Lab3-Report

517030910412 陆晗

2020.5.8

Design decision

Eviction policy I use the LRU(least recent use) policy to decide which page to be evited. Because I think the FIFO(first in first out) method is not very reasonable and LRU is simple to implement. I use Map to record the pages in the Bufferpool with value. Each time the page is loaded to the buffer or be used, I will increase the value to capacity of the Bufferpool. Meanwhile, I will decrease other page value by 1 and make should they will not be below zero. If I need to evict one page, I will find the min value page.

API No change for API.

Incomplete elements I think none. If there exits, thanks very much to inform me and I will fill the hole as soon as quickly.

Others I spent **two days** in finishing the code. I find two things make me confused and cost me some time to explore. First is that Hashmap can recognize null as key while concunrrentHashmap cannot. So I make error in no group by as first. Second is the numpage() maybe need IO operation and division calculation. This will cost much time. And I did some structure change in the heapfile iterator implement. Good habits are really important.