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.掌握数据库管理系统的存储管理器的工作原理；  2.掌握数据库管理系统的缓冲区管理器的工作原理；  3.使用C++面向对象程序设计方法实现缓冲区管理器。 |

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）

|  |
| --- |
| 硬件：x64笔记本  软件系统：windows10  开发工具 ：vscode、VMware、Ubuntu桌面版 |

三、实验过程（介绍实验过程、设计方案、实现方法、实验结果等）

|  |
| --- |
| 一、~BufMgr()  1.写回脏页；  2. 释放缓冲池内存；  3. 释放bufDescTable；  4. 释放hashTable  二、advanceClock()    三、allocBuf(FrameId & frame)  1. 判断是否有空闲页；  2. 使用Clock算法挑选出一个空闲页框；  3. 如果该页面为脏页则需要先去刷新脏页；  4. 如果当前页面有效，则从hashtable中删除掉；  四、readPage(File\* file, const PageId pageNo, Page\*& page)  1、判断要访问的页面是否在缓存池中；  2、如果在，则：    3、如果不在，则：    五、unPinPage(File\* file, const PageId pageNo, const bool dirty)  1、判断要访问的页面是否在缓存池中；  2、如果在，则：    3、如果不在，则什么也不做，直接返回；  六、flushFile(const File\* file)    七、allocPage(File\* file, PageId &pageNo, Page\*& page)    八、disposePage(File\* file, const PageId PageNo)  1、判断要访问的页面是否在缓存池中；  2、如果在，则将它从缓冲池中删除；  3、从文件中删除该页面； |

四、实验结论（总结实验发现及结论）

|  |
| --- |
| 1、Clock算法是LRU算法的近似，依靠指针的旋转来挑选淘汰页；  2、当一个页面要被读写时，需要先被从磁盘上加载到缓存池中，缓冲池中的页面被修改后变成脏页，该页面在被淘汰时，需要先刷新到磁盘上； |