Chapter 14

Multiple Choice Questions

1. The following characteristics of disks make them convenient for being the secondary storage:
A. it is possible to read a block from the disk, modify the block, and write it back into the same place
B. a disk can access directly any block of information it contains
C. all of the above
D. none of the above

Ans: C
Feedback: 14.1
Difficulty: Easy

2. The basic file systems _______________
A. reads and writes physical blocks on the storage device.
B. tracks unallocated blocks and provides them the when it is required.
C. manages directory structure.
D. is responsible for protection.

Ans: A
Feedback: 14.1
Difficulty: Medium

2. _______________ is used to implement a file system.
A. A boot control block
B. A volume control block
C. A directory structure.
D. all of the above

Ans: D
Feedback: 14.2
Difficulty: Easy

3. UNIX inode is an example of _______
A. a contiguous allocation.
B. a linked allocation
C. an indexed allocation
D. an index file

Ans: C
Feedback: 14.4.3
Difficulty: Easy

4. What is the size of the bit vector of a 1TB disk with 512-byte blocks?
A. 2 MB
B. 2^8 MB
C. 28 MB
D. 8 MB

Ans: B
Feedback: 14.5.1
Difficulty: Easy

5. The FAT method ______________
A. keeps information about the block where bit vector is stored.
B. employs space maps to manage information about free blocks.
C. does not store information about free blocks.
D. incorporates free-block accounting into the allocation data structure.

Ans: C
Feedback: 14.4.3
Difficulty: Easy

6. Using ____-bit pointers limits the size of a file to 4 GB.
A. 32
B. 64
C. 128
D. 256

Ans: A
Feedback: 14.6.1
Difficulty: Easy

7. In __________ write, the data are stored in the cache, and control returns to the caller.
A. a non-buffered
B. a buffered
C. an asynchronous
D. a synchronous

Ans: C
Feedback: 14.6.2
Difficulty: Medium

8. Indexed allocation
A. supports direct access
B. does not suffer from external fragmentation.
C. all of the above
D. none of the above

Ans: C
Feedback: 14.4.3
Difficulty: Easy

9. Unified virtual memory uses ____________ to cache both process page and file data
A. disk block caching
B. double caching
C. buffer caching
D. page caching

Ans: D
Feedback: 14.6.2
Difficulty: Hard

10. Sequential access can be optimized by
A. free-behind technology
B. read-ahead technology
C. all of the above
D. none of the above

Ans: C
Feedback: 14.6.2
Difficulty: Medium

11. WAFL (*write-anywhere file layout*)
A. is a file system dedicated to single user operating systems
B. is a distributed file system
C. cannot provide files via NFS
D. can provide files via NFS, but not via CIFS

Ans: B
Feedback: 14.8
Difficulty: Medium

Essay Questions

1. Describe the role of the logical file system module.
Ans: The logical file system manages metadata information, which includes all of the file-system structure except the actual data (or contents of the files). The logical file system manages the directory structure to provide the file-organization module with the information the latter needs, given a symbolic file name. It maintains file structure via file-control blocks.

Feedback: 14.1
Difficulty: Medium

2. When a boot control block may be empty?
Ans: It could be empty if the disk does not contain an operating system.

Feedback: 14.2
Difficulty: Easy

3. What kind of in-memory information are used for file-system management?
Ans: We may enumerate several structures, as: (i) an in-memory mount table, (ii) an in-memory directory structure cache, (iii) the system-wide open-file table, (iv) the per-process open-file table, (v) buffers hold file-system blocks.

Feedback: 14.2.1
Difficulty: Easy

4. What happens when a process closes the file?
Ans: When a process closes the file, the per-process table entry is removed, and the system-wide entry's open count is decremented. When all users that have opened the file close it, any updated metadata are copied back to the disk-based directory structure, and the system-wide open-file table entry is removed.

Feedback: 14.2.1
Difficulty: Easy

5. What are the main drawback of a linear list of directory entries?
Ans: The real disadvantage of a linear list of directory entries is that finding a file requires a linear search. Directory information is used frequently, and users will notice if access to it is slow.

Feedback: 14.3.1
Difficulty: Easy

6. How may we combat the main disadvantage of fixed size hash table?
Ans: The major difficulties with a hash table are its generally fixed size and the dependence of the hash function on that size. We can use a chained-overflow hash table. Each hash entry can be a linked list instead of an individual value, and we can resolve collisions by adding the new entry to the linked list. Lookups may be somewhat slowed, because searching for a name might require stepping through a linked list of colliding table entries.

Feedback: 14.3.2
Difficulty: Hard

7. Why is the whole block not available to a user when linked allocation is used?
Ans: Because each block contains a pointer to the next block. These pointers are not made available to the user. Thus, if each block is 512 bytes in size, and a block address (the pointer) requires 4 bytes, then the user sees blocks of 508 bytes.

