Operating System

Prof. Vibhuti Patel, Assistant Professor
Parul Institute of Computer Application

Parul® University

DIGITAL LEARNING CONTENT

CHAPTER-4

Disk Scheduling and Files

Disk Scheduling

- Disk scheduling is done by OS to schedule I/O requests arriving for the disk.
- Disk scheduling is also called as I/O scheduling.
- Disk scheduling is very important because, multiple I/O requests may arrive by different processes and only one I/O request can be served at a time by the disk controller.

Disk Scheduling Terminology

- Seek Time: Seek time is the time taken to locate the disk arm to a specified track where the data is to be read or write. So the disk scheduling algorithm that gives minimum average seek time is better.

- Rotational Latency: Rotational Latency is the time taken by the desired sector of disk to rotate into a position so that it can access the read/write heads. So the disk scheduling algorithm that gives minimum rotational latency is better.

- Transfer Time: Transfer time is the time to transfer the data. It depends on the rotating speed of the disk and number of bytes to be transferred.

Disk Scheduling Terminology

- Disk Access Time:

  Disk Access Time = Seek Time + Rotational Latency + Transfer Time

- Disk Response Time: Response Time is the average of time spent by a request waiting to perform its I/O operation.

Disk Scheduling Algorithms

1. First In First Out (FIFO)
2. Short Service Time First (SSTF)
3. SCAN
4. Circular-SCAN (C- SCAN)

Example:

A disk with 200 tracks and that the disk request queue has random requests in it. The requested tracks, in the order received by the disk scheduler, are

55, 58, 39, 18, 90, 160, 150, 38, 184.

First In First Out

- The simplest form of scheduling is FIFO scheduling, which processes items from the queue in sequential order.

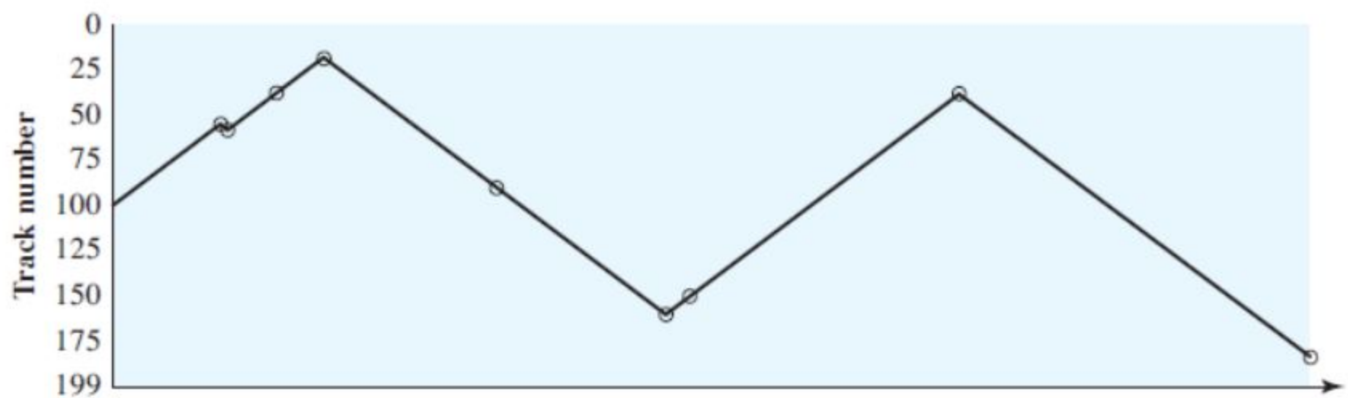


Image source : Book

First In First Out

(a) FIFO (starting at track 100)	
Next track accessed	Number of tracks traversed
55	45
58	3
39	19
18	21
90	72
160	70
150	10
38	112
184	146
Average seek length	55.3

Image source : Book

Short Service Time First

- The SSTF algorithm is select the disk I/O request that requires the least movement of the disk arm from its current position.

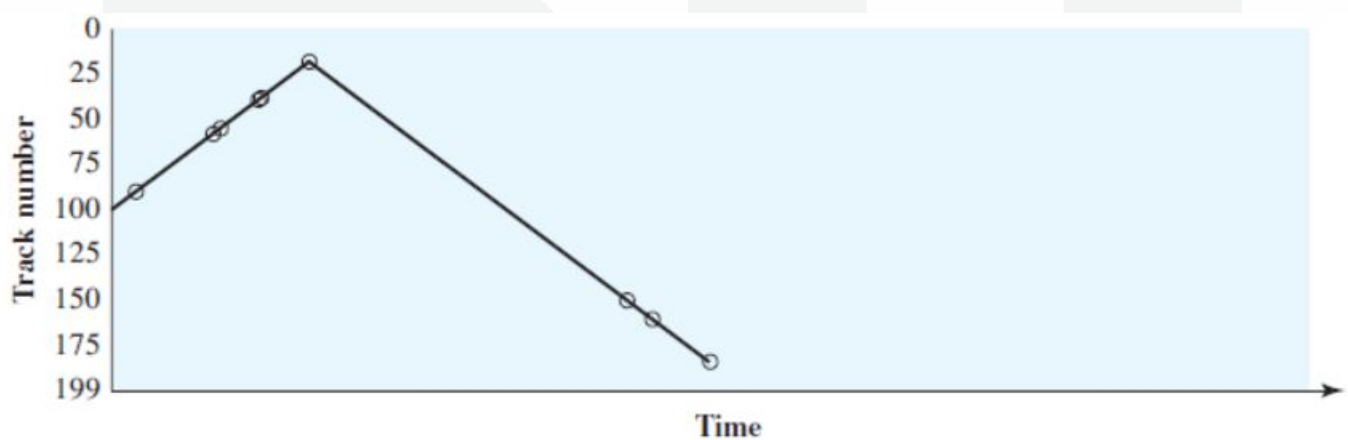


Image source : Book

Short Service Time First

(b) SSTF (starting at track 100)	
Next track accessed	Number of tracks traversed
90	10
58	32
55	3
39	16
38	1
18	20
150	132
160	10
184	24
Average seek length	27.5

