



COSC 6360 FIRST QUIZ

COSC 6360
Fall 2019



White Quiz



First question

- Which UNIX shell construct lets users combine two or more UNIX utilities to perform a single task?



First question

- Which UNIX shell construct lets users combine two or more UNIX utilities to perform a single task?
 - **Pipes (as in “ls -alg | more”).**



Second question

- What is the purpose of Unix/Linux ***symbolic links***?



Second question

- What is the purpose of Unix/Linux ***symbolic links***?
 - **To let a directory entry point to a file or a directory that is located in a different partition.**



Third question

- Where do Unix/Linux file systems store ***file names***?



Third question

- Where do Unix/Linux file systems store *file names*?
 - In the directory entry/entries pointing to the file.



Fourth question

- What does an FFS *cylinder group* contain
- How do these cylinder groups improve the performance of the file system?



Fourth question

- What does an FFS *cylinder group* contain
 - **Both i-nodes and data blocks.**
- How do these cylinder groups improve the performance of the file system?
 - **They let the data blocks of a file reside on disk closer to its i-node thus eliminating long seeks.**



Fifth question

- When we create a new file, in which order should we write to disk
 - (a) The i-node of the new file and
 - (b) The directory entry that points to it?



Fifth question

- When we create a new file, in which order should we write to disk
 - (a) The i-node of the new file and
 - (b) The directory entry that points to it?
- **We should write first the i-node of the new file then the directory entry pointing to it.**



Sixth question

- When can we safely delete log entries in a ***journaling file system***?



Sixth question

- When can we safely delete log entries in a *journaling file system*?
- **Once the metadata update has been written at its proper location in the file system.**



Seventh question

- The current FreeBSD scheduler divides its tasks between a short-term scheduler that is executed every time a core is released and a long-term scheduler that is executed once every second. What is the main advantage of this approach?



Seventh question

- The current FreeBSD scheduler divides its tasks between a short-term scheduler that is executed every time a core is released and a long-term scheduler that is executed once every second. What is the main advantage of this approach?
 - **It keeps the short-term scheduler very simple.**



Eighth question Part A

- How does the FreeBSD page replacement policy favor pages that have been frequently referenced?



Eighth question Part A

- How does the FreeBSD page replacement policy favor pages that have been frequently referenced?
 - **Free BSD uses a modified Clock policy that takes into account the *usage count* of a page when deciding whether to expel it.**



Eighth question Part B

- Would this approach work on any hardware?



Eighth question Part B

- Would this approach work on any hardware?
 - **No, it requires a processor having a *page-referenced bit* or *access bit* as the hand of the clock moves much faster than in the two-handed version of the BSD Clock policy.**



Yellow quiz



First question

- What are Linux/UNIX *special files*?



First question

- What are Linux/UNIX *special files*?
 - **Special files are hardware devices and other non-file entities that can be accessed through the standard file I/O interface.**



Second question

- Why did FFS introduce ***block fragments***?



Second question

- Why did FFS introduce ***block fragments***?
 - **FFS introduced block fragments to *reduce internal fragmentation in the file system, especially when the file system contained many very small file.***



Third question

- How do UNIX file systems implement ***access control lists***?

- What is the main advantage of this approach?



Third question

- How do UNIX file systems implement ***access control lists***?
 - **Nine bits specify the read, write and execute permissions for the owner of the file, the file group, and all other users**
- What is the main advantage of this approach?
 - **The access control list is *small enough to fit in the file i-node***



Fourth question

- Why should we *replicate superblocks*?



Fourth question

- Why should we *replicate superblocks*?
 - **Because losing the superblock of a disk partition would render the whole partition unreadable.**



Fifth question

- When we delete a file, in which order should we write to disk
 - (a) The new value of the deleted file i-node and
 - (b) The directory entry that pointed to it?

- **We should overwrite first the directory entry that pointed to the file before reclaiming the file i-node.**



Fifth question

- When we delete a file, in which order should we write to disk
 - (a) The new value of the deleted file i-node and
 - (b) The directory entry that pointed to it?



Sixth question

- What is the main advantage of ***synchronous journaling file systems***?



Sixth question

- What is the main advantage of ***synchronous journaling file systems***?
 - It guarantees the *durability of metadata updates*.
 - *No updates will ever be lost after a crash.*



Seventh question

- The current FreeBSD scheduler divides its tasks between a short-term scheduler that is executed every time a core is released and a long-term scheduler that is executed once every second. What is the main advantage of this approach?



Seventh question

- The current FreeBSD scheduler divides its tasks between a short-term scheduler that is executed every time a core is released and a long-term scheduler that is executed once every second. What is the main advantage of this approach?
 - **It keeps the short-term scheduler very simple.**



Eighth question Part A

- Why did the UNIX page replacement policy of the mid-eighties use two hands?



Eighth question Part A

- Why did the UNIX page replacement policy of the mid-eighties use two hands?
 - **That policy simulated by software a missing page-referenced bit, which caused two context switches each time a page with its simulated page-referenced bit equal to zero was accessed again, and had to minimize the linear speed of the clock hand.**



Eighth question Part B

- Why did this second hand disappear in the current FreeBSD page replacement policy?



Eighth question Part B

- Why did this second hand disappear in the current FreeBSD page replacement policy?
 - **Since FreeBSD runs on an architecture having a page-referenced bit, the linear speed of the clock hand is not anymore an issue.**