SOLUTIONS FOR THE THIRD 4330 QUIZ

Jehan-Francois Paris Summer 2014

First question

- A 32-bit virtual memory system has 8KB pages.
 - How many bits of the address are used by the byte offset
 - How many bits of the address are used by the page number?
 - How many pages are there in a process address space?

- A 32-bit virtual memory system has 8KB pages.
 - How many bits of the address are used by the byte offset?

log₂ 8K = 13 bits

How many bits of the address are used by the page number?

■ 32 - 13 = 19 **bits**

How many pages are there in a process address space?

Second question

- Which of the following properties apply to these page replacement policies:
 - □ Handles fairly well real-time processes.
 - □ Has an unacceptably high overhead.
 - □ Occasions additional context switches.
 - □ Performs very poorly.
 - □ Used by a popular operating system.
- Policies include Local LRU, Local FIFO, Global LRU, Global FIFO, BSD Clock, Mach, Windows

Which of the following properties apply to these page replacement policies: □ Handles fairly well real-time processes. Windows □ Has an unacceptably high overhead. Local LRU, Global LRU Occasions additional context switches BSD Clock, Mach, Windows □ Performs very poorly. Local FIFO, Global FIFO \Box Used by a popular operating system. BSD Clock, Mach, Windows

Third question

- A 32-bit FFS file system has a block size of 8 kilobytes. How many bytes—not blocks—of a given file can be accessed :
 - Directly from the i-node?
 - □ With one level of indirection?
 - □ With two levels of indirection?





Block size = 8KB

Up to 4M double indirect blocks but file size cannot exceed 4GB

- A 32-bit FFS file system has a block size of 8 kilobytes. How many bytes—not blocks—of a given file can be accessed :
 - □ Directly from the i-node?

```
12×8K = 96K bytes
```

With one level of indirection?

(8K/4)×8K = 16M_bytes

□ With two levels of indirection?

4G - 16M - 96K bytes

 Describe the contents of a UNIX *directory* entry.

 Describe the contents of a UNIX *directory entry*.

□ A UNIX directory entry contains a name and an i-node number.

Why does Berkeley FFS use blocking writes for all metadata updates?

Why does Berkeley FFS use blocking writes for all metadata updates?

To guarantee the durability of these updates and the consistency of the file system in the presence of system crashes.

How can we prevent deadlocks by denying the hold and wait condition?

How can we prevent deadlocks by denying the hold and wait condition?

We should impose an all -or-nothing resource allocation scheme that does not allow processes to hold resources while waiting for more resources

What is the purpose of the valid bit in a virtual memory system?

What is the purpose of the valid bit in a virtual memory system?

□ It tells whether the page table entry is valid and the page is in main memory or it is invalid and the page is missing.

What does the Berkeley Fast File System do to reduce *disk seeks*?

- What does the Berkeley Fast File System do to reduce *disk seeks*?
 - It organizes each disk partition into cylinder groups that contain a fragment of the i-node table and enough data blocks to let most files reside in the same cylinder group as their i-node.

• What is the *main advantage* of *mapped files*?

- What is the main advantage of mapped files?
 - Mapped files transfers the data blocks that a process accesses directly into the address space of the process, thus eliminating the context switches that would otherwise be required to bring them from the I/O buffer.

Fifth question

What is the function of the /etc/group file in the UNIX file system?

What would happen in an intruder could modify that file?

- What is the function of the /etc/group file in the UNIX file system?
 - The /etc/group contains a list of all protection groups on the system with their memberships.
- What would happen in an intruder could modify that file?
 - The intruder could add the account he or she—is using to each group and gain group access to all files on the system.

Sixth question

What is *internal fragmentation*? (5 points)

Why is it a more important issue in file systems than in virtual memory systems?

• What is *internal fragmentation*?

Fragmentation occurs because each process must occupy a fixed number of pages and each file must occupy a fixed number of blocks. Hence the last file page and the last file block are almost never full.

Why is it a more important issue in file systems than in virtual memory systems?

Many file systems have numerous files whose size can be a fraction of the block size of the file system while even the smallest process will contain several pages.

Seventh question

Explain how hashed page tables work.



PN = page number PFN = page frame number The table only contains the pages that reside in main memory.

More explanations

- As hashed page tables only contain the pages that currently reside in main memory, they are accessed through a hash function.
- Each hash table entry contains a pointer to the head of a—possibly empty—linked list containing the page table entries corresponding to a given hashed value of their page number.