**Overall Status**

First understood the algorithm thoroughly for the insert and the delete records . Studied the algorithm given in the textbook “Database management System by R. Ramakrishnan and J.Gehrke(Third edition)”.

Starting with insert algorithm, completed insert of root i.e. first leaf page and after that the remaining pages . After insert, completed delete records with all the occurrences of the requested key and delete batch records . Completed all the major components given .

**File Descriptions**

No additional files or functions were created as given functions were sufficient.

Program can delete range delete including all the occurrences of

Keys given in the range.

e.g. In case 5,If we have multiple records of 96 record and we have selected lower key as 95 and higher key as 97 then all the records i.e. 95,96,97 will be deleted.

**Division of Labor**

Project was done individually.

Time spent: three days 4 hours each(Around 12 hours)

**Logical errors and how you handled them**

While implementing the project encountered some logical errors

1. Looping the division of the pages when copying records from one page to other. Here firstly when divided the pages, first half records were getting copied to the new page and the remaining half to the old page i.e. first page division occurred on 63rd record in which 1-31 records were getting copied to new page created and 32-62 were copied to old page whereas it should be vice versa as function used was getFirst().

This logical error solved by firstly moving all the records to new page and again moving half records to the old page.

1. At the time of inserting 95th record recursive function was iterating infinitely as it wanted to insert record in index page and before that again recursive function was getting called.

Solved this logical error by first searching for the condition whether the current page id is equal to root page id if yes then insert in root page.

1. While deleting multiple records of similar occurrences was getting error on how to find the duplicate records and continuing the loop. Solved this by using getcurrent() function which returns occurrence

And continued the loop using continue.

1. It was thoughtful to find the condition for whether tree was empty or not