实验1 数据和元数据的组织

一、实验目的

1.理解逻辑文件结构和物理文件结构；

2.理解逻辑数据文件读写模块和物理数据文件读写模块；

二、实验内容

1.根据数据和元数据存储要求，选择合适的数据结构组织数据与元数据，设计合理的物理数据文件的结构存储数据和元数据。

2.设计具有基本功能的物理数据文件读写模块和逻辑数据文件读写模块。

三、实验要求

1. 设计一个物理文件格式，满足以下要求：

* 保存数据表的记录信息（数据内容）；
* 以数据表形式保存元数据信息（表名，表的字段名称、类型、长度等信息）；

1. 逻辑数据读写以记录为单位，每次读写一记录或多记录；
2. 物理数据读写以页为单位，每次读写一页或多页。自定义页面大小。
3. 逻辑数据读写模块需实现以下函数：

* AddRecord：增加记录；
* DeleteRecord：删除记录
* UpdateRecord：修改符合指定条件记录的某个字段的值；
* FindRecord：查询符合制定条件记录；

四、参考资料

DBF文件格式说明

DBF文件是一种以二进制进行存储的表格数据文件，其文件内部有着严格的格式要求，具体由文件头和记录项组成。其中文件头中包括字段的相关信息。DBF文件的数据结构如下表所示：

|  |  |  |  |
| --- | --- | --- | --- |
| 组成 | 内容 | 位置（Byte） | 说明 |
| 文件头 | 文件头定义 | 0-31 | 包括版本信息、更新时间、记录条数、文件头长度等 |
| 字段1定义 | 32-64 | 字段名称、类型、字段长度(Byte)、精度等 |
| 字段2定义 | 65-97 | 同上 |
| …… |  | 同上 |
| 字段n定义 | -n\*32+31 | 同上 |
| 值为0x0D | n\*32+32 | 表示终止字段定义 |
| 表格记录数据 | 第1行数据 | n\*32+33-X | 表示第1行数据 |
| 第2行数据 |  | 表示第2行数据 |
| …… |  |  |

注意，在表格记录数据中每行数据具体占多长字节，这个由文件头中定义的字段数目以及字段长度来决定，如果该文件一共只有两个字段，其中第一个字段为数值，其长度为4，第二个字段为字符串，长度为50，则每一行数据占的字节长度为4+50=54，在读取数据时也是读取前4个为第一个字段对应的值，读取第5-54个为第二个字段对应的值。

另外，为便于理解表格与下面内容的关系，特说明字段即是指表格中的列，记录指表格中的行数据，DBF按行数据方式来存储，即在文件头中定义了列数、列的名称、列的数据类型、列长度等等，然后在后面的记录数据中插入每行数据。

文件头中格式及说明如下：

|  |  |  |
| --- | --- | --- |
| 位置 | 类型 | 说明 |
| 0 | 1个字节 | 表示当前的版本信息：   * 0x02 FoxBASE * 0x03 FoxBASE+/Dbase III plus, no memo * 0x30 Visual FoxPro * 0x31 Visual FoxPro, autoincrement enabled * 0x43 dBASE IV SQL table files, no memo * 0x63 dBASE IV SQL system files, no memo * 0x83 FoxBASE+/dBASE III PLUS, with memo * 0x8B dBASE IV with memo * 0xCB dBASE IV SQL table files, with memo * 0xF5 FoxPro 2.x (or earlier) with memo * 0xFB FoxBASE |
| 1－3 | 3个字节 | 表示最近的更新日期，按照YYMMDD格式，以1900年为起始，即第一个字节表示文件最后保存时的年份-1900，第二个字节的值为保存时的月，第三个字节的值为保存时的日。 |
| 4－7 | Int32 | 文件中的记录条数，即表格的行数。 |
| 8－9 | Int16 | 文件头中的字节数，在此之后的字节为表格记录数据 |
| 10－11 | Int16 | 一条记录中的字节长度，即每行数据所占的长度 |
| 12－13 | 2个字节 | 保留字节，用于以后添加新的说明性信息时使用，这里用0来填写。 |
| 14 | 1个字节 | 表示未完成的操作。 |
| 15 | 1个字节 | dBASE IV编密码标记。 |
| 16－27 | 12个字节 | 保留字节，用于多用户处理时使用。 |
| 28 | 1个字节 | DBF文件的MDX标识。在创建一个DBF 表时 ，如果使用了MDX 格式的索引文件，那么 DBF 表的表头中的这个字节就自动被设置了一个标志，当你下次试图重新打开这个DBF表的时候，数据引擎会自动识别这个标志，如果此标志为真，则数据引擎将试图打开相应的MDX 文件。 |
| 29 | 1个字节 | 页码标记. |
| 30－31 | 2个字节 | 保留字节，用于以后添加新的说明性信息时使用，这里用0来填写。 |
| 32－N | （x\*32）个字节 | 这段长度由表格中的列数（即字段数，Field Count）决定，每个字段的长度为32，如果有x列，则占用的长度为x\*32，这每32个字节里面又按其规定包含了每个字段的名称、类型等信息，具体见下面的表。 |
| N＋1 | 1个字节 | 作为字段定义的终止标识，值为0x0D。 |

每个字段定义格式如下表，每个字段定义都用32个字节来完成：

|  |  |  |
| --- | --- | --- |
| 位置 | 内容 | 说明 |
| 0－10 | 11个字节 | 字段的名称，是ASCII码值。 |
| 11 | 1个字节 | 字段的数据类型，为ASCII码值。每个值对应不同的字段数据类型，如N表示数值，C表示字符串，关于具体的数据类型说明见下表。 |
| 12－15 | 4个字节 | 保留字节，用于以后添加新的说明性信息时使用，默认为0。 |
| 16 | 1个字节 | 字段的长度，表示该字段对应的值在后面的记录中所占的长度。 |
| 17 | 1个字节 | 字段的精度。 |
| 18－19 | 2个字节 | 保留字节，用于以后添加新的说明性信息时使用，默认为0。 |
| 20 | 1个字节 | 工作区ID。 |
| 21－31 | 11个字节 | 保留字节，用于以后添加新的说明性信息时使用，默认为0。 |

字段数据类型：

|  |  |  |
| --- | --- | --- |
| 代码 | 数据类型 | 允许输入的数据 |
| B | 二进制型 | 各种字符。 |
| C | 字符型 | 各种字符。 |
| D | 日期型 | 用于区分年、月、日的数字和一个字符，内部存储按照YYYYMMDD格式。 |
| G | (General or OLE) | 各种字符。 |
| N | 数值型(Numeric) | - **.** 0 1 2 3 4 5 6 7 8 9 |
| L | 逻辑型（Logical） | ? Y y N n T t F f (? 表示没有初始化)。 |
| M | (Memo) | 各种字符。 |