# Advanced Operating System Quiz

**Name:_____ Sciper Number: _____ Score:_____**

## MCQs:（2 x 12 = 24）

1. What is the meaning of the following GCC inline assembly? **A**

```
asm("movl %ecx, %eax")
```

   a. Move content in register %ecx to %eax
   b. Move content in register %eax to %ecx
   c. None of above

2. When the paging mechanism is enabled, what kind of address does the userspace program use? **B**
   a. Physical address
   b. Virtual address
   c. External address
   d. None of above

3. Suppose DRAM has a capacity of 3 pages. There are no pages residing in DRAM at the beginning. Suppose the virtual page access sequence is 0,1,2,0,1,3,0,3,1,0,3. If we use the LRU replacement policy, how many page faults will be triggered? D
   a. 1
   b. 2
   c. 3
   d. 4

4. Let's assume L1 cache size is 32KB with cache line size of 64 Byte. And your page size is 4KB. What are the maximum and minimum TLB entries required to cover all the data in the L1 cache, respectively? A
   a. 2^9 and 2^3
   b. 2^12 and 2^6
   c. 2^9 and 2^9
   d. 2^3 and 2^3

5. If the pages are thrashing between DRAM and swap, which methods can reduce thrashing? B
   a. Increase the swap size
   b. Kill some processes
   c. Increase the priority of all userspace processes

6. Suppose a user-mode process attempts to write to a file on disk. What is the name of the mechanism that this process needs to invoke to accomplish that? B
   a. Trap
   b. System call
   c. Exception
   d. Non-maskable interrupt (NMI)

7. IPI is a special form of interrupt that: B
    a. Interrupts another CPU synchronously
    b. Interrupts another CPU asynchronously
    c. No OS uses it.
    d. Is specifically used for scheduling tasks.

8. Consider the scenario in which two processes run on an operating system that uses a multilevel feedback queue (MLFQ) scheduling algorithm with two priority queues. Processes in the highest priority queue have a time-slice of 5 ms, while the ones in the lowest priority queue have a time-slice of 10 ms. Processes A and B start at the highest priority queue, and process A is scheduled first. Process A runs for 5 ms and is then preempted by the operating system. Process B then runs for 2 ms and then yields. What is the next process that will be scheduled, and how long will its time-slice be? C
    a. Process A, with a 5 ms time-slice
    b. Process A with a 10 ms time-slice
    c. Process B with a 5 ms time-slice
    d. Process B with a 10 ms time-slice

9. How many child processes are created in the following code snippet? C

```
for (int i = 0; i < 3; ++i)
    fork();
```

    a. 4
    b. 6
    c. 7
    d. 8

10. Consider a storage device with 4 KB-sized blocks. Assume that the file system uses a multilevel inode data structure to track the data blocks of a file. The inode has 64 bytes of space to store pointers to data blocks, including a single indirect block, a double indirect block, and several direct blocks. What maximum file size can be stored in such a file system? C
    a. 64*4KB
    b. 8*4KB + 64*4KB
    c. (4KB/8)*4KB + (4KB/8)*(4KB*8)*4KB + 6*4KB
    d. (4KB/8)*(4KB*8)*4KB + 6*4KB

11. Consider a system with 10 Disks. For each of the following RAID levels, how many disks does a system get to use to store actual data for RAID0, RAID1, and RAID4 respectively? A
    a. 10, 5, 9
    b. 5, 10, 9
    c. 9, 10, 5
    d. 10, 10, 9

12. Which mechanism below is NOT used to ensure the crash consistency in a file system D
    a. Journaling
    b. Copy-on-Write
    c. Write-ahead-log
    d. Monitor

# Fill in the blanks: 6

13. The CPU realizes that it is running the kernel code by checking the _____**CR3**_____.
14. The OS relies on _____**Interrupts**_____ to take the control back from a userspace process and schedule other processes fairly.
15. One must allocate a separate trap table for every syscall. ___**False**___ (True/False)
16. The __**kernel/ interrupt handler**_____ is responsible for handling concurrent syscalls and interrupts by __**disabling intterrupt**_____ when processing an interrupt handler routine and __**enabling interrupt**_____ when done processing.