Image source : Book

SCAN

- It is also known as the elevator algorithm because it works as elevator.
- It is required to move in one direction only, satisfying all outstanding requests in route, until it reaches the last track in that direction or until there are no more requests in that direction. This refinement is sometimes referred to as the LOOK policy. The service direction is then reversed and the scan proceeds in the opposite direction, again picking up all requests in order.

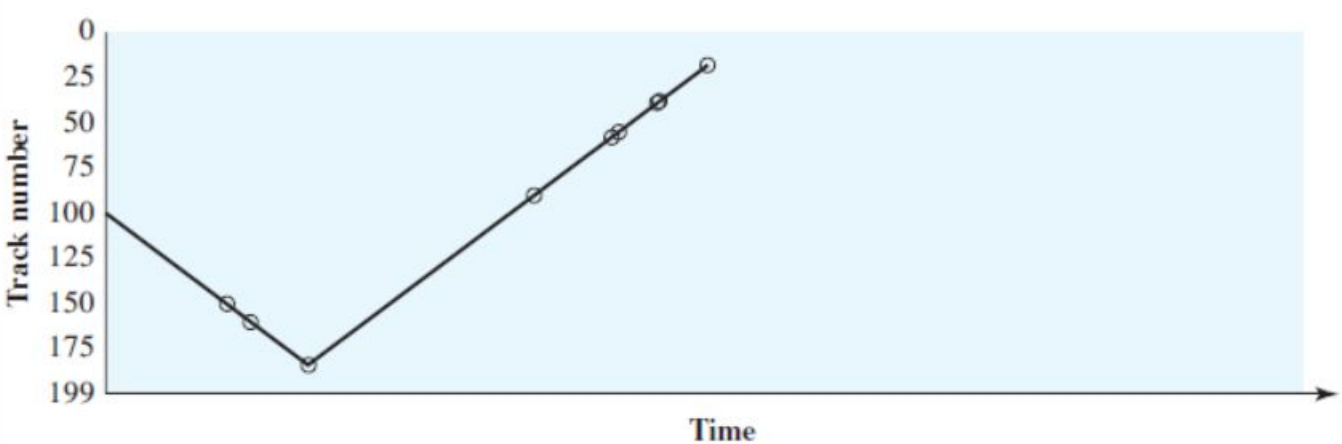


Image source : Book

SCAN

(c) SCAN (starting at track 100, in the direction of increasing track number)

Next track accessed	Number of tracks traversed
150	50
160	10
184	24
90	94
58	32
55	3
39	16
38	1
18	20
Average seek length	27.8

Image source : Book

Parul® University

DIGITAL LEARNING CONTENT

Circular - SCAN

- It restricts scanning to one direction only. Thus, when the last track has been visited in one direction, the arm is returned to the opposite end of the disk and the scan begins again. This reduces the maximum delay experienced by new requests.

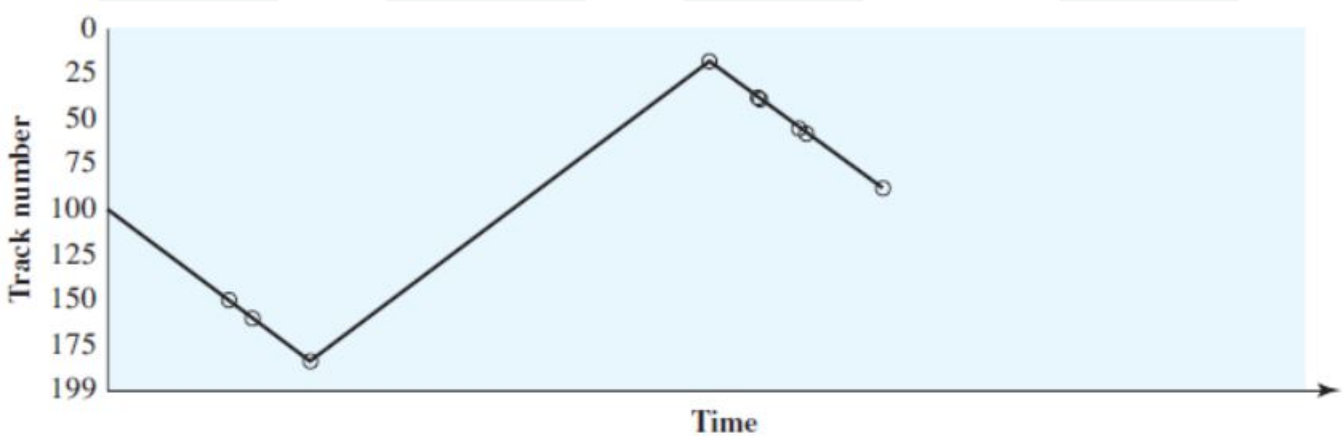


Image source : Book

Circular - SCAN

(d) C-SCAN (starting at track 100, in the direction of increasing track number)	
Next track accessed	Number of tracks traversed
150	50
160	10
184	24
18	166
38	20
39	1
55	16
58	3
90	32
Average seek length	35.8

Image source : Book

RAID (Redundant Array of Independent Disk)

- The RAID scheme consists of seven levels, 0 through 6. These levels do not imply a hierarchical relationship but designate different design architectures that share three common characteristics:
- RAID is a set of physical disk drives viewed by the operating system as a single logical drive.
- Data are distributed across the physical drives of an array in a scheme known as striping.
- Redundant disk capacity is used to store parity information, which guarantees data recoverability in case of a disk failure.

