Foxbase 索引文件分析和C语言的模拟实现

黄曙荣　　倪明生
（盐城工专基础部　盐城市物资局信息中心，盐城，224003）

Foxbase 是目前微机上使用非常广泛的关系型数据库管理系统，其运行速度快，使用方便，用户界面友好，受到客户的欢迎。其索引文件的组织和维护优于其它微机数据库管理系统。

1　数据库文件结构

Foxbase 的数据库包含了数据字典和数据两部分。具体结构为：

1.1　**数据字典**　文件开始的32字节为数据字典的描述部分。第1字节为03H，为数据库文件的标识；2—4字节分别为数据库文件的最后更新的日期；5—8字节为该数据库文件中包含的记录数；9—10字节为文件中数据字典的长度；10—11字节为每一记录（定长度记录）的长度，而其后的部分每32字节描述一个字段：头10字节为字段名，第12字节为字段类型，第13字节表示该字段在整条记录中的偏移，第一字段偏移为01H，是由于每条记录第一个字节为20H（空格），它既是记录间的分隔符，又是记录是否被删除的标识位（*表示删除）。第17字节为字段的域宽，第18字节为小数的位数，由于域宽用一个字节表示，故其长度不能超过254。上述结构与dBASE Ⅲ的数据库文件结构兼容。

1.2　**数据部分**　紧接着数据字典之后的就是数据部分，每条记录按输入的顺序存放，根据记录号就可以找到记录的物理存储位置，从而找到记录。Foxbase 的索引文件正是将记录号作为索引指针建立索引文件，而没有使用具体的物理偏移，因而简洁明了。

2　索引策略

Foxbase 的索引文件是采用改进的B+树索引策略，它可迅速进行随机和顺序查找，文件的建立和维护也很方便，是一种比较高级的索引策略。一个m阶B+树的定义为：(1)每个节点至多有m个儿子；(2)每个非叶节点（根节点除外）必须有>=(m+1)/2个儿子；(3)根节点至少有两个儿子（除非它本身也为一个叶）；(4)有k个儿子的非叶节点有k个关键字，分别对应于其子树根节点最大关键字；(5)所有的叶节点都在同一层上，并包含了全部关键字信息及相应记录指针，节点内关键字按升序（或降序）排列，各节点按关键字顺序连接。

在Foxbase中，被索引文件的全部记录信息均放在叶节点上，该节点中的最大关键字被复制到上面的层中，而把上面的层作为索引，并且叶节点的记录是按关键字递增（或递减）的顺序存放。实际上是一种多叉平衡树形，因而查找特别方便，插入或删除的平衡算法也比较简单，是组织索引文件的有效结构。

3　存储结构

Foxbase 索引文件的后缀为.IDX，在磁盘上的存贮是按B+树的策略进行组织的，它通常

?1994-2014 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

以微机磁盘存取的基本单位一个扇区作为存贮节点，其大小一般为 512Bytes，这样不仅存取方便，而且可有效地进行磁盘缓冲区管理及索引文件内部节点的物理位置的定位。

由于节点大小固定，因而根据索引文件索引关键字的长度不同，其索引文件所对应的 B+树的阶数也不一样，从而直接影响到索引文件的效率，其阶数 m=(512−12)/(l+4)，式中 512 为节点的大小，12 为节点控制信息的长度，I 为关键字的长度，4 为 4Bytes 的长整型数指针(在叶节点中表现为被索引文件的记录号，在中间节点中表现为层间链接的指针)，显然 I 越大，阶数 m 越小，其存取的效率也越低。

根据索引关键字的类型，Foxbase 索引文件有两种稍有差异的存贮方式。

3.1　**数值型关键字**　用 8 字节(64 位)的双精度规格化浮点数进行存贮以及其它相应处理，其阶数是固定的(512−12)/(8+4)=41，浮点数的存贮方式类似于 IEEE 的标准格式，最高位第 63 位为尾数符号位，第 52−62 位为用移码表示的基为 2 的阶码，第 0−51 位为尾数。

3.2　**字符串型关键字**　按关键字的实际长度给出，其阶数是可变的。

Foxbase 索引文件在磁盘上的存贮分为两个部分。开始的第一个扇区为索引文件的控制节点；其余部分为索引文件的索引信息部分。具体的细节是：

(1)主控节点　长度为 512Bytes，00−03 字节指明根节点在索引文件中位置，是相对于文件头的文件位移指针，用字节数表示；04−07 字节为 FF FF FF FF，实际上是一空指针；08−0BH4 字节的长整型数表示整个索引文件的大小(包括主控节点)；0C−0D 为关键字长度；从 10H 往后为关键字名称的描述：例如 key1+key2，表示索引关键字由对应数据库字段 key1 和 key2 组成，如为数字型则为两者之和，字符串型则为 key1 和 key2 的连接。

