Final Problem Set

Started: May 4 at 5:48pm

Quiz Instructions

The final quiz will be worth 25% of the Final Project grade. Please upload your answers by **5th May, 11:59 AM**.

You have 10 attempts to add your answers. An offline version of the quiz is available **here**. You may use it to work on your questions offline. Once you have your answers ready, you may add them to the quiz one by one. Only your latest attempt will count. Your grade will be available after the due date.

Questions will have instructions on the format of the answer properly. Be sure to follow the guidelines in each question properly.

If you have any questions, post a *private* question on Piazza. If we think we can clarify anything, we will answer it and make it public.

Question 1 1 pts

Virtual Memory (1)

Suppose that you have a computer system that uses 16-bit virtual addresses, with 1KB pages sizes.

Suppose that you have a computer system that uses 16-bit virtual addresses, with 1KB pages sizes. How many bits do you need for the page offset?

10

Question 2 — 1 pts

Virtual Memory (2)

Suppose that you have a computer system that uses 16-bit virtual addresses, with 1KB pages sizes.

How many bits do you need for the virtual page number?

6

Question 3 — 1 pts

Virtual Memory (3)

Suppose that you have a computer system that uses 16-bit virtual addresses, with 1KB pages sizes.

How many distinct virtual pages can be represented in the address space of each process?

64

Question 4 — 11 pts

Virtual Memory (4)

Suppose that you have a computer system that uses 16-bit virtual addresses, with 1KB pages sizes.

Which of the following pieces of information would likely be stored within the page table of this address space? Write Y for Yes, and N for No next to each of the following:

Be sure to use exactly 'Y' and 'N' (case sensitive, no quotes!)

(a) Virtual page number; N

(b) Physical frame number; Y

(c) Access bit; Y

(d) Tag; N

(e) LRU bits; N

(f) Dirty bit; Y

(g) CR3; N

(h) Pointer to the root of the kernel's system call table: N

(i) Read/Write permission bit: Y

(j) User/kernel permission bit: Y

(k) Present bit: Y

Question 5

1 pts

Virtual Memory (5)

Suppose that you have a computer system that uses 16-bit virtual addresses, with 1KB pages sizes.

Now suppose that the physical address space is 4KB. How many distinct physical frames are supported in the address space?

4

Question 6 — 1 pts

Virtual Memory (6)

Suppose that you have a computer system that uses 16-bit virtual addresses, with 1KB pages sizes.

Suppose that you realize a flat page table to maintain the virtual to physical page mappings. Based on answers to the questions for part 1-5, how many bits would your page table need?

256

Question 7 — 1 pts

Virtual Memory (7)

All x86-64 CPUs maintain a register that acts as a proxy for the process ID. What is this register? Do not use a percentage or dollar sign, keep all letters in caps, and express the number in digits, rather than in written form (e.g., EAX, EBX, ESP, etc.).

CR3

Question 8 — 1 pts

Virtual Memory (8)

Now suppose that you have a machine that uses a 1024-bit virtual address space and a 128-bit physical address space. Suppose that each page table entry has 16 bytes. Suppose that each page is 64KB, and suppose that the page table is implemented as as a multi-level page table, similar to what you saw in class for x86-64 machines.

How many levels does this page table have? (you can assume all bits in virtual address are used in the translation process, unlike X86-64 which uses only 48 bits)

84

Question 9

1 pts

C Code Execution (1)

Assume that malloc does not use any header/footer structs when allocating memory and uses a separate data structure to keep track of free memory?

Suppose that we execute the following snippet of code:

```
Line 0: int *a = (int *)malloc(sizeof(int)*NUM_ELEMENTS);
Line 1: a[ 5 ] = 54;
Line 2: a[ 6 ] = 32;
```

For line 0, answer Yes or No for each of the following:

Be sure to use exactly 'Y' (for true) and 'N' (for false) (case sensitive, no quotes!)

