Data structures
B-trees

1 Indexed databases

A database index is a data structure that improves the speed of data retrieval operations on a database table at the cost of additional writes and storage space to maintain the index data structure. Indexes are used to quickly locate data without having to search every row in a database table every time a database table is accessed. Indexes can be created using one or more columns of a database table, providing the basis for both rapid random lookups and efficient access of ordered records.

In this assignment, we will simulate the database using a large csv file. We will implement a B-tree and use it as an index of the database to quickly retrieve the requested records.

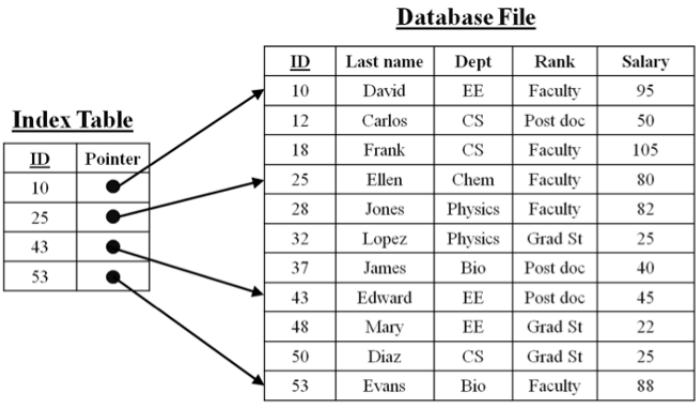

Figure 1: A database index.

2 Assignment

- The provided start code contains several .csv files of different lengths, these represent a database that is too large to keep in memory.
- The Database.java file can search the .csv files. To do this, it just performs a linear search through the file.
- Implement a B-tree that can store (int, long) key, values.
- Implement the constructor of IndexedDatabase. Read the entire file, line by line and store the id of the record with the corresponding location in the file in the b-tree. The RandomAccessFile class has method "getFilePointer" that returns the current offset while reading the file.
- Implement the "findById" method. First, retrieve the file location from the b-tree, then read the line at that location from the file and return the information.
- How much faster is it to use the index ?