(2)文件索引节点　长度也为 512 字节，是索引文件的主体。00 字节为该节点性质的标识；01 表示该节点为根节点，02 表示该节点为叶节点，00 则为中间节点，03 表示根叶组合节点，即当被索引文件的记录数较少，一个索引节点就可以放进全部索引信息时，将全部信息放在根节点中，用 03 表示这种情况；02−03 字节表示该节点中的索引关键项的个数；第 04−07 字节表示与该节点同层的左相邻节点的指针，用该节点相对于文件起始位置的字节数表示，第 08−0BH 则表示右相邻节点的指针，若为 FF FF FF FF，则表示该节点左(或右)已没有其它节点；从 0CH 开始为索引项的排列。索引项由索引关键字+4 字节的指针组成，在中间层该指针表示层间索引指针，在叶节点则表示记录号。

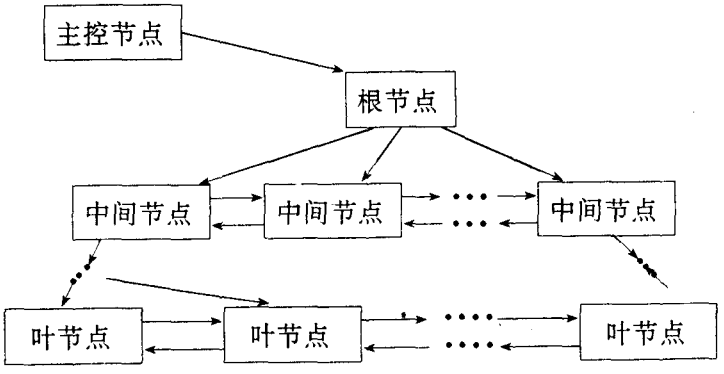

图 1　整个存贮结构简图

整个存贮结构可用图 1 表示。由图 1 可以看出，由于是多叉平衡树形，在进行随机查找时，最多进行(H+1)(H 为树形高度)次磁盘操作，又由于中间节点和叶节点之间有指针左右勾连，可以很方便地进行正反向的顺序查找，是较为有效的索引策略。

?1994-2014 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

4 随机查找

B+树形是二叉排序树的自然推广，其查找算法很简单：要查找关键字为key的数据库记录是否存在，可以先将控制节点读入内存，然后再根据根节点的位移将根节点读入内存，从根节点开始内部查找。若大于根中所有索引项的关键字，则查找失败；否则沿合适的层间指针读入层间的内部节点，一直找到合适的叶节点，进行内查找判断是否存在。

5 顺序查找

由于叶节点间有指针互相勾连，因而在读入任一叶节点后，可很方便地读入相邻叶节点进行正反向的顺序查找。

6 索引文件的建立和插入

索引文件的建立包括建立主控节点和索引节点(即B+树树形)。即①建立主控节点和根节点；②反复插入索引关键项，直到所有索引项插入完为止。

每次插入分两步完成：其一是，沿着树形查找到要插入的叶节点；其二是，若该叶节点未满，则直接插入即可；若该节点已满，则进行分裂：生成一新节点，并复制一半索引项到新节点中，将该索引项插入合适的位置(可能在新节点中，也有可能在原节点中)，修改上层父亲节点对应的索引项的指针指向新节点，原指针和原叶节点中余下的索引项中的最大关键字组成新的索引项在上层中进行新的插入，并可能引起新的分裂。在极端的情况下，分裂可一直延续到根节点，从而导致树形的高度增加，这是B+树长高的唯一方式。

7 C语言的实现

按上面分析，可以比较容易给出具体算法的实现。笔者用TurboC2.0编写了随机查找和插入以及初始化索引文件全部的代码，其建立的索引文件和Foxbase2.0兼容。由于是对树形进行操作，采用递归算法更加简明，但不易控制，在运行中时间和空间的开销也较大，因而笔者在程序中采用了堆栈的数据结构来实现，用堆栈来保存树形查找的路径，清晰易读。

(上接第20页)

对于某些比较复杂的电路，可把电路中的某一部分电路整体调换，使要化简的接点移到一条公共线上进行化简。例如图4(a)电路，可把电路分成4组。为了化简合并接点K_2，先把其中一组H_0～H_3与接点K_1、K_2组成的部分电路整体倒接，即将接于电源负极的H_0～H_3接到$\overline{K_3}$上，而把接在$\overline{K_3}$接点的K_2接点换接到电源负极上，再把其它3组：H_4～H_7、H_8～H_{11}、H_{12}～H_{15}以同样的方法整体倒接。这样使全部K_2接点移接到电源负极上，再把相同的接点合并，电路合并后引起的寄生回路用二极管隔离。

