Database Systems Lab

Assignment 6: Relation Manager and External Sort Implementation in MuBase

Marks: 100

As part of this course, students of your senior batch developed the foundation layers of *MuBase*, a small database management system modeled after University of Wisconsin's MiniBase. They implemented storage manager to keep track of free/allocated disk blocks, buffer manager and a B+ tree indexing layer. In this assignment you will be extending the functionality of *MuBase* to provide support for relations. At the end of this assignment, *MuBase* should be able to support the following operations:

1. Creation of a relation with specified schema
2. Adding/deleting records from a relation
3. Full scan of a relation
4. Predicated scan (search) over a relation (should make use of index when available)
5. Sorting of a relation on a specified attribute

The attached tar.gz has 3 sub-directories. The subdir *interface* contains the interfaces of the various layers (already present in MuBase, as well as the ones you will be implementing). You will be providing the implementation for the classes in `rm.h` and `sort.h`. The subdir *slotted_page* contains interfaces for the slotted page classes. The subdir *mubase* contains the implementation of storage manager, buffer manager and B+ tree indexing layer. There may be some minor variations between this implementation (done by Varun and his team) and the interfaces provided in the *interface* subdir. Go through the README file, it contains important instructions.

Records in a relation will be stored in blocks using the slotted page organization. A framework for implementing the get, store and delete methods for a slotted page is also provided.

For sorting a relation you will be implementing the external sort algorithm (to be discussed in the class).

The project can be done in groups of at most 4 members. Please communicate your team composition by sending a mail to the TAs, with the subject line "Team for MuBase". Although this assignment can be done in groups, evaluation will be done based on the contribution of individuals. So split up the work equally and work independently on your chosen module.