Name: CS520

Final Exam

12 May 2017, 120 minutes, 26 questions, 100 points

> The exam is closed book and notes.
>
> Please keep all electronic devices turned off and out of reach.
>
> Note that a question may require *multiple* checked boxes for a correct answer. Checking *some* but not *all* of the required boxes will result in a *partial* answer worth only 2 of the 4 points. Checking any box that shouldn't be checked results in an *incorrect* answer, worth zero.

1. [0 pts]
   - ◯ Return my exam to my Kingsbury mailbox.
   - ◯ Hold my exam in your office. I will pick it up prior to September 15.
   - √ Shred my exam. I never want to see it again.

2. Consider this C declaration: [4 pts]

```
char *((*f)(float));
```

   This declares a:
   - ◯ function that takes a float, and returns a pointer to a pointer to a char.
   - √ pointer to a function that takes a float, and returns a pointer to a char.
   - ◯ pointer to a pointer to a function that takes a float, and returns a char.
   - ◯ none of the above.

3. Consider the following C function: [4 pts]

```
unsigned int f(void)
{
  int x = -1;
  return *(unsigned char *) &x;
}
```

   On a 2's complement machine with a byte-addressable memory, the function will:
   - ◯ return 0 if the machine is little-endian and 1 otherwise.
   - ◯ return 1 if the machine is little-endian and 0 otherwise.
   - ◯ always return 1.
   - √ always return 255.
   - ◯ none of the above.

4. Consider how -99 (base 10) would be represented in the memory of a Big Endian machine as a 16-bit 2's complement integer. The two bytes, shown left to right in increasing memory address order, would be: [4 pts]
   - √ `0xFF 0x9D`.
   - ◯ `0xFF 0x9C`.
   - ◯ `0x9C 0xFF`.
   - ◯ `0x80 0x63`.
   - ◯ none of the above.

5. Add together the following two 8-bit 2's complement integers (shown in hexadecimal): `0xFF` and `0xFF`. Which of the following are true statements about the result? [4 pts]

   √ The result (in hexadecimal) is `0xFE`.

   √ The result is negative.

   ◯ The result overflows.

   √ The result in decimal is -2.

   ◯ The result in decimal is -126.

6. The UTF-16 sequence (shown in hexadecimal) `0xD800 0xDC10` is represented in UTF-32 (in hexadecimal) as: [4 pts]

   ◯ two Unicode characters, `0x0000D800` and `0x0000DC10`.

   ◯ two Unicode characters, `0x00000000` and `0x00000010`.

   √ a single Unicode character, `0x00010010`.

   ◯ a single Unicode character, `0x00000010`.

   ◯ none of the above.

7. Encode the following Unicode character (shown in hex) as UTF-8: `0x10010`. The sequence of bytes (shown in hex) would be: [4 pts]

   ◯ `01 00 10`.

   ◯ `D0 90`.

   ◯ `E0 81 85`.

   √ `F0 90 80 90`.

   ◯ none of the above.

8. Add the following two IEEE single-precision floating-point values shown in hexadecimal: `0x40A00001` and `0xC0A00002`. The result in hexadecimal is: [4 pts]

   √ `0xB5000000`.

   ◯ `0xB5800000`.

   ◯ `0x80000000`.

   ◯ `0x00000000`.

   ◯ none of the above.

9. Interpret `0xABBADABA` (shown in hexadecimal) as an IEEE single-precision floating-point value. Which of the following statements about this value are true? [4 pts]

   √ It is negative.

   √ Its stored exponent is `0x57`.

   √ Its actual exponent is `-40` (in decimal).

   ◯ It is a denormalized value.

10. Consider the following vm520 program: [4 pts]

```
        jmp  bottom
        halt
        halt
        halt
    bottom:
        halt
```

What is the encoding of the `jmp` instruction? The `jmp` opcode is `0x14`, and it uses format 2.

- ◯ `0x14030000`.
- ◯ `0x00000314`.
- ◯ `0xFFFF3014`.
- √ `0x00003014`.
- ◯ none of the above.

11. Which of the following statements about linkers are true? [4 pts]

- ◯ They combine two intput assembly language files into a single output assembly language file.
- √ They combine two input object files into a single output object file.
- √ They match imported outsymbols in one input file with the exported insymbols in the other input file.
- √ If they find the same exported insymbol in both input files, then they report an error.

12. Which of the following statements about assemblers are true? [4 pts]

- ◯ They have two passes because the definition of a label may come before its use.
- √ They place in the insymbol section of the output object file all labels that are both defined and exported in the input file.
- √ They place in the outsymbol section of the object file all labels that are both imported and referenced in the input file but not defined.
- ◯ They use a symbol table to store the encoded instruction associated with each defined label.

13. In an implementation of an exception mechanism for C programs, the **cancelCatchException** function: [4 pts]

- ◯ pushs the current RBP, the saved RBP and the saved RIP onto the "snapshot" stack.
- ◯ puts the exception number into the RAX register.
- √ pops the "snapshot" stack.
- ◯ none of the above.

