CS 202: Advanced Operating Systems, Fall 2021

85

Final Exam

Name & NetID (login ID) (2 points): 2

Chirag Rajarat craja008

I. True and False (2 points each)

36

1. (T) Lock ranking can solve the deadlock problem.

2. (T) With Read-Copy Update (RCU), multiple writers can update shared data in parallel.

3. (F) If the system has infinite memory, Read-Copy Update (RCU) can replace locking-based synchronization for any type of shared resources.

4. (F) In multi-core OSs, message passing may underperform shared memory when the number of cores is relatively small.

5. (T) Sloppy counters in the Linux Scalability paper help reduce cache coherence traffic.

6. (T) Larger block size of Fast file system (FFS) help improve throughput.

7. (T) Fast file system (FFS) can use sub-blocks to address the internal fragmentation problem.

8. (F) In Log-structure File System (LFS), inode-maps are stored in the superblock.

9. (F) LFS cannot cache Checkpoint Regions because they need to be up-to-date on the disk.

10. (F) Google file system (GFS) requires multiple lookups to reach a file in a directory.

11. (F) Xen uses dynamic binary translation for x86 architectures.

12. (F) Xen requires modifications in both guest OS and user-level programs.

13. (F) Xen mimics the monolithic kernel design.

14. (T) Shadow paging does not require any modification to the guest OS kernel.

15. (T) Despite its overhead for construction and management, shadow paging may be faster in address translation than nested/extended paging.

16. (F) Hardware extension for memory virtualization has more spatial overhead than shadow paging.

17. (F) TinyOS provides multithreading on top of the event-driven kernel.

18. (F) Tock uses a memory protection unit (MPU) to isolate capsules.

19. (T) Processes in Tock OS are preemptively scheduled.

20. (F) In Tock, processes must be written in the type-safe Rust language

II. Short answers (4 points each)

8

21. Here is pseudocode of a simple spinlock implementation using test-and-set:

```
struct lock {
    int held = 0;
}
void acquire (lock) {
    while (test-and-set(&lock->held) == 1);
}
void release (lock) {
    lock->held = 0;
}
```

This code achieves mutual exclusion, but what are the two potential problems of this lock implementation? 2

① If a lock is held, another process which need access is stuck in while loop, until the lock is released. This causes waste of cpu cycles.

② If the critical section is big, other process might have to wait longer for lock, the spin will be longer.

③ The process which helds the lock can get interrupted.

22. Consider a hypothetical hard disk drive with 3000 RPM, 100MB/s of data transfer rate, and 10 ms of seek time. If the file system reads data from the disk in a block size of 2MB per access, the latency is [ ] ms and the throughput is [ ] MB/s. Give your answers for the blanks.

0

$T_{DISK\ I/O} = T_{Seek} + T_{Rotation} + T_{Transfer}$, $T_{seek} = 10ms$

$T_{ROTATION} = (1/3000/60/1000))/2 = 10$

$T_{Transfer} = \frac{2}{(100\,MB/1000\,ms/s)} = 0.015625$

$\Rightarrow$ Latency $= 10+10+0.015625 = 20.015625ms$

Throughput =

2 MB / 20.015 / 1000 s

= 99.92 MB/s

23. Consider a virtualization scenario where the hypervisor and the guest OS uses 4-level paging. If nested/extended paging is used, what is the number of memory accesses required to translate a guest virtual address to a host physical address? Also, what is the number for shadow paging?

Memory access required for GVA to HPA = 5×5 = 25

2

Number of shadow paging = 5

24. Why event-driven systems can be more memory efficient than multi-threaded systems?

4

Threads have different address spaces for user level threads & kernel level threads.
Thus, multi-threaded systems have to maintain both address spaces for a thread.
As a result, event driven systems are more memory efficient

III. Single Choice (3 points each)

39

25. Which of the following is true about Multikernel and Barrelfish?

× A. Multikernel OS views kernel states as 'shared' instead of 'replicated'
B. Barrelfish is an implementation of Multikernel for heterogeneous distributed systems
C. On each core, multiple CPU drivers are preemptively scheduled to handle traps and exceptions
[✓ D.] In Barrelfish, user-level RPC (URPC) uses cache-line-sized messages.

