6.830 Lab 2 writeup

Katie Siegel

Collaborated with Ankush Gupta

1 Design Decisions

The one significant design decision I made was with regards to the nested loops join. Even though I could see better implementations of the join, I chose the nested loops join for simplicity.

Additionally, I chose to use an arraylist when implementing the page eviction policy. Whenever a new page is added to the buffer pool, I appended it to the end of the arraylist, and when evicting a page, I would remove a page from the front of the arraylist. So, I would evict the oldest page in the Buffer Pool. I made this decision because then the most recently added, and thus most likely to be used soon, pages would stay in the Buffer pool.

Next, for my IntegerAggregator implementation, I chose to use switch statements to iterate through the different types of operators. This made sense because there were many different operators, all stored in an enumerator. However, in my StringAggregator implementation, I chose to use an if statement, because only the count operation needed to be supported. To generalize, the loop was simpler when only checking one case, whereas the switch statement makes sense when there were many.

Furthermore, I always tended towards over-checking whether passed-in values were null. I tended to lean on the safe side and make no assumptions on the values that were passed in, to make my code better able to handle unusual cases.

Most of the other parts of the lab were relatively straightforward in implementation, and did not require significant design decisions.

2 Changes to API

I did not make any changes to the API. The most I changed classes was to add a custom Iterator class to certain classes, which did not change the external-facing methods of those classes at all.

3 Missing or Incomplete Code

There is no missing or incomplete code in my project; it passes all of the tests.

4 General Comments

I spent around 14 hours on this lab. I got stuck for long stretches of time whenever I would run a system test; I found that I had to go back to my lab 1 code and debug. For example, I had an error in HeapFileIterator that took me around 5-6 hours to find, which caused me to be stuck on the 4th part of this project for a really long time.