的高度或深度。

(二) 树和森林的遍历

1. 树的遍历



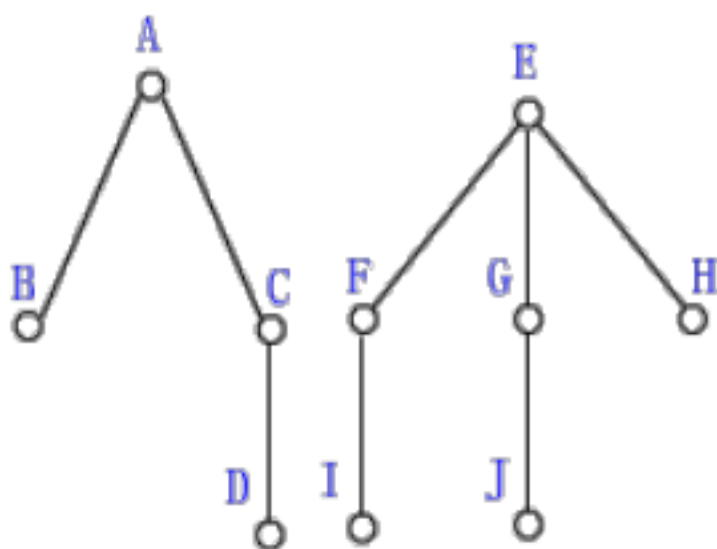
(a) 树

先序遍历：若树不空，则先访问根结点，然后依次先序遍历各棵子树。 ABCDE

后序遍历：若树不空，则先依次后序遍历各棵子树，然后访问根结点。 BDCEA

层次遍历：若树不空，则自上而下自左至右访问树中每个结点。 ABCED

2. 森林的遍历



(a) 森林

前序遍历森林：

若森林非空，则：

访问森林中第一棵树的根结点。

前序遍历第一棵树中根结点的各子树所构成的森林。

前序遍历除第一棵树外其他树构成的森林。

右图前序遍历序列为 ABCDEFIJGH

后序遍历森林：

若森林非空，则：

后序遍历森林中第一棵树的根结点的各子树所构成的森林。

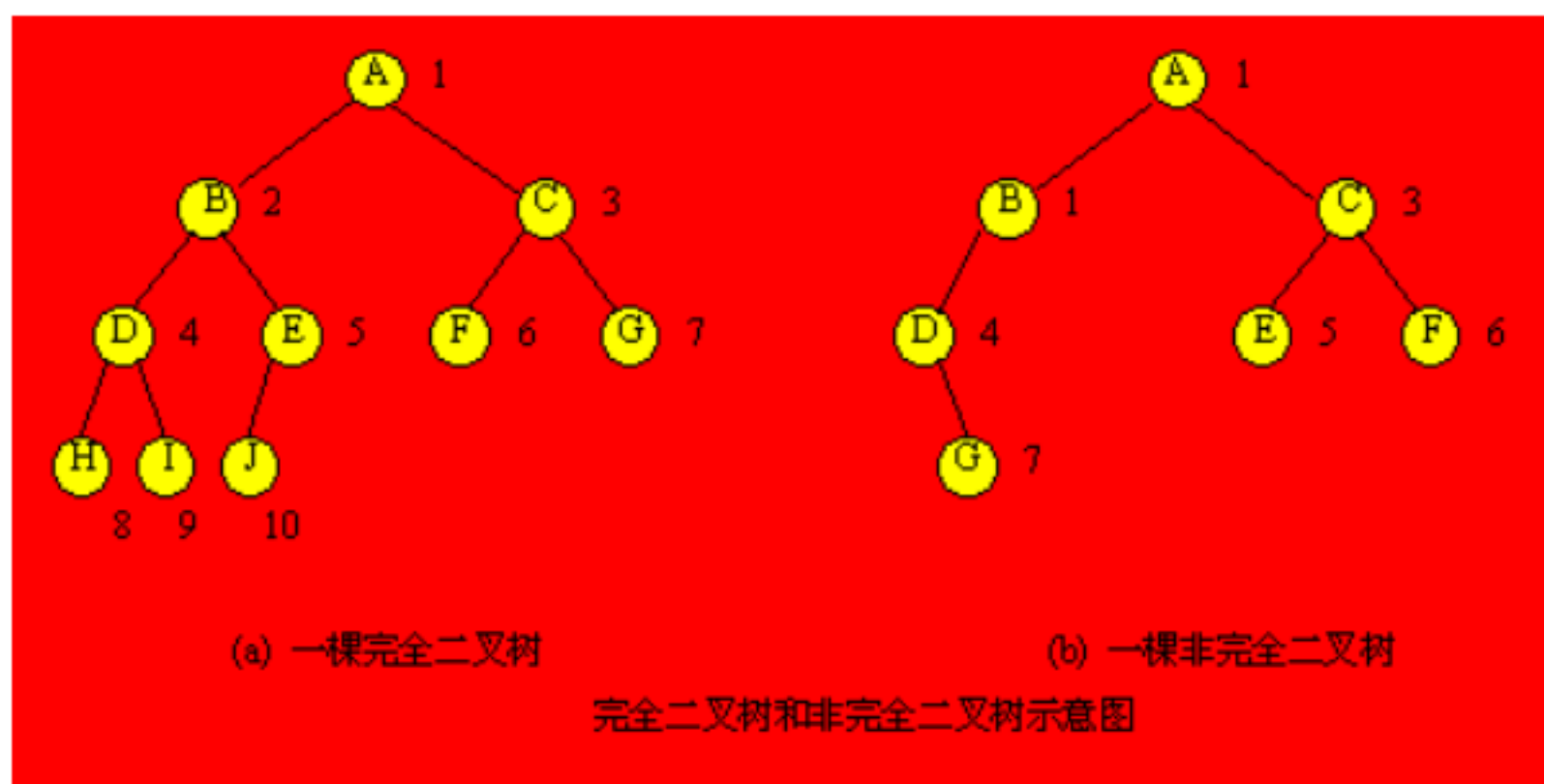