Feedback: 14.4.2
Difficulty: Medium

8. Describe UNIX inode concept.
Ans: File’s inode is a data structure in a UNIX file system that describes a file or a directory. Each inode stores the attributes and disk block location(s) of the object's data by keeping, say, 15 pointers of the index block. The first 12 of these pointers point to direct blocks; that is, they contain addresses of blocks that contain data of the file. Thus, the data for small files (of no more than 12 blocks) do not need a separate index block. If the block size is 4 KB, then up to 48 KB of data can be accessed directly. The next three pointers point to indirect blocks. The first points to a single indirect block, which is an index block containing not data but the addresses of blocks that do contain data. The second points to a double indirect block, which contains the address of a block that contains the addresses of blocks that contain pointers to the actual data blocks. The last pointer contains the address of a triple indirect block.

Feedback: 14.4.3
Difficulty: Medium

9. Why should new allocation algorithms be developed for NVM (nonvolatile memory) devices?
Ans: Because there are no disk head seeks for them, so different algorithms and optimizations are needed. Using an old algorithm that spends many CPU cycles trying to avoid a nonexistent head movement would be very inefficient. Existing file systems are being modified and new ones are being created to attain maximum performance from NVM storage devices. These developments aim to reduce the instruction count and overall path between the storage device and application access to the data.

Feedback: 14.4.4
Difficulty: Medium

10. What method of free space management is used by Oracle's ZFS file system?
Ans: ZFS uses a combination of techniques to control the size of data structures and minimize the I/O needed to manage those structures. First, ZFS creates metaslabs to divide the space on the device into chunks of manageable size. A given volume may contain hundreds of metaslabs. Each metaslab has an associated space map. ZFS uses the counting algorithm to store information about free blocks. Rather than write counting structures to disk, it uses log-structured file-system techniques to record them. The space map is a log of all block activity (allocating and freeing), in time order, in counting format. When ZFS decides to allocate or free space from a metaslab, it loads the associated space map into memory in a balanced-tree structure (for very efficient operation), indexed by offset, and replays the log into that structure. The in-memory space map is then an accurate representation of the allocated and free space in the metaslab. ZFS also condenses the map as much as possible by combining contiguous free blocks into a single entry. Finally, the free-space list is updated on disk as part of the transaction-oriented operations of ZFS. During the collection and sorting phase, block requests can still occur, and ZFS satisfies these requests from the log.

Feedback: 14.5.5
Difficulty: Hard

11. What is a double caching?
Ans: The memory-mapping call, however, requires using two caches - the page cache and the buffer cache. A memory mapping proceeds by reading in disk blocks from the file system and storing them in the buffer cache. Because the virtual memory system does not interface with the buffer cache, the contents of the file in the buffer cache must be copied into the page cache. This situation, known as double caching, requires caching file-system data twice. Not only does this waste memory but it also wastes significant CPU and I/O cycles due to the extra data movement within system memory.

Feedback: 14.6.2
Difficulty: Hard

12. What is a role of fsck in UNIX?
Ans: It is a systems program known as the consistency checker, which compares the data in the directory structure with the data blocks on storage and tries to fix any inconsistencies it finds.

Feedback: 14.7.1
Difficulty: Medium

13. Why should the permanent backup be stored far away from the regular backup?
Ans: The permanent backups should be stored far away from the regular backups to protect against hazards, such as fire, that can destroy the computer and all the backups.

Feedback: 14.7.4
Difficulty: Easy

True/False Questions

1. The I/O control level consists of device drivers and interrupt handlers to transfer information between the main memory and the disk system

Ans: T
Feedback: 14.1
Difficulty: Easy

2. The logical file system module includes the free-space manager.

Ans: F
Feedback: 14.1
Difficulty: Medium

3. The open() call returns a pointer to the appropriate entry in the system wide-open-file table.

Ans: F
Feedback: 14.2.2
Difficulty: Medium

4. Contiguous allocation of a file is defined by the address of the first block and length (in block units) of the file.

Ans: T
Feedback: 14.4.1
Difficulty: Medium

5. The file allocation table (FAT) has one entry for each block and is indexed by block number.

Ans: T
Feedback: 14.4.2
Difficulty: Easy

6. The FAT method incorporates free-block accounting into the allocation data structure.

Ans: T
Feedback: 14.5.2
Difficulty: Easy

7. The addresses of a given number of free blocks can be found faster using linked list rather than using grouping.

Ans: F
Feedback: 14.5.3
Difficulty: Medium

8. For counting method (used by free space management), the entries in the free-space list can be stored in a linked list, rather than a balanced tree, for efficient lookup, insertion, and deletion.

Ans: F
Feedback: 14.5.4
Difficulty: Medium

9. Storage devices that do not allow overwrite (such as NVM devices) need only the free list for managing free space.

Ans: F
Feedback: 14.5.6
Difficulty: Hard

10. The consistency check is always able to recover the structures, e.g., resulting in loss of files and entire directories.

Ans: F
Feedback: 14.7.2
Difficulty: Medium

11. Solaris ZFS file system never overwrites blocks with new data.
Ans: F
Feedback: 14.7.3
Difficulty: Medium