(a) Line 0 may result in a system call. Y

(b) Line 0 will definitely result in changes to the page table. N

(c) Line 0 always triggers a major page fault. N

(d) Line 0 always triggers a minor page fault. N

(e) Line 0 always suffers a TLB miss. Y

Question 10

1 pts

C Code Execution (2)

Assume that malloc does not use any header/footer structs when allocating memory and uses a separate data structure to keep track of free memory?

Suppose that we execute the following snippet of code:

```
Line 0: int *a = (int *)malloc(sizeof(int)*NUM_ELEMENTS);
Line 1: a[ 5 ] = 54;
Line 2: a[ 6 ] = 32;
```

For line 1, answer Yes or No for each of the following:

Be sure to use exactly 'Y' (for true) and 'N' (for false) (case sensitive, no quotes!)

(f) Line 1 may result in a system call. Y

(g) Line 1 will definitely result in changes to the page table. N

(h) Line 1 always triggers a major page fault. N

(i) Line 1 always triggers a minor page fault. Y

(j) Line 1 always suffers a TLB miss. Y

(k) Line 1 will likely result in a cache miss. Y

Question 11 1 pts

C Code Execution (3)

Assume that malloc does not use any header/footer structs when allocating memory and uses a separate data structure to keep track of free memory?

Suppose that we execute the following snippet of code:

```
Line 0: int *a = (int *)malloc(sizeof(int)*NUM_ELEMENTS);
Line 1: a[ 5 ] = 54;
```

```
Line 2: a[ 6 ] = 32;
```

For line 2, answer Yes or No for each of the following:

Be sure to use exactly 'Y' (for true) and 'N' (for false) (case sensitive, no quotes!)

(l) Line 2 may result in a system call. N

(m) Line 2 will definitely result in changes to the page table. N

(n) Line 2 always triggers a major page fault. N

(o) Line 2 always triggers a minor page fault. N

(p) Line 2 always suffers a TLB miss. N

(q) Line 2 will likely result in a cache miss. N

Question 12

1 pts

Virtual Machine (1)

Suppose that you are running code on a virtualized environment. Suppose further that your guest virtual address space is 1024-bit, that your guest physical address space is 136-bit, and that your system physical address space is 64-bit. All page sizes are 64KB and page table entries are 16 bytes.

How many levels does the guest page table have? (you can assume all bits in virtual address are used in the translation process, unlike X86-64 which uses only 48 bits)

84

Question 13

1 pts

Virtual Machine (2)

Suppose that you are running code on a virtualized environment. Suppose further that your guest virtual address space is 1024-bit, that your guest physical address space is 136-bit, and that your system physical address space is 64-bit. All page sizes are 64KB and page table entries are 16 bytes.

How many page table levels are present in the nested page table? (you can assume all bits in virtual address are used in the translation process, unlike X86-64 which uses only 48 bits)

10

Question 14

1 pts

Virtual Machine (3)

Suppose that you are running code on a virtualized environment. Suppose further that your guest virtual address space is 1024-bit, that your guest physical address space is 136-bit, and that your system physical address space is 64-bit. All page sizes are 64KB and page table entries are 16 bytes.

How many memory references does a two-dimensional page table walk take? Assume that a memory reference can load 16 bytes at a time.

860

Question 15

1 pts

Caches (1)

Consider a cache for 128KB of data, storing 64B lines, that is 16-way set-associative. Suppose that the cache implements LRU replacement per set, and that each line needs a valid bit. Finally, suppose that the cache is write-back.

How many bits for the data and metadata (not including tags) does each cache line need?

518

Question 16

1 pts

Caches (2)

Consider a cache for 128KB of data, storing 64B lines, that is 16-way set-associative. Suppose that the cache implements LRU replacement per set, and that each line needs a valid bit. Finally, suppose that the cache is write-back.

