Database Systems Lab

SESSION 6

Update Operation

In this lab session, you will implement the DELETE operation to delete existing record in the data file.

Main PDS functions

A) pds_open

No change from Session 5

B) put_rec_by_key

```
// Seek to the end of the data file
// Create an index entry with the current data file location using ftell
// set is_deleted of index entry to FALSE (0)
// Add index entry to BST using offset returned by ftell
// Write the key at the current data file location
// Write the record after writing the key
int put_rec_by_key( int key, void *rec );
```

C) get_rec_by_ndx_key

```
// get_rec_by_key
// Search for index entry in BST
// Check status of is_deleted and proceed accordingly
// If is_deleted is TRUE, return PDS_REC_NOT_FOUND
// Seek to the file location based on offset in index entry
// Read the key at the current file location
// Read the record after reading the key
int get_rec_by_ndx_key( int key, void *rec );
```

D) pds_close

```
// Open the index file in wb mode (write mode, not append mode)
// Unload the BST into the index file by traversing it in PRE-ORDER
// Skip those index entries whose is_delete flag is TRUE
// Free the BST by call bst_destroy()
// Close the index file and data file
int pds_close();
```

E) get_rec_by_non_ndx_key

```
// Search based on a key field on which an index
// does not exist. This function actually does a full table scan
// by reading the data file until the desired record is found.
// Retrieve the corresponding index entry from BST
// Check the status of is_deleted in the corresponding index entry
// If is_deleted is TRUE, return PDS_REC_NOT_FOUND
// The io_count is an output parameter to indicate the number of records
// that had to be read from the data file for finding the desired record
```

F) modify_rec_by_key

No change from Session 5

G) delete_rec_by_key

```
// delete_rec_by_ndx_key
// Search for index entry in BST
// If index entry not found, return PDS_DELETE_FAILED
// Check status of is_deleted and proceed accordingly
// If is_deleted is TRUE, return PDS_DELETE_FAILED
// If is_deleted is FALSE, set it to TRUE and return PDS_SUCCESS
int delete_rec_by_ndx_key( int key );
```

Testing

1. Use the contact loader program given to you for loading a large number of records into PDS.
2. Test your program with various functions with the help of modified pds_tester.c given to you along with test case files given to you.

Submission

- Upload the ONLY pds.c file to LMS