Review of Keyterms for Midterm

Reference: Folk, Zoellick and Riccardi - Section "Key Terms" at the end of each Chapter.
Note: This review only reminds you about some key terms related to concepts studied in this course; you should refer to the Chapters and class notes in order to review each topic in depth.

Chapter 2 - Fundamental File Processing Operations

Physical File - A file that actually exists on secondary storage. It is the file as known by the computer operating system and that appears in its file directory.

Logical File - The file as seen by the program. The use of logical files allows a program to describe operations to be performed on a file without knowing what physical file will be used. The program may then be used to process any one of a number of different files that share the same structure.

Open - A function or system call that makes a file ready for use. It may also bind a logical file name to a physical file. Its arguments include the logical file name and the physical file name and may also include information on how the file is expected to be accessed.

Close - A function or system call that breaks the link between a logical file name and the corresponding physical file name.

Read - A function or system call used to obtain input from a file or device. When viewed at the lowest level, it requires three arguments: (1) a Source file logical name corresponding to an open file; (2) the Destination address for the bytes that are to be read; and (3) the Size or amount of data to be read.

Write - A function or system call used to provide output capabilities. When viewed at the lowest level, it requires three arguments: (1) a Destination file name corresponding to an open file; (2) the Source address of the bytes that are to be written; and (3) the Size or amount of the data to be written.

Seek - A function or system call that sets the read/write pointer to a specified position in the file. Languages that provide seeking functions allow programs to access specific elements of a file *directly*, rather than having to read through a file from the beginning (*sequentially*) each time a specific item is desired. In C, the `fseek` function provides this capability.

Chapter 3 - Secondary Storage and System Software

Magnetic Disks

Sector Organization - Disk drive organization that uses sectors.

Block Organization - Disk drive organization that allows the user to define the size and organization of blocks and then access a block by giving its block address or the key of one of its records.

Sector - The fixed-sized data blocks that together make up the tracks on certain disk drives. Sectors are the smallest addressable unit on a disk whose tracks are made up of sectors.

Tracks - The set of bytes on a single surface of a disk that can be accessed without seeking (without moving the access arm). The surface of a disk can be thought of as a series of concentric circles with each circle corresponding to a particular position of the access arm and read/write heads. Each of these circles is a track.

Cylinder - The set of tracks on a disk that are directly above and below each other. All of the tracks in a given cylinder can be accessed without having to move the access arm - they can be accessed without the expense of seek time.

Count Subblock - On block-organized drives, a small block that precedes each data block and contains information about the data block, such as its byte count and its address.

Seek Time - The time required to move the access arm to the correct cylinder on a disk drive.

Rotational Delay - The time it takes for the disk to rotate so the desired sector is under the read/write head.

Transfer Time - Once the data we want is under the read/write head, we have to wait for it to pass under the head as we read it. The amount of time required for this motion and reading is the transfer time.

Cluster - Minimum unit of space allocation on a sectored disk, consisting of one or more contiguous sectors. The use of large clusters can improve sequential access times by guaranteeing the ability to read longer spans of data without seeking. Small clusters tend to decrease internal fragmentation.

Extent - One or more adjacent clusters allocated as part (or all) of a file. The number of extents in a file reflects how dispersed the file is over the disk. The more dispersed a file, the more seeking must be done in moving from one part of the file to another.

Magnetic Tapes

Blocking factor - The number of records stored in one block.

Effective recording density - Recording density after taking into account the space used by interblock gaps, nondata subblocks, and other space-consuming items that accompany data.

Nominal recording density - Recording density on a disk track or magnetic tape without taking into account the affects of gaps or nondata subblocks.

Effective transmission rate - Transmission rate after taking into account the time used to locate and transmit the block of data in which a desired record occurs.

Nominal transmission rate - Transmission rate of a disk or tape unit without taking into account the effects of such extra operations as seek time for disks and interblock gap traversal time for tapes.

CD-ROM

(no key terms included, but students should review the topic)

System Software

File manager - The part of an operating system that is responsible for managing files, including a collection of programs whose responsibilities range from keeping track of files to invoking I/O processes that transmit information between primary and secondary storage.

File allocation table (FAT) - A table that contains mappings to the physical locations of all the clusters in all files on disk storage.

I/O processor - A device that carries out I/O tasks, allowing the CPU to work on non-I/O tasks.

Controller - Device that directly controls the operation of one or more secondary storage devices, such as disk drives and magnetic tape units.

Buffering - When input or output is saved up rather that sent off to its destination immediately, we say that it is *buffered*. In later chapters, we find that we can dramatically improve the performance of programs that read and write data if we buffer the I/O.

Chapter 4 - Fundamental File Structure Concepts

Field - The smallest logically meaningful unit of information in a file. A record in a file is usually made up of several fields.

