Exam Review

Tips: Review slide decks 22 – 38 along with assigned readings since 2/19. The final exam is not comprehensive, but answering certain questions may require mastery of prerequisite concepts from the first half of the course. The content in the slide decks and concepts from the assigned homework/laboratory assignments will have a somewhat higher priority (Not specific questions about the subject of the lab assignments). The exam is 153 points and could consist of a mixture of T/F, multiple-choice, fill-in-the-blank, short answer, essay, code-tracing, and code-writing exercises (all/some question types might be on the test, not all are shown in the review). An appendix of the registers, as shown in the textbook, will be provided. No digital or analog aids will be permitted (including calculators, problems should be friendly on the exam, not guaranteed for the review 😊). I highly recommend practicing code-tracing and code-writing exercises using pencil and paper only.

There is probably more to this review than we can cover in a class period.

1. T/F: Logic gates are typically implemented using diodes and transistors
2. T/F: The output of an OR gate is false if all its inputs are true.
3. T/F: A NOT gate inverts the input signal, giving an output of true if the input is false.
4. T/F: A process is an instance of a program in execution
5. T/F: A program's state consists entirely of its code and the data in memory.
6. T/F: The microarchitecture of a CPU, rather than its Instruction Set Architecture (ISA), defines the set of commands or instructions that the CPU can understand and execute.
7. T/F: A dedicated GPU (Graphics Processing Unit) is solely responsible for outputting images to a display and does not perform any general-purpose parallel processing tasks.
8. A fixed-point number with an 8-bit integer part and a 4-bit fractional part (using unsigned representation) has an implicit scaling factor of:

   a) 2^4

   b) 2^8

   c) 2^{-4}

   d) 2^{-8}

9. When designing an ALU, the subtraction of two numbers A and B is typically implemented by:

   a) Performing A + B directly.

b) Performing A + (one's complement of B).

c) Performing A + (two's complement of B).

d) Performing A - B using a dedicated subtractor circuit separate from the adder.

10. What is the main challenge when implementing integer division compared to integer multiplication in hardware?

a) Division requires significantly more memory.

b) Division algorithms are inherently iterative and can be more complex to control.

c) Division is a non-deterministic operation.

d) Division requires floating-point hardware.

11. Which of the following best describes a hardwired control unit?

a) It uses a microprogram stored in a control memory to generate control signals.

b) Its logic is implemented using fixed logic gates and flip-flops, making it faster but harder to modify.

c) It is more flexible and easier to design for complex instruction sets.

d) It is primarily used in Complex Instruction Set Computers (CISC).

12. Compared to a system bus, an expansion bus is typically designed to:

a) Provide faster communication between the CPU and main memory.

b) Connect slower peripheral devices to the system.

c) Carry addresses only, not data or control signals.

d) Be implemented using only serial connections.

13. The Von Neumann bottleneck refers to the limitation in system performance caused by:

a) The need for instructions and data to share a single bus for fetching and storing.

b) The speed difference between the CPU and the arithmetic logic unit.

c) The heat generated by high-speed processors.

d) The limited number of registers available in the CPU.

14. Which type of parallelism involves applying the same operation to multiple data elements simultaneously using specialized instructions and hardware (like vector processors or GPUs)?

a) Instruction-Level Parallelism

b) Task-Level Parallelism

c) Bit-Level Parallelism

d) Data-Level Parallelism (SIMD)

15. A quad word is __________ bits, but a nibble is ____________ bits.

16. _________ ___________ performs computation on the bits, but ___________ ___________ regulates updating of the memory elements

17. Concurrent flows are also known as ____________ _______________ (two words)

18. The circuitry within the CPU responsible for performing arithmetic and logical operations is known as the ____________

19. A memory's ____________ time is the time elapsed between the initiation of a memory operation and its completion.

20. In a synchronous bus, a shared ____________ signal is used to coordinate data transfers between devices.

21. A type of non-volatile memory that can be electrically erased and reprogrammed is known as ____________.

22. An unexpected event that occurs during program execution and disrupts the normal flow, often caused by errors like division by zero or invalid memory access, is called an ____________.

23. What are the major process states?

24. What is the address range of an architecture where n bits is reserved for addresses? On Linux systems, where does the code segment usually start?

25. What instruction is a program running in user mode not allowed to use?

26. What part of the Linux file system allows a user process to explore kernel data structures?

27. What decimal number does the bit pattern 0x2F410000 represent if it is a single-precision floating point number? Use the IEEE 754 single precision standard.

28. Write down the binary representation of the decimal number 4.75, assuming the IEEE 754 single precision format.

29. Suppose we are given the task of generating code to multiply the integer variable `x` by various different constant factors `K`. To be efficient, we want to only use the operations `+`, `-`, and `<<`. For the following values of `K`, write C expressions to perform the multiplication using at most three operations per expression.

    a.) K = 9 b.) K = -15 c.) K = 72 d.) K = 80

30. What is a system?
31. What are the three elements specific by an Instruction Set Architecture?
32. What is the difference between SRAM and DRAM?

33. For a function with prototype:\

```
long transform(long p, long q, long r);
```

GCC generates the following assembly code:

```
transform:
  addq %rdx, %rsi
  subq %rsi, %rdi
  movq %rdi, %rax
  movq %rsi, %rcx
  sarq $63, %rcx
  xorq %rcx, %rax
  ret
```

Parameters p, q, and r are passed in registers %rdi, %rsi, and %rdx respectively. The code stores the return value in %rax.

Write C code for transform that will have an effect equivalent to the assembly code shown.

34. For a function with prototype:

```
long cumulative_calc(long limit);
```

and a helper function:

```
long square_and_add_5(long val);
```

Write code that calculates a sum by iterating from 1 up to a limit, adding twice the current number if it's even, or adding twice the result of squaring the number and adding 5 if it's odd.

35. Use a jump table to return a category code based on whether an input character is a digit, uppercase letter, lowercase letter, or other.
For a function with prototype:

```
long check_char_category(char input_char);
```

This function should return a category code based on the input character:

0: If the character is a digit ('0'-'9')
1: If the character is an uppercase letter ('A'-'Z')
2: If the character is a lowercase letter ('a'-'z')
Any other character: Return -1

Assume the input character is in the lower 8 bits of %rdi. The function will use a jump table based on a derived index.

36. A jle .Ltarget instruction follows a cmpq %rbx, %rax. Which specific flag or combination of flags in the %rflags register does jle inspect to determine whether to jump?