RAID 0 (Non Redundant)

- For RAID 0, the user and system data are distributed across all of the disks in the array.
- This has a notable advantage over the use of a single large disk: If two different I/O requests are pending for two different blocks of data, then there is a good chance that the requested blocks are on different disks.
- Thus, the two requests can be issued in parallel, reducing the I/O queuing time.

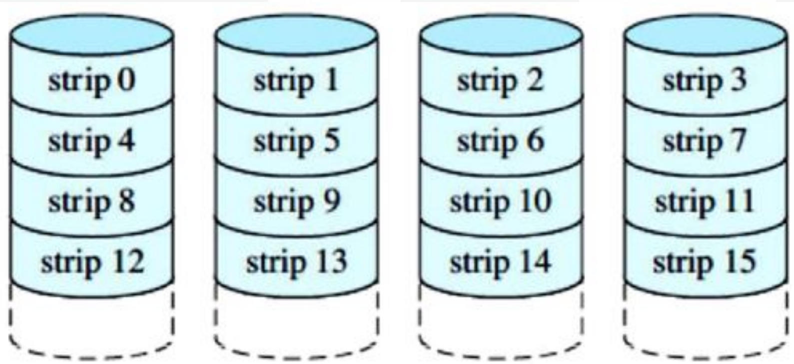


Image source : Book

RAID 1 (Mirrored)

- In RAID 1, redundancy is achieved by the simple expedient of duplicating all the data. Figure shows data striping being used, as in RAID 0. But in this case, each logical strip is mapped to two separate physical disks so that every disk in the array has a mirror disk that contains the same data. RAID 1 can also be implemented without data striping, though this is less common.

- There are a number of positive aspects to the RAID 1 organization:

1. A read request can be serviced by either of the two disks that contains the requested data, whichever one involves the minimum seek time plus rotational latency.

.

RAID 1 (Mirrored)

2. A write request requires that both corresponding strips be updated, but this can be done in parallel. Thus, the write performance is dictated by the slower of the two writes (i.e., the one that involves the larger seek time plus rotational latency).

3. Recovery from a failure is simple. When a drive fails, the data may still be accessed from the second drive.

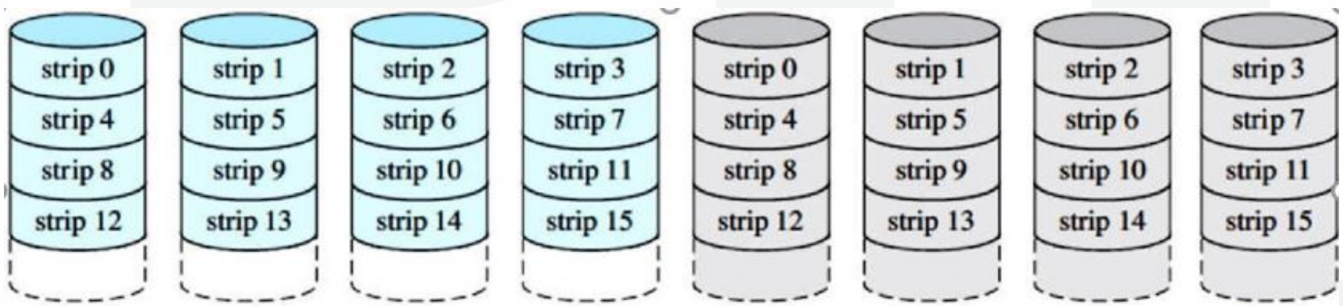


Image source : Book

RAID 1 (Mirrored)

- The main disadvantage of RAID 1 is the cost; it requires twice the disk space of the logical disk that it supports. Because of that, a RAID 1 configuration is likely to be limited to drives that store system software and data and other highly critical files.

RAID 2 (Redundancy through Hamming Code)

- RAID levels 2 make use of a parallel access technique. In a parallel access array, all member disks participate in the execution of every I/O request.

- Typically, the spindles of the individual drives are synchronized so that each disk head is in the same position on each disk at any given time.

- Although RAID 2 requires fewer disks than RAID 1, it is still rather costly. The number of redundant disks is proportional to the log of the number of data disks.

- On a single read, all disks are simultaneously accessed. The requested data and the associated error-correcting code are delivered to the array controller.