26. Which of the following can be bad for OS scalability? **[CEP1]**

[✓ A.] Minimize the number of locks by merging critical sections
× B. Use per-core data structures if possible
C. Use lock-free data structures for read-mostly data
D. Place frequently-updated data on separate cache lines

27. Find a statement that does **NOT** correctly discuss interrupts and polling for I/O handling. **[CEP2]**

× A. Polling can be more efficient than interrupts for short I/O operations

× B. Polling uses CPU only when the I/O operation is complete

[✓ C.] Polling typically results in smaller cache-miss rates than interrupts

D. Using interrupts with DMA can reduce the power consumption of the processor

28. Choose the correct statement about the original UNIX file system (UFS).

× A. UFS stores checkpoint regions within the super block.

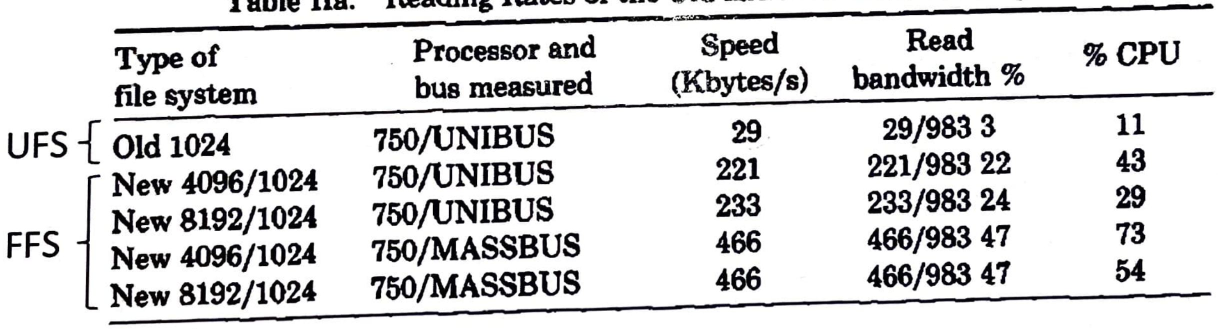
[✓ B.] UFS stores the metadata of each file in an inode.

C. UFS metadata keeps the information regarding the content type of each file (e.g. image, text, program code)

D. UFS has more internal fragmentation than the Fast File System (FFS).

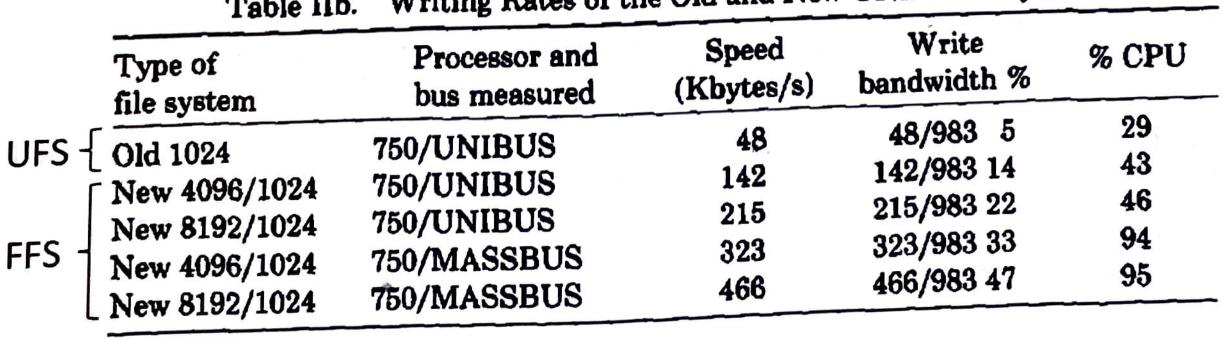
29. Below compares the performance of FFS and UFS. Which of the following is a correct observation about these results? **[CEP3]**