访问第一棵树的根结点。

后序遍历除第一棵树外其他树构成的森林。

右图后序遍历序列为 BDCAIFJGHE

二、二叉树

(一) 基本概念



二叉树是 n ($n \geq 0$) 个结点的有限集，它可以是空集 ($n=0$)，或者由一个根结点及两棵互不相交的、分别称作这个根的左子树和右子树的二叉树组成。

二叉树的性质

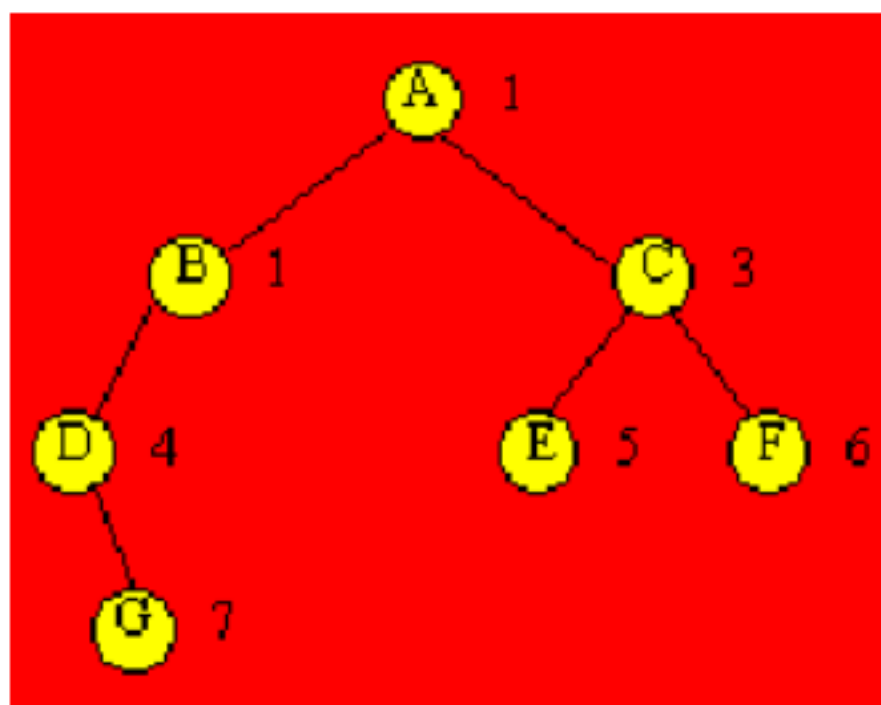
1. 二叉树的第 i 层上至多含有 2^{i-1} 个结点 ($i \geq 1$)。
2. 深度为 k 的二叉树上至多含有 $2^k - 1$ 个结点 ($k \geq 1$)。
3. 对任何一棵二叉树，若它含有 n_0 个叶子结点、 n_2 个度为 2 的结点，则必存在关系式： $n_0 = n_2 + 1$ 。