RAID 2 (Redundancy through Hamming Code)

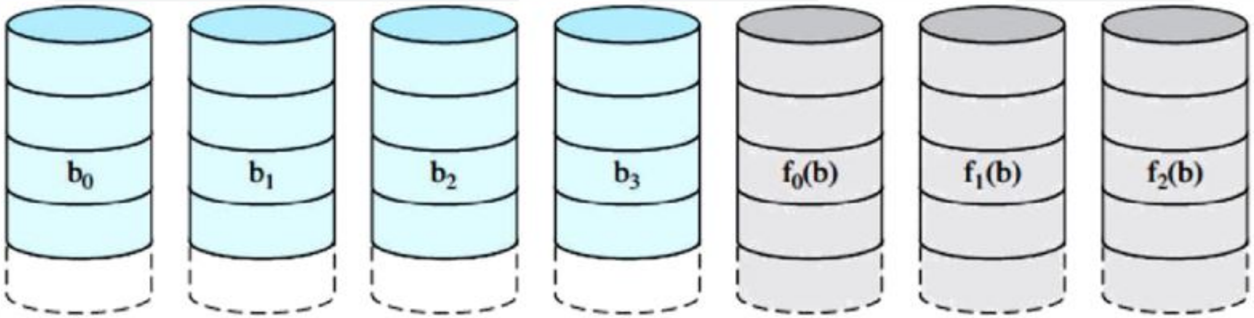


Image source : Book

RAID 3 (Bit Interleaved Parity)

- RAID 3 is organized in a similar fashion to RAID 2. The difference is that RAID 3 requires only a single redundant disk, no matter how large the disk array. RAID 3 employs parallel access, with data distributed in small strips. Instead of an error correcting code, a simple parity bit is computed for the set of individual bits in the same position on all of the data disks.

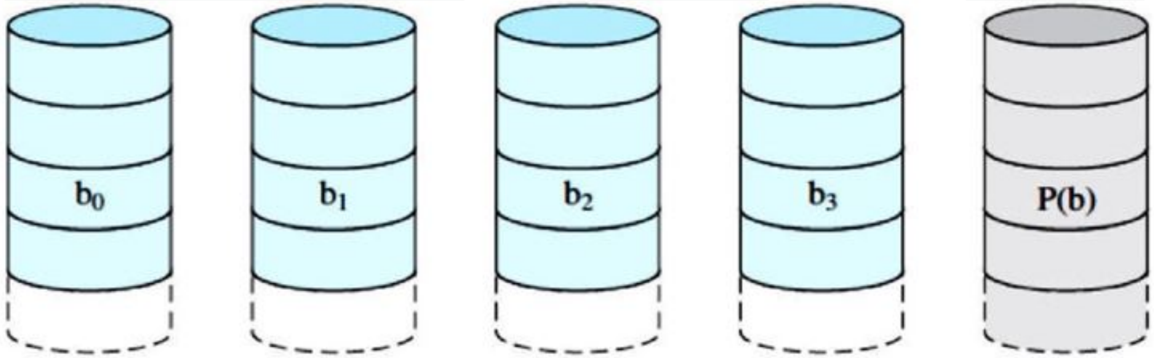


Image source : Book

RAID 4 (Block Level Parity)

- RAID levels 4 make use of an independent access technique. In an independent access array, each member disk operates independently, so that separate I/O requests can be satisfied in parallel.

- Because of this, independent access arrays are more suitable for applications that require high I/O request rates and are relatively less suited for applications that require high data transfer rates.

- RAID 4 involves a write penalty when an I/O write request of small size is performed. Each time that a write occurs, the array management software must update not only the user data but also the corresponding parity bits.

RAID 4 (Block Level Parity)

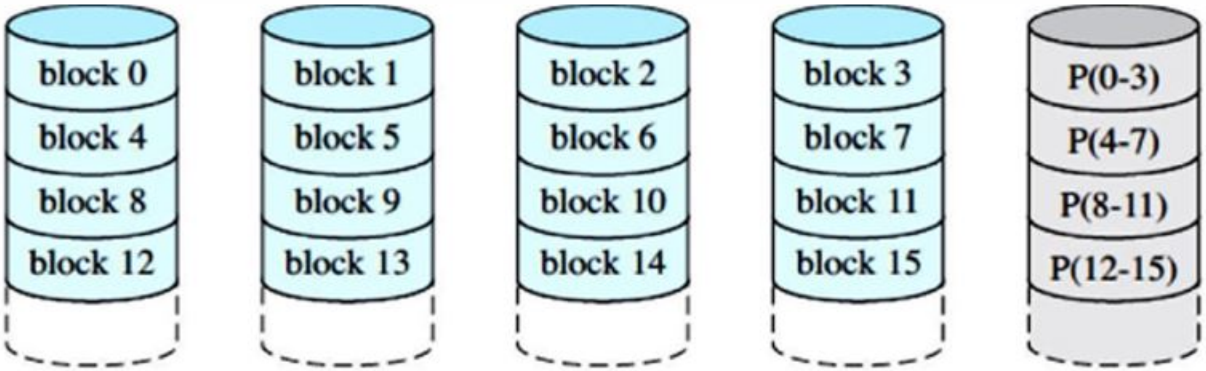


Image source : Book

RAID 5 (Block Level Distributed Parity)

- RAID 5 is organized in a similar fashion to RAID 4. The difference is that RAID 5 distributes the parity strips across all disks.

- A typical allocation is a round-robin scheme, as illustrated in Figure. For an n-disk array, the parity strip is on a different disk for the first n stripes, and the pattern then repeats.