Now suppose that the cache is being integrated into a system with 128-bit virtual addresses, 16KB pages, and 64-bit physical addresses. Suppose also that it is physically-addressed. How many bits are needed per tag?

51

Question 17

1 pts

Caches (3)

Consider a cache for 128KB of data, storing 64B lines, that is 16-way set-associative. Suppose that the cache implements LRU replacement per set, and that each line needs a valid bit. Finally, suppose that the cache is write-back. Suppose that the cache is being integrated into a system with 128-bit virtual addresses, 16KB pages, and 64-bit physical addresses. Suppose also that it is physically-addressed

How many SRAM bits are needed to realize this cache?

1,165,312

Question 18

1 pts

Caches (4)

Consider a cache for 128KB of data, storing 64B lines, that is 16-way set-associative. Suppose that the cache implements LRU replacement per set, and that each line needs a valid bit. Finally, suppose that the cache is write-back. Suppose that the cache is being integrated into a system with 128-bit virtual addresses, 16KB pages, and 64-bit physical addresses. Suppose also that it is physically-addressed. Suppose that you write a loop of code where you access array elements such that you force a cache miss for the cache above on every single memory reference. Assume that the loop has many billions of iterations.

What is the minimum number of memory references that you would need in such a loop, assuming **LRU** replacement?

17

Question 19

1 pts

Caches (5)

Consider a cache for 128KB of data, storing 64B lines, that is 16-way set-associative. Suppose that the cache implements LRU replacement per set, and that each line needs a valid bit. Finally, suppose that the cache is write-back. Suppose that the cache is being integrated into a system with 128-bit virtual addresses, 16KB pages, and 64-bit physical addresses. Suppose also that it is physically-addressed. Suppose that you write a loop of code where you access array elements such that you force a cache miss for the cache above on every single memory reference. Assume that the loop has many billions of iterations.

What is the minimum number of memory references that you would need in such a loop, assuming **FIFO** replacement?

16

Question 20

1 pts

Caches (6)

Consider a cache for 128KB of data, storing 64B lines, that is 16-way set-associative. Suppose that the cache implements LRU replacement per set, and that each line needs a valid bit. Finally, suppose that the cache is write-back. Suppose that the cache is being integrated into a system with 128-bit virtual addresses, 16KB pages, and 64-bit physical addresses. Suppose also that it is physically-addressed. Suppose that you write a loop of code where you access array elements such that you force a cache miss for the cache above on every single memory reference. Assume that the loop has many billions of iterations.

Which of the following is true about random cache line replacement for this cache?

- ○ The number of minimum memory references in a loop required to force a cache miss on every access is the same as part 4.
- ○ The number of minimum memory references in a loop required to force a cache miss on every access is 1 + the number in part 5. Assume that the loop has many billions of iterations.
- ○ Random replacement can never lead to a better hit rate than LRU.
- ○ Random replacement can never lead to a better hit rate than FIFO
- ◉ The minimum number of memory references in a loop required to force a cache miss on every access must be at least as the answer calculated in part 4 but could be higher. Assume that the loop has many billions of iterations.

Question 21

1 pts

Caches (7)

Now suppose that you have a two-level cache. Suppose that the cache introduced in (4) is an L1 cache, and the cache introduced in (5) is an L2 cache. Suppose that your L2 cache stores 2MB of data, has 64B lines, and is 128-way set-associative.

What is the minimum number of memory references that are needed in a loop to ensure that the cache access misses in both the L1 and the L2 cache? Assume LRU replacement and that the loop has many billions of iterations

145

Question 22

1 pts

Caches (8)

Now suppose that you have a two-level cache. Suppose that the cache introduced in (4) is an L1 cache, and the cache introduced in (5) is an L2 cache. Suppose that your L2 cache stores 2MB of data, has 64B lines, and is 128-way set-associative.