深度为 k ($k \geq 1$) 且有 $2^k - 1$ 个结点的二叉树称为满二叉树。

如果在一棵深度为 k ($k \geq 1$) 的满二叉树上删去第 k 层上最右边的连续 j ($0 \leq j < 2^{k-1}$) 个结点，就得到一棵深度为 k 的完全二叉树。满二叉树也是完全二叉树。

(二) 二叉树的遍历

1. 先根遍历



若需遍历的二叉树为空，执行空操作；否则，依次执行下列操作：

- (1) 访问根结点。
- (2) 先根遍历左子树。
- (3) 先根遍历右子树。

右图的遍历序列为： ABDGCEF

2. 中根遍历

若需遍历的二叉树为空，执行空操作；否则，依次执行下列操作：

- (1) 中根遍历左子树。
- (2) 访问根结点。
- (3) 中根遍历右子树。

右图的遍历序列为： DGBAECF

3. 后根遍历

若需遍历的二叉树为空，执行空操作；否则，依次执行下列操作：

- (1) 后根遍历左子树。
- (2) 后根遍历右子树。
- (3) 访问根结点。

右图的遍历序列为： GDBEFCA

【要点】

树和二叉树的概念和他们的各种遍历方法。

第六节 图

一、图的基本概念

在树形结构中，节点间具有分支层次关系，每一层上的节点只能和上一层中的至多一个节点相关，但可能和下一层的多个节点相关。

在图状结构中，任意两个节点之间都可能相关。

有序偶对用尖括号括起来；无序偶对用圆括号括起来。 (x, y) 与 (y, x) 被认为是同一条边，但 $\langle x, y \rangle$ 与 $\langle y, x \rangle$ 则是不同的两条弧。

图的边或弧附带数值，这种数值叫权。

【要点】

一个具有 n 个顶点的无向完全图的边数为 $n(n-1)/2$ 。一个具有 n 个顶点的有向完全图的弧数为 $n(n-1)$ 。

二、图的存储结构

图的两种存储结构：邻接矩阵表示法和邻接表表示法。

图的遍历包括深度优先遍历和广度优先遍历两种方法。

第七节 查找

一、查找的基本概念

查找表是由同一类型的数据元素（或记录）构成的集合。分为两类：静态查找表和动态查找表。

二、线性表的查找

顺序查找：从表的一端开始，顺序扫描线性表，依次将扫描到的结点关键字和给定值 K 相比较。若当前扫描到的结点关键字与 K 相等，则查找成功；若扫描结束后，仍未找到关键字等于 K 的结点，则查找失败。

二分查找：要求线性表是有序表，每次将查找区间中间位置上的数据元素的键值与给定值 K 比较，若不等则缩小查找区间，并在新的区间内重复上述过程，直到查找成功或查找区间长度为 0（查找不成功）为止。

【要点】

二分查找法要求线性表是有序表，掌握其查找方法。

【例题·单选题】（2010年×省信用社招聘考试真题）在顺序表（3, 6, 8, 10, 12, 15, 16, 18, 21, 25, 30）中，用二分法查找关键码值 11 所需的关键码比较次数为（ ）。

A.2 B.3 C.4 D.5

『正确答案』 D

『答案解析』根据二分法查找方法，各数值比较次数如下（用圆圈里的数来表示：3, 6, 8, 10, 12, 15, 16, 18, 21, 25, 30），对于 11 来说，是查找不成功，所以应该比较 5 次。