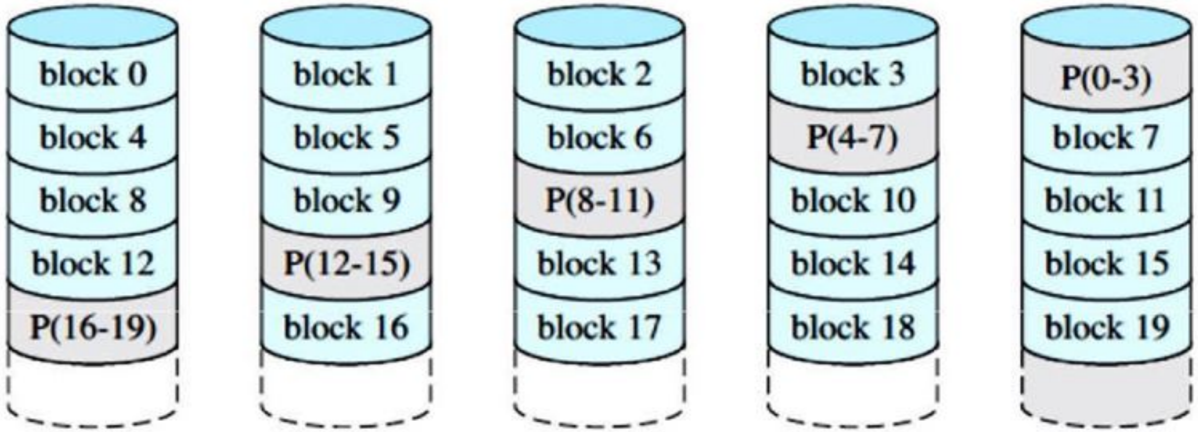


Image source : Book

RAID 6 (Dual Redundancy)

- In the RAID 6 scheme, two different parity calculations are carried out and stored in separate blocks on different disks. Thus, a RAID 6 array whose user data require N disks consists of N+2 disks.

- Figure illustrates the scheme. P and Q are two different data check algorithms. One of the two is the exclusive-OR calculation used in RAID 4 and 5. But the other is an independent data check algorithm.

- This makes it possible to regenerate data even if two disks containing user data fail. The advantage of RAID 6 is that it provides extremely high data availability.

RAID 6 (Dual Redundancy)

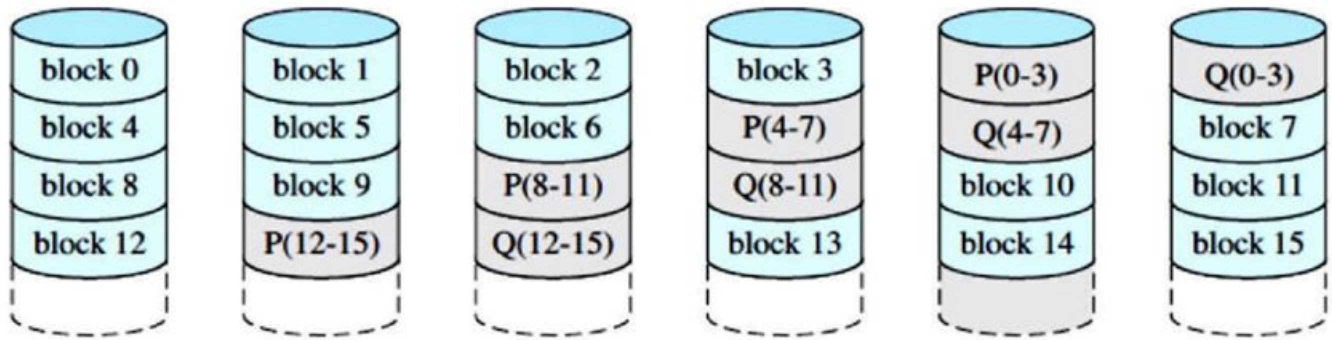


Image source : Book

Disk Cache

- A disk cache is a mechanism for improving the time it takes to read from or write to a hard disk.
- The disk cache is usually included as part of the hard disk. A disk cache can also be a specified portion of random access memory (RAM).
- The disk cache holds data that has recently been read and, in some cases, adjacent data areas that are likely to be accessed next. Write caching is also provided with some disk caches.

File Management Concepts

Directory: Collection of files is a file directory. The directory contains information about the files, including attributes, location and ownership.

- Much of this information, especially that is concerned with storage, is managed by the operating system.

- The directory is itself a file, accessible by various file management routines.

Directory

Information contained in a device directory are:

- Name
- Type
- Address
- Current length
- Maximum length
- Date last accessed
- Date last updated
- Owner id
- Protection information

Directory

Operation performed on directory are:

- Search for a file
- Create a file
- Delete a file
- List a directory
- Rename a file
- Traverse the file system

Directory

Advantages of maintaining directories are:

- Efficiency: A file can be located more quickly.
- Naming: It becomes convenient for users as two users can have same name for different files or may have different name for same file.
- Grouping: Logical grouping of files can be done by properties e.g. all java programs, all games etc.

File

- A file is a named collection of related information that is recorded on secondary storage such as magnetic disks, magnetic tapes and optical disks.

- In general, a file is a sequence of bits, bytes, lines or records whose meaning is defined by the files creator and user.

File Organization

There are five different File organizations used.

1. The pile
2. The sequential file
3. The indexed sequential file
4. The indexed file
5. The direct, or hashed, file

The Pile (Variable Length)

- The pile is the least-complicated form of file organization. Data are collected in the order in which they arrive.
- Each record consists of one burst of data. The purpose of the pile is simply to accumulate the mass of data and save it.
- Records may have different fields, or similar fields in different orders. Thus, each field should be self- describing, including a field name as well as a value.
- The length of each field must be implicitly indicated by delimiters, explicitly included as a subfield, or known as default for that field type.