# Writing questions:

17. What is wrong with the following code: 6

```
1. #include <stdatomic.h>
2. #include <stdbool.h>
3.
4. typedef struct {
5.     atomic_int lock;
6. } spinlock_t;
7.
8. void spinlock_init(spinlock_t *s) {
9.     s->lock = 1;
10.  }
11.
12.  void spinlock_lock(spinlock_t *s) {
13.      while (atomic_compare_exchange_strong(&s->lock,
14.  &(int){0}, 1)) {
15.      }
16.  }
```

```
17.
18.  void spinlock_unlock(spinlock_t *s) {
19.      s->lock = 0;
20.  }
21.
22.  int main() {
23.      spinlock_t my_spinlock;
24.      spinlock_init(&my_spinlock);
25.
26.      // Code to test the spinlock would go here.
27.      spinlock_lock(&my_spinlock);
28.      // Critical section
29.      spinlock_unlock(&my_spinlock);
30.
31.      return 0;
32.  }
```

What are at least three issues in the code wrt correctness and fairness?

1. Init
2. Not starvation free
3. Convert address of 0

18. Which hardware mechanism gives applications an illusion of almost infinite memory? **4**

Virtual memory and page translation

19. What is the MMU and how does it work with page table and TLB for address translation?**6**

Virtual -> Physical translation
First check TLB and then check the page table
Load page table to TLB if necessary

20. A computer system has a 36-bit virtual address space with a page size of 8K, and 4 bytes per page table entry.
    a. How many pages are in the virtual address space?**2**

A 36 bit address can address 2^36 bytes in a byte addressable machine. Since the size of a page 8K bytes (2^13), the number of addressable pages is 2^36 / >2^13 = 2^23

    b. What is the maximum size of addressable physical memory in this system? **2**

With 4 byte entries in the page table we can reference 2^32 pages. Since each page is 2^13 B long, the maximum addressable physical memory size is 2^32 * 2^13 = 2^45 B (assuming no protection bits are used).

    c. If the average process size is 8GB, would you use a one-level, two-level, or three-level page table? Why? **4**

8 GB = 2^33 B
We need to analyze memory and time requirements of paging schemes in order to make a decision. Average process size is considered in the calculations below.

**1 Level Paging**

Since we have 2^23 pages in each virtual address space, and we use 4 bytes per page table entry, the size of the page table will be 2^23 * 2^2 = 2^25. This is 1/256 of the process' own memory space, so it is quite costly. (32 MB)

**2 Level Paging**

The address would be divided up as 12 | 11 | 13 since we want page table pages to fit into one page and we also want to divide the bits roughly equally.

Since the process' size is 8GB = 2^33 B, I assume what this means is that the total size of all the distinct pages that the process accesses is 2^33 B. Hence, this process accesses 2^33 / 2^13 = 2^20 pages. The bottom level of the page table then holds 2^20 references. We know the size of each bottom level chunk of the page table is 2^11 entries. So we need 2^20 / 2^11 = 2^9 of those bottom level chunks.

The total size of the page table is then:

| //size of the outer page table | | //total size of the inner pages | | |
|---|---|---|---|---|
| 1 * 2^12 * 4 | + | 2^9 * 2^11 * 4 | = | 2^20 * ( 2^-6 + 4) ~4MB |

**3 Level Paging**

For 3 level paging we can divide up the address as follows:

8 | 8 | 7 | 13

Again using the same reasoning as above we need 2^20/2^7 = 2^13 level 3 page table chunks. Each level 2 page table chunk references 2^8 level 3 page table chunks. So we need 2^13/2^8 = 2^5 level-2 tables. And, of course, one level-1 table.

The total size of the page table is then:

| //size of the outer page table | //total size of the level 2 tables | //total size of innermost tables | |
|---|---|---|---|
| 1 * 2^8 * 4 | 2^5 * 2^8 *4 | 2^13 * 2^7 * 4 | ~4MB |

As easily seen, 2-level and 3-level paging require much less space then level 1 paging scheme. And since our address space is not large enough, 3-level paging does not perform any better than 2 level paging. Due to the cost of memory accesses, choosing a 2 level paging scheme for this process is much more logical.