EE380 Fall 2025 Final Exam

Your name is:

Your UK (ABC123 format) account name is:

The two questions you skipped are:

There are 22 questions; you are to answer any 20 questions. You will lose 5 points for each of the questions that you did not answer correctly, except the two questions that you said to skip (above). For each multiple choice question, mark only the ☐ in front of the single best answer; questions for which you are to mark all that apply state that at the start of the question. The other questions should get **short** answers; excessively long answers may be considered incorrect. The test is to be taken without using electronic devices, closed book, closed notes, and closed minds (read neither tests nor minds of others ;-)).

You can detach and keep the reference sheet at the end of this exam!

1. For this question, *mark all that apply*. Which of the following are true statements about your Verilog projects in this course?

☐ In Your Average Problem, your **`testbench`** was supposed to check that your module produced correct answers by comparing results to those of an oracle module

☐ In the Multicycle Team Project, **`mainmem()`** was a module that could take multiple clock cycles to complete a read request

☐ In the Single-Cycle Team Project, **`mainmem()`** was a module that could take multiple clock cycles to complete a read request

☐ In the Pipelined Team Project, the pipeline had six stages

☐ In the Pipelined Team Project, **`halt <= 1;`** was not set until the offending instruction reached the last stage of the pipeline

2. *For this question, mark all answers that apply.* Which of the following Verilog expressions results in a value of 1 (decimal)?

☐ **`4'bzx01`**

☐ **`{2{1'b1}}-2`**

☐ **`a[1]`** *where* **`reg [7:0] a = 2;`**

☐ **`(a == 1'bx)`** where **`reg a = 1'bx;`**

☐ **`((a<<1)=={a[6:0],1'b0})`** where **`reg [7:0] a;`**

3. *For this question, mark all answers that apply.* Which of the following statements about Verilog are true?

☐ It is OK to have multiple **`always`** blocks in a module

☐ In general, things named with **`$`** are not Synthesizable

☐ **`reg [31:0] a;`** means exactly the same thing as **`reg a[31:0];`**

☐ Given **`wire a, b;`**, the statement **`assign {a,b}=3;`** is perfectly acceptable

☐ For wires **`a`**, **`b`**, and **`c`**, the Verilog code **`xor d(a, b, c);`** means exactly the same thing as **`assign a=b^c;`**

4. The company that just hired you has a dispenser for free, cold, cans of everybody's favorite caffeinated beverage. Unfortunately, since you're the most junior employee, in addition to your regular workload, it is your job to keep the dispenser stocked. Being a clever computer engineer, you decide to put a little sensor in the dispenser to detect when there is only one can left in the dispenser and pop-up a notification on your workstation computer. Which of the following I/O mechanisms do you think would be most appropriate for your workstation computer to use for the sensor: DMA, interrupts, or polling? Explain your choice.

5. Given the following MIPS code, in which "..." refers to a sequence of instructions that do not alter the value of **`$t0`**, would predicting the **`beq`** is always not-taken be better or worse than predicting always taken? Why?

```
    li   $t0,1000
a:  beq  $t0,$zero,b
    ...
    addi $t0,$t0,-1
    j    a
b:  ...
```

6. For this question, *mark all that apply*. Which of the following statements about IEEE 754 single-precision (32-bit) floating-point arithmetic is/are correct?

☐ 0 is not representable as a normalized **`float`** value

☐ Both 10 and 1/10 are precisely representable as floating-point numbers

☐ If all values are normal and within range, **`(a+b)`** is always equal to **`(b+a)`**

☐ If all values are normal and within range, **`(a+(b+c))`** is always equal to **`((a+b)+c)`**

☐ If all values are normal and within range, **`(a*(b*c))`** is always equal to **`((a*b)*c)`**

7. For this question, *mark all that apply*. Which of the following are true statements about parallel computers?

☐ With the same number of PEs, SIMD hardware is generally simpler than MIMD hardware

☐ Programs for both SIMD computers and GPUs make use of the concept of disabling some PEs for some operations

☐ In most computer networks, latency between nodes is equal to 1/bandwidth (e.g., 1ns for 1Gb/s Ethernet transmission)

☐ In a large parallel computer, the key to both high reliability and high availability is having redundant or spare parts for the system

☐ Although pipelined parallel processing is used in most computers, computer programmers typically don't need to know about it; the compiler and hardware automatically make use of it

8. For this question, *mark all that apply*. Which of the following things is generally *not* stored in a **`stack frame`** even within a recursive function?