Parul® University

DIGITAL LEARNING CONTENT

The Pile (Variable Length)

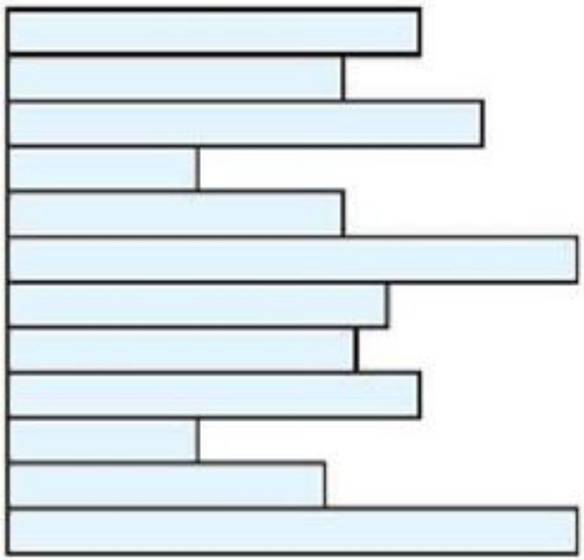


Image source : Book

The Sequential File (Fixed Variable Length)

- The most common form of file structure is the sequential file. In this type of file, a fixed format is used for records.
- All records are of the same length, consisting of the same number of fixed-length fields in a particular order. Because the length and position of each field are known, only the values of fields need to be stored; the field name and length for each field are attributes of the file structure.
- One particular field, usually the first field in each record, is referred to as the key field. The key field uniquely identifies the record; thus key values for different records are always different

The Sequential File (Fixed Variable Length)

- Further, the records are stored in key sequence: alphabetical order for a text key, and numerical order for a numerical key.

- Sequential files are typically used in batch applications and are generally optimum for such applications if they involve the processing of all the records (e.g., a billing or payroll application).

- The sequential file organization is the only one that is easily stored on tape as well as disk.

The Sequential File (Fixed Variable Length)

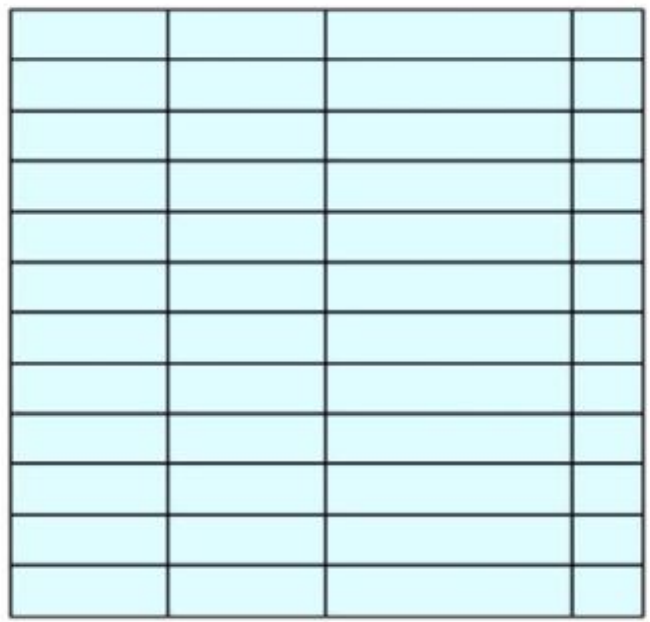


Image source : Book

The Indexed Sequential File

- The indexed sequential file maintains the key characteristic of the sequential file: records are organized in sequence based on a key field.

- **Two features are added:** an index to the file to support random access, and an overflow file. The index provides a lookup capability to reach quickly the vicinity of a desired record. The overflow file is similar to the log file used with a sequential file but is integrated so that a record in the overflow file is located by following a pointer from its predecessor record.

The Indexed Sequential File

- In the simplest indexed sequential structure, a single level of indexing is used. The index in this case is a simple sequential file.

- Each record in the index file consists of two fields: a key field, which is the same as the key field in the main file, and a pointer into the main file. To find a specific field, the index is searched to find the highest key value that is equal to or precedes the desired key value. The search continues in the main file at the location indicated by the pointer.

The Indexed Sequential File

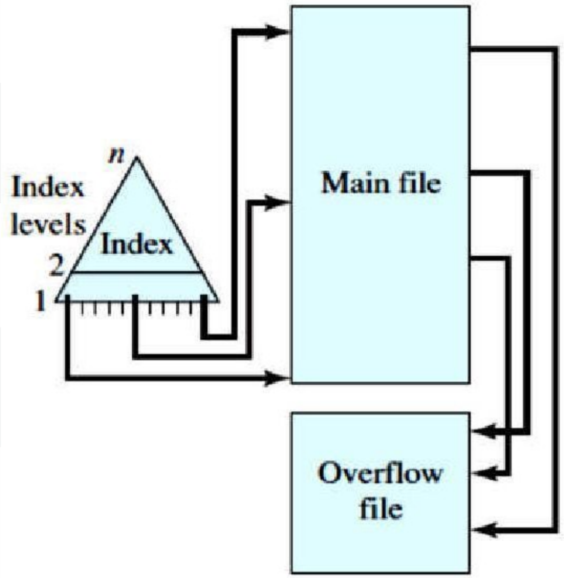


Image source : Book

The Direct or Hashed File

- The direct, or hashed, file exploits the capability found on disks to access directly any block of a known address.

- As with sequential and indexed sequential files, a key field is required in each record. However, there is no concept of sequential ordering here. The direct file makes use of hashing on the key value.

- Direct files are often used where very rapid access is required, where fixed length records are used, and where records are always accessed one at a time. Examples are directories, pricing tables, schedules, and name lists.