Table IIa. Reading Rates of the Old and New UNIX File Systems



	Type of file system	Processor and bus measured	Speed (Kbytes/s)	Read bandwidth %	% CPU
UFS	Old 1024	750/UNIBUS	29	29/983 3	11
FFS	New 4096/1024	750/UNIBUS	221	221/983 22	43
	New 8192/1024	750/UNIBUS	233	233/983 24	29
	New 4096/1024	750/MASSBUS	466	466/983 47	73
	New 8192/1024	750/MASSBUS	466	466/983 47	54

Table IIb. Writing Rates of the Old and New UNIX File Systems



	Type of file system	Processor and bus measured	Speed (Kbytes/s)	Write bandwidth %	% CPU
UFS	Old 1024	750/UNIBUS	48	48/983 5	29
FFS	New 4096/1024	750/UNIBUS	142	142/983 14	43
	New 8192/1024	750/UNIBUS	215	215/983 22	46
	New 4096/1024	750/MASSBUS	323	323/983 33	94
	New 8192/1024	750/MASSBUS	466	466/983 47	95

× A. UFS outperforms FFS for both reads and writes

× B. Reading data is generally faster than writing in UFS

[✓ C.] The bandwidth utilization of FFS is better than that of UFS

D. The CPU usage of UFS is higher than FFS

30. Which of the following factor(s) of disk access can be improved by the use of Cylinder Groups in the Fast File System (FFS)?

A. Seek time
B. Rotation time
C. Data transfer time
[✓ D. Seek time & Rotation time]
E. Rotation time & Data transfer time

31. Which of the following is the major improvement that the Log-structured File System (LFS) achieves over FFS?

✗ A. Efficiency of large file reads
B. Lower space overhead of metadata in the storage device
[✓ C. Better performance on small random writes]
D. Lower complexity of in-memory data structures used by the file system
E. Spatial locality of data on disk

32. The Network File System (NFS) uses a stateless protocol. Why? **[CEP4]**

[✓ A. Easier to recover from server crashes]
✗ B. Easier to guarantee file consistency
✗ C. To reduce network traffic
✓ D. To improve data locality

33. Which of the following matches the optimization goal of the Google File System (GFS)?

✗ A. Optimize for small files
B. Optimize for future generation storage devices
✗ C. Optimize for random writes
✗ D. Optimize for access latency
[✓ E. Optimize for throughput]
✗ F. Optimize for low external fragmentation of free space

34. Regarding the GFS architecture, which of the following is correct?

A. The master server caches data blocks from chunkservers.
[✓ B. The same data block can be stored in multiple chunkservers.]
C. Clients can cache metadata as well as the actual data blocks to improve performance
D. Shadow masters provide the same capabilities as the master when the master is down.

35. Choose the correct statement about the Domain0 of Xen.

A. It is the initial domain started by the Xen hypervisor at boot time and terminates once other domains start.
B. Domain0 runs at a higher privilege level than other virtual machines
[✓ C.] Domain0 runs all the device drivers for the host machine hardware.
D. Guest kernel in Domain0 does not need to be modified as it has control interface to the hypervisor

36. Which of the following is **NOT** offered by hardware virtualization extensions? **[CEP5]**

A. Host and guest modes (a.k.a. VMX root and VMX non-root)
[✓ B.] Binary translation acceleration
C. Nested/extended paging
D. DMA/interrupt remapping

37. Which of the following is **NOT** the optimization considered in TinyOS?

A. All TinyOS files are merged into a single C file at compile time
B. TinyOS does not allow dynamic memory allocation
C. TinyOS employs an event-driven architecture
[✓ D.] TinyOS supports updating part of the code at runtime
E. TinyOS puts CPU into sleep mode when the task queue is empty

38. Which of the following is **NOT** the correct explanation about the Grant regions in Tock OS?

A. Located in the application address space
B. Stores kernel states
C. Protected by a Memory Protection Unit (MPU)
[✓ D.] Can be shared with other processes
E. Can be created dynamically