Now suppose that you want to build a TLB to perform fast virtual to physical address translation for the two-level cache hierarchy described above. Suppose that your TLB stores 512 entries, and is organized to be 8-way set associative. Suppose that you have a virtual memory system with the parameters detailed in Caches 2 (i.e. 128-bit virtual addresses, 16KB pages, and 64-bit physical addresses). How many bits do does each TLB entry's tag require?

1,024

Question 23

1 pts

TLB (1)

Now suppose that we have an architecture with the following parameters. Virtual

addresses that are 64 bits, page sizes that are 4KB, physical addresses that are 64 bits, page tables that are four-level. Suppose that you have a TLB that is 128 entries 4-way set associative, and one level of cache that is 4MB, 4-way set associative, with 1KB lines. Suppose that the TLB can be accessed in 1 cycle (i.e., this is the cost of a TLB hit), the cache can be accessed in 2 cycles (i.e., this is the cost of a cache hit), and that a cache miss has a miss penalty of 20 cycles (i.e., a cache access that results in a miss requires 2 cycles for lookup and a further 20 cycles to search memory, retrieve the desired cache line, and fill it into the cache). Moreover, based on the above, a TLB hit is 1 cycle, while a TLB miss is 1 cycle plus the time taken to find the page table entry. Suppose that both TLB and cache use LRU replacement.

Suppose that you make a memory reference for virtual address 0x1000000000008A01. What is the virtual page number that needs to be translated? Provide your answer in hexadecimal notation, beginning with "0x". Each hex character should be capitalized.

0x1000000000008

Question 24

1 pts

TLB (2)

Now suppose that we have an architecture with the following parameters. Virtual addresses that are 64 bits, page sizes that are 4KB, physical addresses that are 64 bits, page tables that are four-level. Suppose that you have a TLB that is 128 entries 4-way set associative, and one level of cache that is 4MB, 4-way set associative, with 1KB lines. Suppose that the TLB can be accessed in 1 cycle (i.e., this is the cost of a TLB hit), the cache can be accessed in 2 cycles (i.e., this is the cost of a cache hit), and that a cache miss has a miss penalty of 20 cycles (i.e., a cache access that results in a miss requires 2 cycles for lookup and a further 20 cycles to search memory, retrieve the desired cache line, and fill it into the cache). Moreover, based on the above, a TLB hit is 1 cycle, while a TLB miss is 1 cycle plus the time taken to find the page table entry. Suppose that both TLB and cache use LRU replacement. Suppose that you make a memory reference for virtual address 0x1000000000008A01.

What set number (in decimal number) in the TLB must be probed to look for this address?

8

Question 25

1 pts

TLB (3)

Now suppose that we have an architecture with the following parameters. Virtual addresses that are 64 bits, page sizes that are 4KB, physical addresses that are 64 bits, page tables that are four-level. Suppose that you have a TLB that is 128 entries 4-way set associative, and one level of cache that is 4MB, 4-way set associative, with 1KB lines. Suppose that the TLB can be accessed in 1 cycle (i.e., this is the cost of a TLB hit), the cache can be accessed in 2 cycles (i.e., this is the cost of a cache hit), and that a cache miss has a miss penalty of 20 cycles (i.e., a cache access that results in a miss requires 2 cycles for lookup and a further 20 cycles to search memory, retrieve the desired cache line, and fill it into the cache). Moreover, based on the above, a TLB hit is 1 cycle, while a TLB miss is 1 cycle plus the time taken to find the page table entry. Suppose that both TLB and cache use LRU replacement. Suppose that you make a memory reference for virtual address 0x1000000000008A01.

Suppose that at the time of this translation request, the TLBs and caches are empty. In other words, you suffer a TLB miss. How many cycles will be spent handling the TLB miss?