14. Which of the following statements about the Intel 64 are true? [4 pts]

- ◯ All parameters are passed on the stack.
- √ Stack frames are linked together by their saved RBP slots.
- √ Integer function return values are placed in RAX.
- √ The **call** instruction pushes the return address on top of the runtime stack.

15. Which of the following statements about a *conservative* garbage collector are true? [4 pts]

- ◯ It might incorrectly identify a block as garbage.
- √ It might incorrectly identify a block as not garbage.
- ◯ It might get confused if the object graph (how allocated blocks point to each other) contains a cycle.
- ◯ It will never combine adjacent garbage blocks.

16. Which of the following statements about using a POSIX condition variable are true? [4 pts]

- √ The caller of *pthread_cond_signal* must hold the lock of an associated mutex.
- √ The caller of *pthread_cond_wait* will execute again after being signaled and re-acquiring the lock of the associated mutex.
- ◯ If *pthread_cond_signal* is called when there are no threads waiting on the condition variable, then the signal will be applied to the next thread that calls *pthread_cond_wait*.
- ◯ Only one thread at a time is allowed to wait on a condition variable.

17. Which of the following statements about a POSIX mutex are true? [4 pts]

- ◯ Only one thread at a time can be waiting to lock a mutex.
- √ It is an error if a thread tries to unlock a mutex for which it does not own the lock.
- √ A critical section is protected by first locking a mutex, then executing the critical section, then unlocking the mutex.
- ◯ Two threads can both hold the lock for a mutex at the same time.

18. Concerning the concurrent hash table assignment, which of the following statements are true: [4 pts]

- ◯ We required each hash bucket to have its own mutex.
- √ Using multiple locks (mutexes) would allow multiple threads to safely update elements of different sections of a hash table at the same time.
- ◯ We needed to use a special hash function that was concurrency-aware.
- √ We only used mutexes, and did not use condition variables.

19. Which of the following statements about thread control blocks (TCBs) are true? [4 pts]

- √ The TCBs for all ready-to-run threads are linked together to form the ready list.
- √ A TCB is used to store the current state of a thread when the thread yields to another thread.
- ◯ The main thread does not have a TCB.
- √ The TCBs for all threads waiting to lock a mutex are linked together to form the wait queue for the mutex.

20. Which of the following statements about the Intel 64 CMPXCHG instruction are true? [4 pts]

- √ It is used with the LOCK prefix to ensure that its memory actions are atomic.
- ◯ It is a convenient way to swap the values of two memory locations in a single instruction.
- √ It is used when implementing mutexes in order to protect the contents of the struct containing the state of the mutex from concurrent updates by two threads.
- √ Its functionality can be utilized in C via the *atomic_flag_test_and_set* library function.

21. A memory cache with only one set is known as a: [4 pts]

- ◯ direct-mapped cache.
- ◯ set-associative cache.
- √ fully-associative cache.
- ◯ translation lookaside buffer.
- ◯ none of the above.

22. Consider how the following two C loops would be accessed by a memory system with 32 words and a direct-mapped cache with 16 sets and a block size of 2 words. [4 pts]

```
for (i = 0; i < 32; i++)
  a[i] = i;
sum = 0;
for (i = 0; i < 32; i++)
  sum += a[i];
```

Assume only the array is in the cache and the first word of the array is at address zero. If the cache is initially empty, how many cache hits would there be for writes and how many cache hits would there be for reads?

- √ 32 read hits and 16 write hits.
- ◯ 16 read hits and 32 write hits.
- ◯ 0 read hits and 0 write hits.
- ◯ 16 read hits and 16 write hits.
- ◯ none of the above.

23. Which of the following are examples of a program exhibiting spatial locality: [4 pts]

- √ iterating through all members of an array in order.
- ◯ repeatedly incrementing a loop counter variable inside a loop.
- √ a sequence of instructions being executed in order without any branch or call instructions.
- ◯ putting the function return address in the RAX register.

24. Which of the following statements about virtual memory are true? [4 pts]

- ◯ The translation lookaside buffer (TLB) contains an entry for every virtual page.
- √ The page table indicates which virtual pages are currently in physical memory.
- √ A page table entry can include memory-protection information such as indicating that a page is read-only.
- √ A read of a word in virtual memory can cause the virtual memory to perform a disk write.

25. Which of the following statements about the amodl/h (address modification) instruction for the IAS computer are true? [4 pts]

- ◯ It allowed the modification of the address at which the program would start executing.
- √ It used the contents of the accumulator register to modify the address field of another instruction.
- √ It compensated for the lack of an addressing mode to support using the contents of a register as a pointer.
- ◯ There is a similar instruction available on the Intel 64.

26. Which of the following statements about the Control Data Corporation (CDC) 6600 are true? [4 pts]

- ◯ It was not a stored-program computer.
- √ Its memory consisted of 60-bit words.
- ◯ It utilized the IEEE floating-point standard.
- ◯ It utilized the ASCII character encoding standard.