第八节 排序

一、排序概述

排序，就是使文件中的记录按关键字递增（或递减）次序排列起来。

按是否访问外存，可将排序方法分为内部排序和外部排序。

对于内部排序，根据设置有序序列的方式的不同，又可以划分为插入排序、交换排序、选择排序、归并排序和其他排序方法。

二、常用排序方法简介

（一）插入排序

每步将一个待排序的记录按其关键字的大小插到前面已经排序的序列中的适当位置，直到全部记录插入完毕为止。

1. 直接插入排序

依次将每个记录插入到一个有序序列中去。

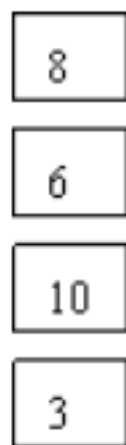
2. 希尔排序

先取一个小于 n 的整数 d_1 作为第一个增量，把文件的全部记录分成 d_1 个组。所有距离为 d_1 的倍数的记录放在同一个组中。先在各组内进行直接插入排序；然后，取第二个增量 $d_2 < d_1$ 重复上述的分组和排序，直至所取的增量 $d_t = 1$ ($d_t < d_{t-1} < \dots < d_2 < d_1$)，即所有记录放在同一组中进行直接插入排序为止。

（二）交换排序

两两比较待排序记录的关键字，发现两个记录的次序相反时即进行交换，直到没有反序的记录为止。

1. 冒泡排序



对于 $1 \sim n$ 个记录，先将第 n 个和第 $n-1$ 个记录的键值进行比较，如 $r[n].key < r[n-1].key$ ，则将两个记录交换。然后比较第 $n-1$ 个和第 $n-2$ 个记录的关键字。依此类推，直到第 2 个记录和第 1 个记录进行比较交换，这称为一趟冒泡。这一趟最明显的效果是：将键值最小的记录传到了第 1 位。然后对 $2 \sim n$ 个记录进行同样操作，则具有次小键值的记录被安置在第 2 位上。重复以上过程，每次的移动都向最终排序的目标前进，直至没有记录需要交换为止。

2. 快速排序

在待排序的 n 个记录中任取一个记录（例如就取第 1 个记录），以该记录的键值为标准，将所有记录分为两组（一般都是无序的），使得第 1 组中各记录的键值均小于或等于该键值，第 2 组中各记录的键值均大于该键值。然后把该记录排在这两组的中间（这也是该记录最终的位置）。此称为一趟快速排序（或一次划分），对所分成的两组再分别重复上述方法，直到所有记录都排在适当位置为止。

【要点】

冒泡排序和快速排序方法。

【例题·单选题】（2010 年 \times 省信用社招聘考试真题）用某种排序方法对序列（25，84，21，47，15，27，68，35，20）进行排序，记录序列的变化情况如下：

原始序列：258421471527683520

第一趟排序结果：201521254727683584

第二趟排序结果：152021253527476884

第三趟排序结果：152021252735476884

则采用的排序方法是（ ）。

A. 直接选择排序 B. 冒泡排序 C. 快速排序 D. 二路归并排序

『正确答案』 C

『答案解析』直接选择排序第一趟就应该选出一个最小的在第一个位置；冒泡排序法是两两比较，25 与 84 比较，25 要小，位置不变；快速排序法使比基准数（第一趟是 25）小的数在基准数左边，比基准数大的数在基准数右边；二路归并排序是分组比较，第一趟应该是“2584”、“2147”、“1527”、“6835”、“20”这几组进行排序。

（三）选择排序

每一趟从待排序的记录中选出关键字最小的记录，顺序放在已排好序的子文件的最后，直到全部记录排序完毕。常用的选择排序方法有直接选择排序和堆排序。

1. 直接选择排序

先在所有的记录中选出键值最小的记录，把它与第 1 个记录交换；然后在其余的记录中再选出键值最小的记录与第 2 个记录交换；依此类推，直至所有记录排序完成。

2. 堆排序

对一组待排序记录的键值，先将它们按堆定义排成一个堆，这称为建堆。这就找到最小的键值。然后将最小键值取出，用剩下的键值再建堆，便得到次最小的键值。如此反复进行，直到得到最大键值，从而将全部键值排好序为止。

（四）归并排序

将这些有序的子序列进行合并，从而得到有序的序列。合并是一种常见运算，其方法是：比较各子序列的第一个记录的键值，最小的一个就是排序后序列的第一个记录的键值。取出这个记录，继续比较各子序列现在的第一个记录的键值，便可找出排序后的第二记录。如此继续下去，最终可以得到排序结果。