20

Question 26

1 pts

TLB (4)

Now suppose that we have an architecture with the following parameters. Virtual addresses that are 64 bits, page sizes that are 4KB, physical addresses that are 64 bits, page tables that are four-level. Suppose that you have a TLB that is 128 entries 4-way set associative, and one level of cache that is 4MB, 4-way set associative,

with 1KB lines. Suppose that the TLB can be accessed in 1 cycle (i.e., this is the cost of a TLB hit), the cache can be accessed in 2 cycles (i.e., this is the cost of a cache hit), and that a cache miss has a miss penalty of 20 cycles (i.e., a cache access that results in a miss requires 2 cycles for lookup and a further 20 cycles to search memory, retrieve the desired cache line, and fill it into the cache). Moreover, based on the above, a TLB hit is 1 cycle, while a TLB miss is 1 cycle plus the time taken to find the page table entry. Suppose that both TLB and cache use LRU replacement. Suppose that you make a memory reference for virtual address 0x1000000000008A01.

How many total cycles will it take to account for both the TLB miss and the replay?

20

Question 27

1 pts

TLB (5)

Now suppose that we have an architecture with the following parameters. Virtual addresses that are 64 bits, page sizes that are 4KB, physical addresses that are 64 bits, page tables that are four-level. Suppose that you have a TLB that is 128 entries 4-way set associative, and one level of cache that is 4MB, 4-way set associative, with 1KB lines. Suppose that the TLB can be accessed in 1 cycle (i.e., this is the cost of a TLB hit), the cache can be accessed in 2 cycles (i.e., this is the cost of a cache hit), and that a cache miss has a miss penalty of 20 cycles (i.e., a cache access that results in a miss requires 2 cycles for lookup and a further 20 cycles to search memory, retrieve the desired cache line, and fill it into the cache). Moreover, based on the above, a TLB hit is 1 cycle, while a TLB miss is 1 cycle plus the time taken to find the page table entry. Suppose that both TLB and cache use LRU replacement. Suppose that you make a memory reference for virtual address 0x1000000000008A01

Suppose that the root of the page table is stored at physical frame number 0xA000000000, the second level of the page table required for the target virtual page is at physical frame 0xB000000000, the third level of the page table required for the target virtual page is at physical frame 0xC000000000, and the fourth level (the leaf) is at physical frame 0xD000000000. Finally, suppose that the physical frame pointed

to by the target virtual page number is 0xE000000000. List at least one cache set number that has data allocated within it.

2

Question 28

1 pts

TLB (6)

Now suppose that we have an architecture with the following parameters. Virtual addresses that are 64 bits, page sizes that are 4KB, physical addresses that are 64 bits, page tables that are four-level. Suppose that you have a TLB that is 128 entries 4-way set associative, and one level of cache that is 4MB, 4-way set associative, with 1KB lines. Suppose that the TLB can be accessed in 1 cycle (i.e., this is the cost of a TLB hit), the cache can be accessed in 2 cycles (i.e., this is the cost of a cache hit), and that a cache miss has a miss penalty of 20 cycles (i.e., a cache access that results in a miss requires 2 cycles for lookup and a further 20 cycles to search memory, retrieve the desired cache line, and fill it into the cache). Moreover, based on the above, a TLB hit is 1 cycle, while a TLB miss is 1 cycle plus the time taken to find the page table entry. Suppose that both TLB and cache use LRU replacement. Suppose that you make a memory reference for virtual address 0x1000000000008A01

For the set number selected in part 5, provide the tag of the cache line that is in the MRU position. Express the tag in hexadecimal notation, starting with "0x". Each hex character should be capitalized.

