

# CSE544 Homework 3 Writeup

Hai Van Le

February 2024

## 1 Answers

**Describe any design decisions you made. For example any class or complex data structure you add to the project. If you used something other than a nested-loops join, describe the tradeoffs of the algorithm you chose.**

I chose nested-loops join for the Join method. In particular, it iterates through each tuple of the outer relation (child1) and for each tuple, scans through the entire inner relation (child2) to find matching tuples based on the join predicate. A trade-off of using nested-loops join is that it can be inefficient for large datasets as it has a worst-case time complexity of  $O(n^2)$  where  $n$  is the number of tuples in the relations while for hashjoin, it is  $O(n + m)$ , where  $n$  and  $m$  are the number of tuples of two tables.

In BufferPool, the getPage() method retrieves the specified page with the associated permissions from the buffer pool. It first checks if the buffer pool has reached its maximum capacity. If yes, it throws a DbException indicating that the buffer pool is full. If the page is not present, it is fetched from the appropriate location, read from the database using "readPage" method and added to buffer pool. If the page is present, it returns it.

In HeapPage, getNumEmptySlots() calculates and returns the number of empty slots (unused slots) on the page. It first initializes a counter variable to zero to count the number of empty slots then iterate over each slot on the page. Next, for each slot, it check if it is unused. If yes, it increments the counter. Finally, after iterating over all slots, it returns the final count of empty slots. This function provides valuable information about the current occupancy status of the page. isSlotUsed(int i) is a function used to determine whether a specific slot on the page is currently occupied.

**Discuss and justify any changes you made to the API.**

I'm not sure how to answer this. I didn't make any changes.

**Describe any missing or incomplete elements of your code.**

I did not do the entire extra credit to complete the architecture of SimpleDB. I only passed unit and system tests for the main exercises and partial credits: IntegerArregator and StringArregator.

**Describe how long you spent on the assignment, and whether there was anything you found particularly difficult or confusing.**

I spent several days on the assignment. I ran into a lot of bugs, especially with HeapFile and BufferPool.