☐ The return address

☐ Local (**`auto`**) variables

☐ Global (**`static`**) variables

☐ Arguments passed to the current function

☐ The heap for **`new`** or **`malloc`** memory allocation

9. Consider executing the following code sequence on the **pipelined** MIPS implementation given at the end of this test **without value forwarding**. Show any true dependences and reorder these instructions so that the same values are computed, but pipelined execution can be expected to take fewer clock cycles. You can show the order using the letters A, B, C, D, and E. (You don't need the fastest reordering, just one that is faster than the order given below.)

```
A: andi $t1,$t0,601
B: addu $t3,$t1,$t2
C: ori  $t4,$t2,42
D: sw   $t5,0($t0)
E: xor  $t0,$t0,$t6
```

10. For this question, *mark all that apply*. Consider executing the MIPS instruction **`xor $t0,$t1,$t2`** and then *any one of the following single MIPS instructions*. Assume that the hardware being used looks like the **pipelined** MIPS implementation given at the end of this test. Mark each that would execute faster using **value forwarding** than without value forwarding.

☐ **`sw $t0,4($t3)`**

☐ **`lw $t4,4($t1)`**

☐ **`ori $t2,$t1,380`**

☐ **`add $t3,$t0,$t3`**

☐ **`bne $t0,$0,place`**

11. For this question, *mark all that apply*. Which of the following statements about the memory hierarchy is true?

☐ A processor writing to a clean cache line makes it dirty

☐ A shorter line size improves handling of temporal locality

☐ L1 cache usually is accessed using logical memory addresses

☐ For the same total cache capacity, a larger set size will usually increase hit ratio

☐ It is possible to suffer a TLB miss for a reference to a datum that is already in cache

Questions 12, 13, and 14 refer to diagrams on the last page of this exam. You may separate that sheet from your exam to make it easier to reference, and do not need to submit it.

12. For this question, *mark all that apply*. Consider the single-cycle and pipelined MIPS implementation diagrams at the back of this exam. Which of the following control signals **must be 0** when executing an `beq` instruction?

☐ RegDst
☐ ALUSrc
☐ RegWrite
☐ MemWrite
☐ MemtoReg

13. For this question, *mark all that apply*. Consider the single-cycle and pipelined MIPS implementation diagrams at the back of this exam. Which of the following control signals **must be 0** for an instruction to behave as a side-effect-free null operation (NOP, pipeline bubble)?

☐ Branch
☐ ALUSrc
☐ RegWrite
☐ MemWrite
☐ MemtoReg

14. *For this question, mark all answers that apply.* At the back of this exam is a diagram showing the internal structure of Intel's Nehalem processor. According to that diagram, which of the following five techniques are used in this design?

☐ There are three levels of cache
☐ The L1 instruction cache is direct mapped
☐ Out-of-order instruction execution with register renaming
☐ Superscalar execution of floating point addition and multiplication
☐ The L1 data TLB can hold entries for more 4K-byte pages than the L1 instruction TLB

15. *For this question, mark all answers that apply.* Which of the following correctly describes the operation of a BHB (Branch History Buffer)?

☐ Conditional jump instructions can have BHB entries

☐ It is accessed at the same time an instruction is fetched, using the same address

☐ It stores the complete history of taken/not-taken decisions for each branch instruction executed

☐ It remembers the target address computed previously so that it doesn't need to add the branch offset to the PC

☐ For previously-executed branch instructions, it provides the prediction of whether that branch will be taken or not taken

16. *For this question, mark all answers that apply.* Which of the following mechanisms used in computer architecture implement some form of speculation?

☐ Value forwarding

☐ Booth's algorithm

☐ Branch prediction

☐ Speculative addition

☐ Carry lookahead addition

17. For this question, *mark all that apply*. Which of the following statements about processor implementations is generally true?

☐ The value on a bus can be read and latched into at most one register at a time

☐ Programmable logic is usually bigger and slower than optimized random logic implementing the same function

☐ The best speedup through pipelining generally comes when every pipe stage has about the same circuit delay

☐ A pipeline bubble means one or more stages of the pipeline contain instructions that produce no observable effect

☐ In general, tri-state driver outputs are enabled at the start of a clock cycle and registers latch new values at the end of the clock cycle

18. You have written a program that runs for a total of **10 seconds** on Intel's fastest processor. You need it to run faster, so you decide to rewrite it to run inside the computer's GPU. The catch is that the GPU cannot directly access the disk drive, so it still needs the Intel processor to spend **2 seconds** to copy the data from disk into main memory (which both the processor and GPU can access). By Amdahl's law, assuming the rest of the program can be perfectly parallelized and arbitrarily many GPU processing elements are available, **what is the maximum speedup one might obtain**?

