CS33503数据库系统实验

实验检查记录

|  |  |  |  |
| --- | --- | --- | --- |
| 实验结果的正确性(60%) |  | 表达能力(10%) |  |
| 实验过程的规范性(10%) |  | 实验报告(20%) |  |
| 加分(5%) |  | 总成绩(100%) |  |

实验报告

一、实验目的（介绍实验目的）

|  |
| --- |
| 1. 掌握各种关系代数操作的实现算法，特别是连接操作的实现算法。  2. 在实验2完成的缓冲区管理器的基础上，使用C++面向对象程序设计方法实现查询执行器。 |

二、实验环境（介绍实验使用的硬件设备、软件系统、开发工具等）

|  |
| --- |
| 1. 硬件设备：Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz 2.71 GHz; 8GB RAM  2. 软件系统：Windows 10  3. 开发工具：Vmware 16 |

三、实验过程（介绍实验过程、设计方案、实现方法、实验结果等）

|  |
| --- |
| 1. 阅读源代码  为了完成本次实验，需要先阅读源代码，了解相关定义与实现。下面简要介绍几个对本次实验比较重要的方法。  HeapFileManager类的createTupleFromSQLStatement提供了将SQL语句翻译为元组的方法，其将元组各个属性的值按顺序存储，并进行一个4字节对齐的操作，空位用0补齐。  HeapFileManager类的insertTuple提供了将元组存入文件的方法，其将元组插入文件中有空余位置的页面中。  JoinOperator类的getCommonAttributes提供了提取左关系与右关系共同属性的方法。  JoinOperator类的construct\_search\_key提供了提取左（或右）关系中“共同属性”所对应的属性值的方法。  JoinOperator类的joinTuples提供了将左关系中一个元组与右关系中另一元组进行连接，得到新的元组表示的方法。  2. 实现NestedLoopJoinOperator类的execute方法  该方法实现的是基于块的嵌套循环连接操作。假设可用内存块数为M，则该方法的算法思路如下：  for 左关系R的每M-1块do  将这M-1块读入缓冲池  for 右关系S的每1块P do  将P读入缓冲池  for P中的每条元组s do  for 缓冲池中能与r进行连接的元组r do  连接r与s，并将连接结果暂存  将最终得到的全部连接结果写入指定文件  上述算法的最终实现代码如下：    3. 实验结果  首先运行make编译项目，然后运行./badgerdb\_main，结果如下：    可以看到，最终的连接结果包含500个元组，使用了3个缓冲区页面，进行了3次IO。 |

四、实验结论（总结实验发现及结论）

|  |
| --- |
| 查询执行器能执行多种操作，例如选择操作、投影操作、去重操作、聚集操作、集合操作和连接操作等。其中，连接操作又有多种执行算法，例如：一趟连接算法、嵌套循环连接算法、排序归并连接算法、哈希连接算法和索引连接算法。本次实验在实验2的基础上，通过C++编程实现了BadgerDB查询执行器中的连接操作，并通过基于块的嵌套循环连接算法来实现。在实验的过程中，我复习了查询执行器的相关知识，对嵌套循环连接算法有了更为深刻的理解。 |