Υλοποίηση Συστημάτων Βάσεων Δεδομένων
2019/20
Εργασία 3

Καλλίνικος Γρηγόρης - 1115201500056
Παιδάκης Θεοδόσης - 1115201500118

Implementation notes:

- AM_Init() is used to initialize the global structures that are needed for the usage of the B+ Tree.

- AM_CreateIndex() creates a file with name fileName, that is based on a B+ Tree. The file must not already exist. The type and length of the first field(which is used for the insertion in the B+ Tree as a key) are described by the second and third parameter, correspondingly. Samewise, the type and the length of the second field are described by the fourth and fifth parameter.

- AM_DextroyIndex() destroys the file with name fileName, deleting the file from the disk. The file cannot be deleted if opens of it exist in the Files_array.

- AM_OpenIndex() opens the file with name fileName. If the file is normally opened, the function returns a small, non-negative integer, which is used to recognize the file. In any other case, it returns an error code.
There is a Files_array kept in the memory for all the opened files. The integer that is returned by the AM_OpenIndex is the position of the table that corresponds to the file that was just opened. The same file may be opened many times and each time it occupies a different position in the table.

- AM_CloseIndex() closes the file that is defined by its parameter. It also removes the entry that corresponds to that file from the Files_array. In order for the file that is defined by the parameter fileDesc to be closed successfully, there must not be opened scans of it.

- AM_InsertEntry() inserts the pair(value1, value2) at the file that is pointed by the parameter fileDesc. The parameter value1 points to the value of the key-field that is inserted to the file and the value2 represents the other field of the record.

- insertEntry() is the recursive insert entry that follows the path from the root down to the leaf that the entry needs to be placed. Then it inserts the entry if there is space, or it splits the leaf into two leafs with equal number of entries, and recursively sends the key of the entry to be inserted to the parent node(which may need to be split).

fileDesc - holds the index of the Files_array in which the file that the insert will take place is.
nodePointer - holds the number of the node/block that the insertion will take place.
newchildentry - is NULL except for the case that there was a split, and it holds the pair <key-value, block-number> that will be inserted to the node parent.
key-value is the first value of the new block that was created due to the split.
block-number is the number of the new block that holds the splitted entries.

- AM_OpenIndexScan() opens a scan(search) of the file that is defined by the parameter fileDesc. This scan has the purpose to find the records whose values in the key-field of the file satisfy the comparison operator op, with regards to the value that is pointed by the parameter value.

The various comparison operators are codes as follows:

* [-] 1 EQUAL (key-field == value)
* [-] 2 NOT EQUAL (key-field != value)
* [-] 3 LESS THAN (key-field < value)
* [-] 4 GREATER THAN (key-field > value)
* [-] 5 LESS THAN or EQUAL (key-field <= value)
* [-] 6 GREATER THAN or EQUAL (key-field >= value)

The function returns a non-negative integer that corresponds to a position of the Scan_array that is implemented and kept updated in the memory in regards to all the scans that are opened at each moment.

- AM_FindNextEntry() returns the value of the second field of the next entry that satisfies the condition that is defined by the scan that corresponds to scanDesc. If there are no more records if returns NULL and sets the global variable AM_errno to AME_EOF.

- AM_CloseIndexScan() terminates the scan of a file and removes the corresponding registry from the table of open scans.

- AM_PrintError() prints the text that is pointed by errString and then prints the message that corresponds to the last error that derived by any of the AM functions. For this purpose, this function uses a global variable AM_errno which is correctly updated in all the rest functions.

- AM_Close() is used to destroy all the structures that have been initialized.