File Sharing

- File sharing is the practice of sharing or offering access to digital information or resources, including documents, multimedia (audio/video), graphics, computer programs, images and e-books.

- It is the private or public distribution of data or resources in a network with different levels of sharing privileges.

Secondary Storage

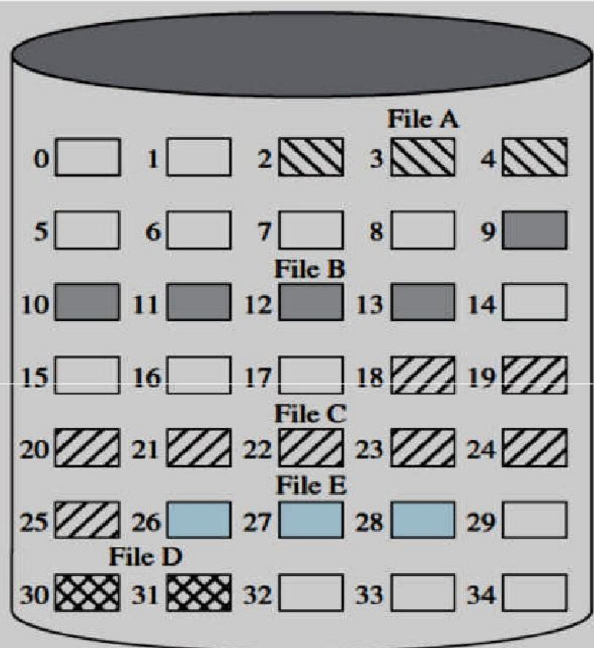
- It is all data storage that is not currently in a computer's primary storage or memory. In a personal computer, secondary storage typically consists of storage on the hard disk and on any removable media, if present, such as a CD or DVD.

- There are three methods are commonly used for file allocation:
  1. Contiguous
  2. Chained
  3. Indexed

Contiguous Allocation Method

- In this method a single contiguous set of blocks is allocated to a file at the time of file creation. Thus, this is a pre-allocation strategy, using variable-size portions.
- The file allocation table needs just a single entry for each file, showing the starting block and the length of the file.
- Contiguous allocation is the best from the point of view of the individual sequential file.
- Multiple blocks can be read in at a time to improve I/O performance for sequential processing. It is also easy to retrieve a single block.

Contiguous Allocation Method



File Allocation Table

File Name	Start Block	Length
File A	2	3
File B	9	5
File C	18	8
File D	30	2
File E	26	3

Image source : Book

Chained Allocation Method

- Allocation is on an individual block basis.
- Each block contains a pointer to the next block in the chain.
- Again, the file allocation table needs just a single entry for each file, showing the starting block and the length of the file.
- Although pre-allocation is possible, it is more common simply to allocate blocks as needed.
- The selection of blocks is now a simple matter: any free block can be added to a chain.
- There is no external fragmentation to worry about because only one block at a time is needed.
- This type of physical organization is best suited to sequential files that are to be processed sequentially. To select an individual block of a file requires tracing through the chain to the desired block.

