CS 6543-001 Advanced Database Systems (Spring 2024)

Program 02: (60 points points)

Due date: 03/12/24 (Tuesday) at the beginning of class.

NOTE: Please submit this assignment to CSCADE. The main program should be named prog02.cpp (or files in c or in Java). You need to submit all the programs, header files (if any).

Requirements:

- All of the programs should be well-documented. (40%)
- All of the programs must be in good programming format.

Program assignment:

To simulate Tiered LMS-tree. To simplify the assignment, no deletion is considered.

- System setup:
  a. Constant *BLOCKSIZE* = 2 (or other value) is used as the number of records can be held in a block.
  b. Constant *THRESHOLD*=4 (or other value) is used as the size ratio between levels.
  c. Mem[*MEMSIZE*]: an array of *MEMSIZE* records store data at level 0.
     - *MEMSIZE* is a constant number for total records (integers) can be held in Mem[]. (This constant *MEMSIZE* should be a number greater than or equal to (*THRESHOLD+1)* BLOCKSIZE*.)
  d. Each layer of LSM has up to THRESHOLD files. The file name for tier ***t*** of layer ***i*** should be in the format: L***i-t***.txt. For example, L1-0.txt is the tier (chunk) 0 of data at level 1.
  e. Between records in a file, there is a newline ('\n').
  f. An array (or vector) to keep track of the number of chunks at each level.
  g. Record structure:

```
struct RecStruct{
    int key;
    int value;
    // int status;     // 0: for normal, 1 for delete tombstone
};
```

- work:

  For each record read from the input file (prog02Data.txt):

  a. If end of file, write the contents (records) of Mem into L0.txt, write the number of chunks at each level to LevelInfo.txt, close all files and exit the program.
  b. Maintain records in Mem sorted.
  c. If repeated key, replace the old one with the newly read one.

d. If allocated Mem is full, flush to L1-i.txt where -i in the file name is a number to indicate the chunk id at level 1.
e. At level j, if the number of files is *THRESHOLD*, flush and merge-sort to next level, j+1.
    - Repeat this process if needed after flush and merge-sort to the next level