|  |
| --- |
| 哈尔滨工业大学(深圳) |
| **《数据库》实验报告** |
|  |
| **实验五**  **查询处理算法的模拟实现**  学 院: 计算机科学与技术   |  |  | | --- | --- | | 姓 名: | 孙铎 | | 学 号: | 200110503 | | 专 业: | 计算机科学与技术 | | 日 期: | 2023-01-01 | |

# 实验目的

*阐述本次实验的目的。*

* 理解索引、散列的作用；
* 掌握关系选择、投影、连接、集合的交、并、差等操作的实现算法；
* 加深对算法I/O复杂性的理解；
* 理解两阶段多路归并排序算法的思想、实现与应用；
* 理解、实现并运用简单的索引。

# 实验环境

*阐述本次实验的环境。*

* Windows 10 操作系统
* gcc version 8.1.0 (x86\_64-win32-seh-rev0, Built by MinGW-W64 project)
* CodeBlocks

# 实验内容

*阐述本次实验的具体内容。*

* 基于ExtMem程序库，模拟实现数据库的一些查询处理算法；
* 实现关系选择、连接操作算法；
* 实现简单的索引，并实现基于索引的关系选择算法；
* 实现集合并、交、差操作算法。

# 实验过程

*对实验中的5个题目分别进行分析，并对核心代码和算法流程进行讲解，用自然语言描述解决问题的方案 。并给出程序正确运行的结果截图。*

1. **对extmem.c的修改以及utils.c工具函数的封装**

* 修改extmem.c：
  + 修改freeBlockInBuffer函数：freeBlockInBuffer函数原本只将内存块标志位设置为可用，并没有真正清空内存块数据，这里将其修改为，**调用memset将内存块全部置0**并设置**标志位**为可用；
  + 修改writeBlockToDisk函数：将writeBlockToDisk函数中将内存块标志位设置为可用的代码修改为**对freeBlockInBuffer函数的调用**，这样实现的效果是一致的。
* 封装utils.c中的工具函数：
  + record2XY函数：
    - 输入：
      * blk：内存块指针
      * recordNum：要**读出**的块中的记录
      * X：接收该条记录的**前4个字节**的传出参数
      * Y：接收该条记录的**后4个字节**的传出参数
    - 输出：无
    - 函数实现：
      1. 声明字符数组str，长度为5
      2. 通过recordNum进行下标运算，读取blk对应记录的**前4个字节**到str中，并调用**atoi函数**将str转换成int型整数赋值给传出参数X
      3. 通过recordNum进行下标运算，读取blk对应记录的**后4个字节**到str中，并调用**atoi函数**将str转换成int型整数赋值给传出参数Y
  + XY2record函数：
    - 输入：
      * blk：内存块指针
      * recordNum：要**写入**的块中的记录
      * X：向记录**前4个字节**写入的数值
      * Y：向记录**后4个字节**写入的数值
    - 输出：无
    - 函数实现：
      1. 声明字符数组fourBytes，长度为4
      2. 判断X**是否等于-1**。如果不等于-1，则**调用sprintf将X转换为字符串**存储至fourBytes中，并通过recordNum进行下标运算，将fourBytes中的4个字节写入blk对应记录的**前4个字节**。如果等于-1，则将**数值0**写入blk对应记录的**前4个字节**。
      3. 判断Y**是否等于-1**。如果不等于-1，则**调用sprintf将Y转换为字符串**存储至fourBytes中，并通过recordNum进行下标运算，将fourBytes中的4个字节写入blk对应记录的**后4个字节**。如果等于-1，则将**数值0**写入blk对应记录的**后4个字节**。
  + nextAddr函数
    - 输入：
      * blk：内存块指针
    - 输出：blk块的后继块地址值
    - 函数实现：
      1. 声明字符数组str，长度为5
      2. 通过下标运算，读取blk**最后8个字节的前4个字节**到str中，**调用atoi函数**将str转换成int型整值，并返回该值。（约定将**后继块地址**存储在blk**最后8个字节的前4个字节**中）
  + shiftRecord函数
  + writeToOutBlk函数

1. **实现基于线性搜索的关系选择算法**

问题分析：

* 要将关系所在的磁盘块依次读入内存，并要依次判断各条记录是否符合选择条件，并输出符合选择条件的记录到磁盘块上。因此，需要重点实现的操作是：
  + 遍历读入的磁盘块中各条记录，**正确解析其数据值**，并判断是否符合选择条件。
  + 将符合选择条件的记录**暂存至新的内存块**，待内存块满时**输出至磁盘块**，同时也要设置好其**后继地址**。
* 核心代码与算法流程：
  1. 定义变量：
     + 1个**内存缓冲区buf**
     + 1个**内存块指针blk**用于存放**关系S的数据块**
     + 1个**内存块指针resBlk**用于暂存满足选择条件**待输出的记录**
     + 1个**记数变量rowCount**记录满足选择条件的**记录条数**；
  2. 调用**initBuffer函数**初始化总大小为520B，块大小为64B的内存缓冲区buf；
  3. 从起始磁盘块17开始，通过解析其后继地址，**遍历关系S的所有块**，并在每次遍历时执行如下操作：
     1. 遍历块中的前7条记录，调用record2XY函数将每条记录的S.C和S.D分别存入变量X与变量Y中
  4. 将**最后一个不超过7条选择结果记录**的resBlk写入磁盘。因为之前每次都是写满一块后，**下一次要写入新的块时才将上一块写入磁盘**，所以遍历结束后，只要存在满足选择条件的记录，resBlk中就一定有**剩余待输出的选择结果记录**；
  5. 输出一些结果信息，详见下方的**实验结果**。

实验结果：

1. **实现两阶段多路归并排序算法（TPMMS）**

问题分析：

* xxxxx
* 核心代码与算法流程：
  1. xxx

实验结果：

1. **实现基于索引的关系选择算法**

问题分析：

* xxxxx
* 核心代码与算法流程：
  1. xxx

实验结果：

1. **实现基于排序的连接操作算法（Sort-Merge-Join）**

问题分析：

* xxxxx
* 核心代码与算法流程：
  1. xxx

实验结果：

1. **实现基于排序的两趟扫描算法，实现交集合操作算法**

问题分析：

* xxxxx
* 核心代码与算法流程：
  1. xxx

实验结果：

# 附加题

*对剩余的两种集合操作进行问题分析，并给出程序正确运行的结果截图。*

1. **实现基于排序的两趟扫描算法，实现并集合操作算法**

问题分析：

* xxxxx
* 核心代码与算法流程：
  1. xxx

实验结果：

1. **实现基于排序的两趟扫描算法，实现差集合操作算法**

问题分析：

* xxxxx
* 核心代码与算法流程：
  1. xxx

实验结果：

# 总结

*总结本次实验的遇到并解决的问题、收获及反思。*

* 之后可以不用Codeblocks吗，Bug太多了
* 4个字节的存储方式，请说清存的是字符而不是二进制数
* 写个OJ可以不
* 明确要求，明确连续和离散，明确内存块利用率