Chained Allocation Method

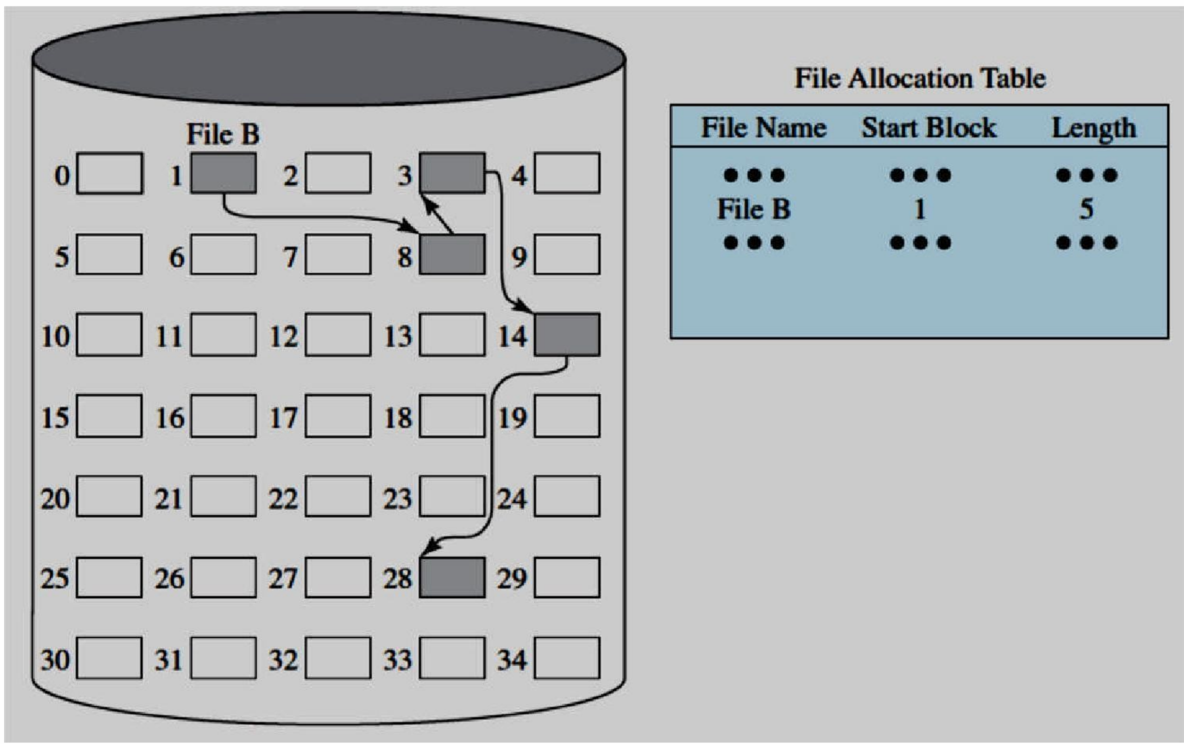


Image source : Book

Indexed Allocation Method

- Indexed allocation addresses many of the problems of contiguous and chained allocation. In this case, the file allocation table contains a separate one-level index for each file; the index has one entry for each portion allocated to the file.
- The file indexes are not physically stored as part of the file allocation table.
- Rather, the file index for a file is kept in a separate block and the entry for the file in the file allocation table points to that block.
- Allocation may be on the basis of either fixed-size or variable-size.

Indexed Allocation Method

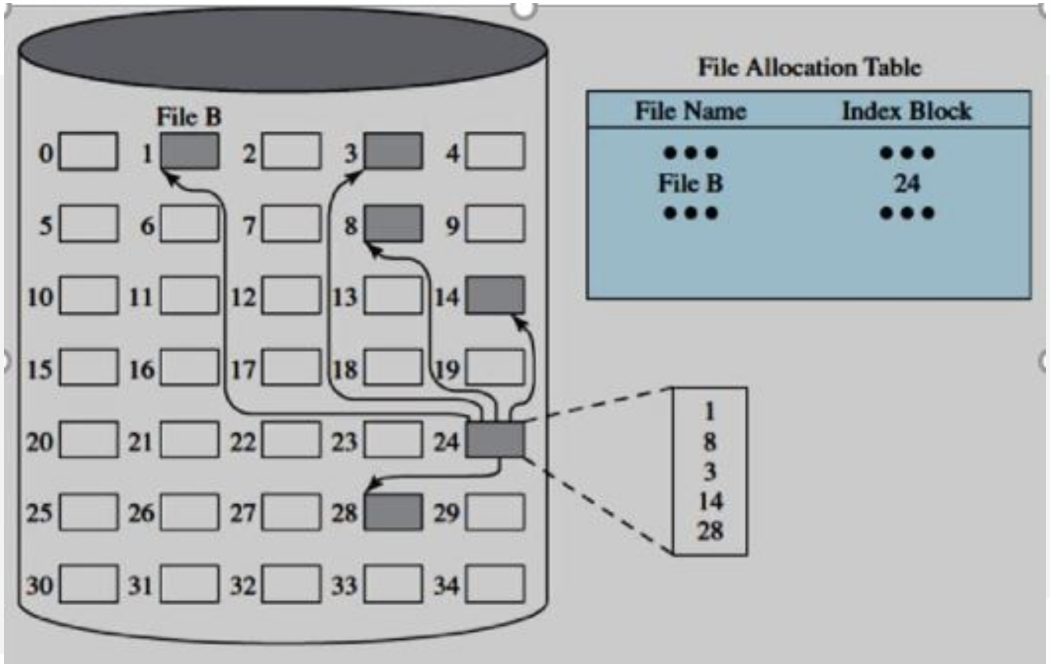
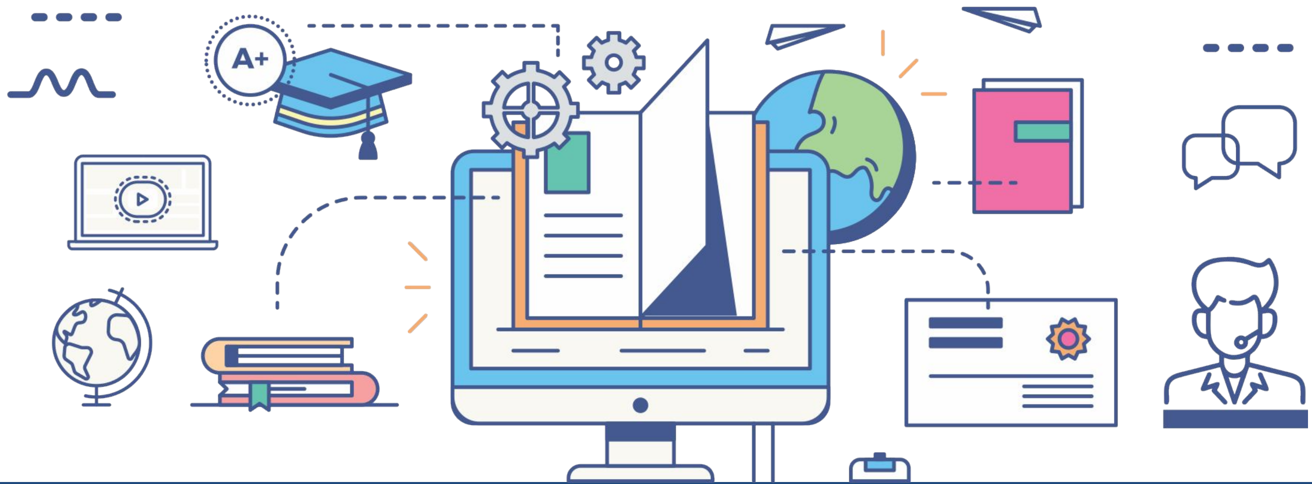


Image source : Book

DIGITAL LEARNING CONTENT



Parul® University

www.paruluniversity.ac.in