0xA000000000

Question 29

1 pts

TLB (7)

Now suppose that we have an architecture with the following parameters. Virtual

addresses that are 64 bits, page sizes that are 4KB, physical addresses that are 64 bits, page tables that are four-level. Suppose that you have a TLB that is 128 entries 4-way set associative, and one level of cache that is 4MB, 4-way set associative, with 1KB lines. Suppose that the TLB can be accessed in 1 cycle (i.e., this is the cost of a TLB hit), the cache can be accessed in 2 cycles (i.e., this is the cost of a cache hit), and that a cache miss has a miss penalty of 20 cycles (i.e., a cache access that results in a miss requires 2 cycles for lookup and a further 20 cycles to search memory, retrieve the desired cache line, and fill it into the cache). Moreover, based on the above, a TLB hit is 1 cycle, while a TLB miss is 1 cycle plus the time taken to find the page table entry. Suppose that you make a memory reference for virtual address 0x1000000000008A01

For the set number selected in part 6, provide the tag of the cache line that is in the LRU position. Express the tag in hexadecimal notation, starting with "0x". Each hex character should be capitalized.

0xA000000000

Question 30

1 pts

TLB (8)

Now suppose that we have an architecture with the following parameters. Virtual addresses that are 64 bits, page sizes that are 4KB, physical addresses that are 64 bits, page tables that are four-level. Suppose that you have a TLB that is 128 entries 4-way set associative, and one level of cache that is 4MB, 4-way set associative, with 1KB lines. Suppose that the TLB can be accessed in 1 cycle (i.e., this is the cost of a TLB hit), the cache can be accessed in 2 cycles (i.e., this is the cost of a cache hit), and that a cache miss has a miss penalty of 20 cycles (i.e., a cache access that results in a miss requires 2 cycles for lookup and a further 20 cycles to search memory, retrieve the desired cache line, and fill it into the cache). Moreover, based on the above, a TLB hit is 1 cycle, while a TLB miss is 1 cycle plus the time taken to find the page table entry. Suppose that you make a memory reference for virtual address 0x1000000000008A01

Suppose that after this memory reference, you make another one for virtual address 0x1000000000008A02. How many clock cycles will it take to service this, including both TLB management and cache management?

2

Question 31

1 pts

```
int main()
{
    if (fork() || fork())
        fork();
    printf("1 ");
    return 0;
}
```

What is the number of 1s printed by executing this program?

5

Question 32

1 pts

Select the true statements about pipe files vs regular files?

Answer 'Y' for True and 'N' for no. (case sensitive and without quotes!)

a. Pipe files cannot be seeked while regular files can. Y

b. Pipe files don't see benefits from caching techniques while regular files can.

N

c. Pipe files are in memory while regular files are in disks/persistent storage media.

Y

Question 33

1 pts

Consider the following program. List all possible outputs this program can make on a Unix system. Each output should be a string of capital letters separated by a , (no spaces).

```
// W(A) is equivalent to write(1, "A", sizeof "A")
#define W(x) write(1, #x, sizeof #x)
int main()
{
    W(A);
    int c = fork();
    W(B);
    if (c)
        wait(NULL);
    W(C);
}
```

Note here that W(A) is a macro which simply writes the character passed to W() to the output stream. For example, W(A) will output A, W(B) will output B.

ABCBC,ABBCC

Question 34

5 pts

```
Suppose there are two processes, A and B, which share semaphores X and Y. At the
 starting time, X = 0, Y = 1. A and B are entering the sections of code shown bel
ow:

Process A:                          Process B:
P(X);                               Operation 3;
P(Y);                               V(X);
Operation 1;                        Operation 4;
V(Y);                               P(Y);
Operation 2;                        Operation 5;
V(X);                               V(Y);
```

Which of the following are true?

Enter 'Y' for true and 'N' for False (case sensitive, no quotes!)

a. Operation 1 cannot start until operation 3 is complete. N

b. Operations 1 and 4 cannot execute simultaneously N

c. Operations 1 and 5 cannot execute simultaneously Y

d. Operations 2 and 5 cannot execute simultaneously N

e. The system may deadlock. N

Question 35

1 pts

Select the correct statement(s):

File descriptors are an abstraction -

- [ ] Built-in to modern disks
- [x] Provided by the standard C library
- [ ] Accessible only in the CPUs privileged mode
- [x] Provided by the OS
- [x] and are basically integers

Quiz saved at 12:27am Submit Quiz