Record - A collection of related fields. For example, the name, address, and so on of an individual in a mailing-list file would make up one record.

Fixed-length record - A file organization in which all records have the same length. Records are padded into blanks, nulls, or other characters so they extend to the fixed length. Since all the records have the same length, it is possible to calculate the beginning position of any record, making *direct access* possible.

Variable-length record - A file organization in which the records have no predetermined length. They are just as long as they need to be and therefore make better use of space than fixed-length records do. Unfortunately, we cannot calculate the byte offset of a variable-length record by knowing only its relative record number.

Chapter 5 - managing Files of Records

Key - An expression derived from one or more of the field within a record that can be used to locate that record. The fields used to build the key are sometimes called the *key fields*. Keyed access provides a way of performing content-based retrieval of records, rather than retrieval based merely on a record's position.

Primary Key - A key that uniquely identifies each record and is used as the primary method of accessing the records.

Sequential Access - Sequential access to a file means reading the file from the beginning and continuing until you have read in everything that you need. The alternative is direct access.

Direct Access - A file accessing mode that involves jumping to the exact location of a record. Direct access to a fixed-length record is usually accomplished by using its *relative record number* (RRN), computing its byte offset, and then seeking to the first byte of the record.

Relative record number (RRN) - An index giving the position of a record relative to the beginning of its file. If a file has fixed-length records, the RRN can be used to calculate the *byte offset* of a record so the record can be accessed directly.

Chapter 6 - Organizing Files for Performance

Data compression

Data compression - Encoding information in a file in such a way as to take up less space.

Huffman code - A variable-length code in which the lengths of the codes are based on their probability of occurrence.

Lempel-Ziv code - A dynamic compression technique you have studied in detail.

Run-length encoding - A compression method in which runs of repeated codes are replaced by a count of the number of repetitions of the code, followed by the code that is repeated.

Reclaiming Space in Files

Fragmentation - The unused space within a file. The space can be locked within individual records (*internal fragmentation*) or between individual records (*external fragmentation*).

Internal fragmentation - A form of fragmentation that occurs when space is wasted in a file because it is locked up, unused, inside of records. Fixed-length record structures often result in internal fragmentation.

External fragmentation - A form of fragmentation that occurs in a file when there is unused space outside or between individual records.

Avail list - A list of the space, freed through record deletion, that is available for holding new records. In the example considered in this chapter, this list of space took the form of a linked list of deleted records.

Placement strategy - As used in this chapter, a placement strategy is a mechanism for selecting the space on the avail list that is to be used to hold a new record added to the file. (ex. first fit, best fit, worst fit)

Internal Sorting and Binary Searching

Binary Searching - A binary search algorithm locates a key in a sorted list by repeatedly selecting the middle element of the list, dividing the list in half, and forming a new, smaller list from the half that contains the key. This process is continued until the selected element is the key that is sought.

Internal sorting - A method for sorting data that is completely contained in main memory.

External sorting - A method for sorting data that cannot fit all at once in main memory.

Keysorting

Keysort - A method of sorting a file that does not require holding the entire file in memory. Only the keys are held in memory, along with pointers that tie these keys to the records in the file from which they are extracted. The keys are sorted, and the sorted list of keys is used to construct a new version of the file that has the records in sorted order. The primary advantage of a keysort is that it requires less memory than a memory sort. The disadvantage is that the process of constructing a new file requires a lot of seeking for records.

Pinned records - A record is pinned when there are other records or file structures that refer to it by its physical location. It is pinned in the sense that we are not free to alter the physical location of the record: doing so destroys the validity of the physical references to the record. These references become useless dangling pointers.

Chapter 7 - Indexing

Index - An index is a tool for finding records in a file. It consists of a *key field* on which the index is searched and a *reference field* that tells where to find the data file record associated with a particular key.

Inverted list - The term *inverted list* refers to indexes in which a key may be associated with a *list* of reference fields pointing to documents that contain the key. The secondary indexes developed toward the end of this chapter are examples of inverted lists.

Entry-sequenced file - A file in which the records occur in the order that they are entered into the file.

Binding - Binding takes place when a key is associated with a particular physical record in the data file. In general, binding can take place either during the preparation of the data file and indexes or during program execution. In the former case, called *tight binding*, the indexes contain explicit references to the associated physical data record. In the latter case, the connection between a key and a particular physical record is postponed until the record is retrieved in the course of program execution.

Chapter 8 - Cosequential Processing (only up to section 8.1)

Cosequential operations - Operations applied to problems that involve the performance of union, intersection, and more complex set operations on two or more sorted input files to produce one or more output files built from some combination of the elements of the input files. Cosequential operations commonly occur in matching, merging and file-updating problems.

Match - The process of forming a sorted output file consisting of all the elements common to two or more sorted input files.

Merge - The process of forming a sorted output file that consists of the union of the elements from two or more sorted input files.