二极管在电路中的主要作用是防止产生寄生回路，所以一般可选用1N4001～1N4007型或其它型号的二极管，只要满足负荷的正向电流和反向电压即可。但如果电路中使用发光二极管时，电路中的二极管一般不能省去。否则，由于发光二极管反向电流较大，仍能产生寄生回路，使不应该亮的发光二极管亮。

?1994-2014 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

由于篇幅所限，图 2 仅给出三个过程的程序流程图：(1)随机查找关键字为 key 的记录；(2)插入键值为 key，记录号为 no 的索引项；(3)建立索引文件。

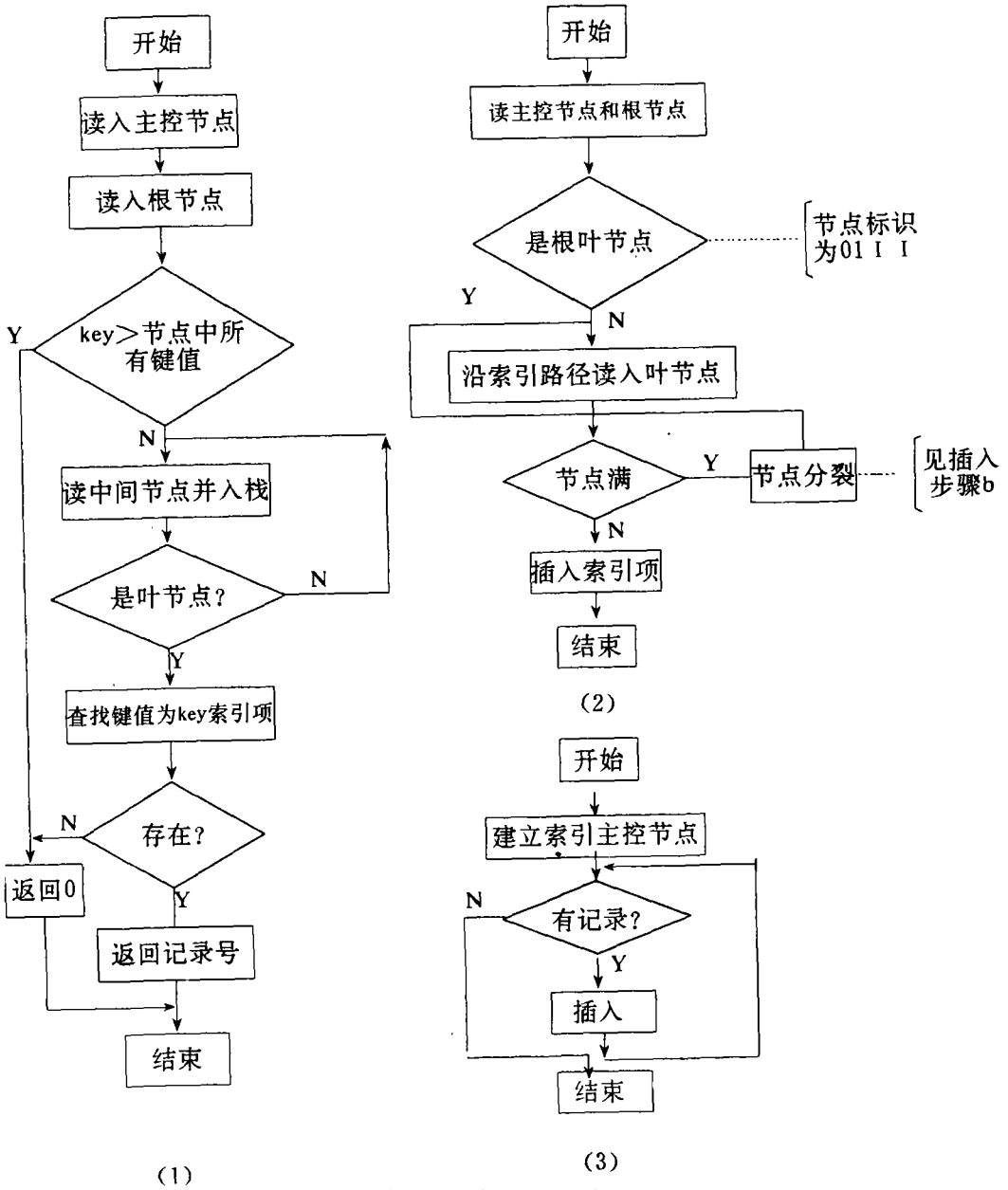

图 2　三个过程的程序流程图

参考文献

1　N. 沃思. 算法＋数据结构＝程序(Algorithms＋Data＝Programs).

2　管文蔚等. 数据结构

3　威廉. 默里等. 80286/80386 汇编语言程序设计

?1994-2014 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net