19. Consider executing the following C program on a system where an `int` can hold a 32-bit 2's complement integer value and a `float` is a 32-bit IEEE 754 floating-point value. What would you expect the following code to do and why? (Hint: `1000000000` is a little less than 2^{30}).

```
main()
{
  float f; int i0;
  for (f=0; f<1000000000; ++f) { ++i; }
  if (f==i) printf("Yes!\n"); else printf("Nope.\n");
}
```

20. A particular microcontroller has an single (L1) data cache that takes 2 clock cycles to access and a main memory that takes 100 cycles to access. Assume that the hit ratio is 99%. Which **one** of the following is closest to the average access time in clock cycles?

☐ 2

☐ 3

☐ 99

☐ 100

☐ 200

21. For this question, *mark all that apply*. Which of the following statements about compilers are true?

☐ Lexical analysis for most computer languages can be done using a simple state machine (i.e., a DFA) without a stack

☐ Although most computer languages are specified by a context free (type 2) grammar, identifiers are really context sensitive

☐ Code generation actions in a simple compiler might be little more than `printf()` calls embedded within the parser

☐ To make `a+b*c` behave as if it were written as `a+(b*c)`, the pattern matching `+` expressions should be called from within the pattern matching `*` expressions

☐ Backpatching and multi-pass resolution are techniques associated with how assemblers can process forward references

22. For this question, *mark all that apply*. Page table mechanisms can be used to implement which of the following?

☐ Protection of the OS memory and memory of other processes from access by an unpriviledged running process

☐ Sharing of DLLs (Dynamic Link Libraries) by multiple processes, even ones running different programs

☐ Virtual memory much larger than the physical main memory, using a disk drive to hold the memory image

☐ DSM (Distributed Shared Memory) in which pages can be shared across multiple computers linked by some network

☐ Mapped file I/O, in which the contents of a file can be read and written without using read or write OS calls

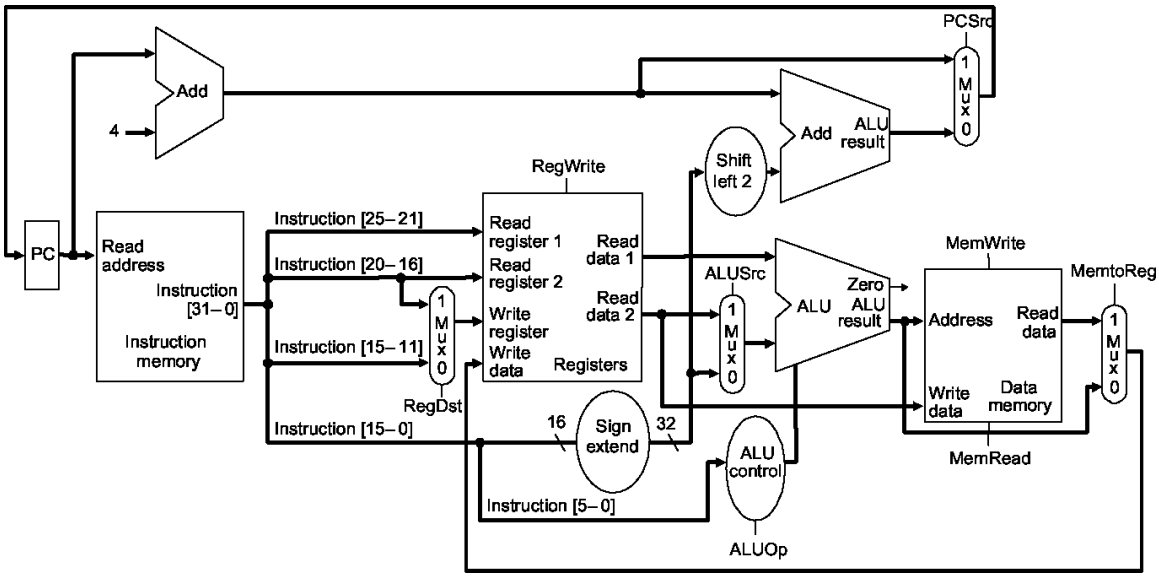
This page is intentionally blank!

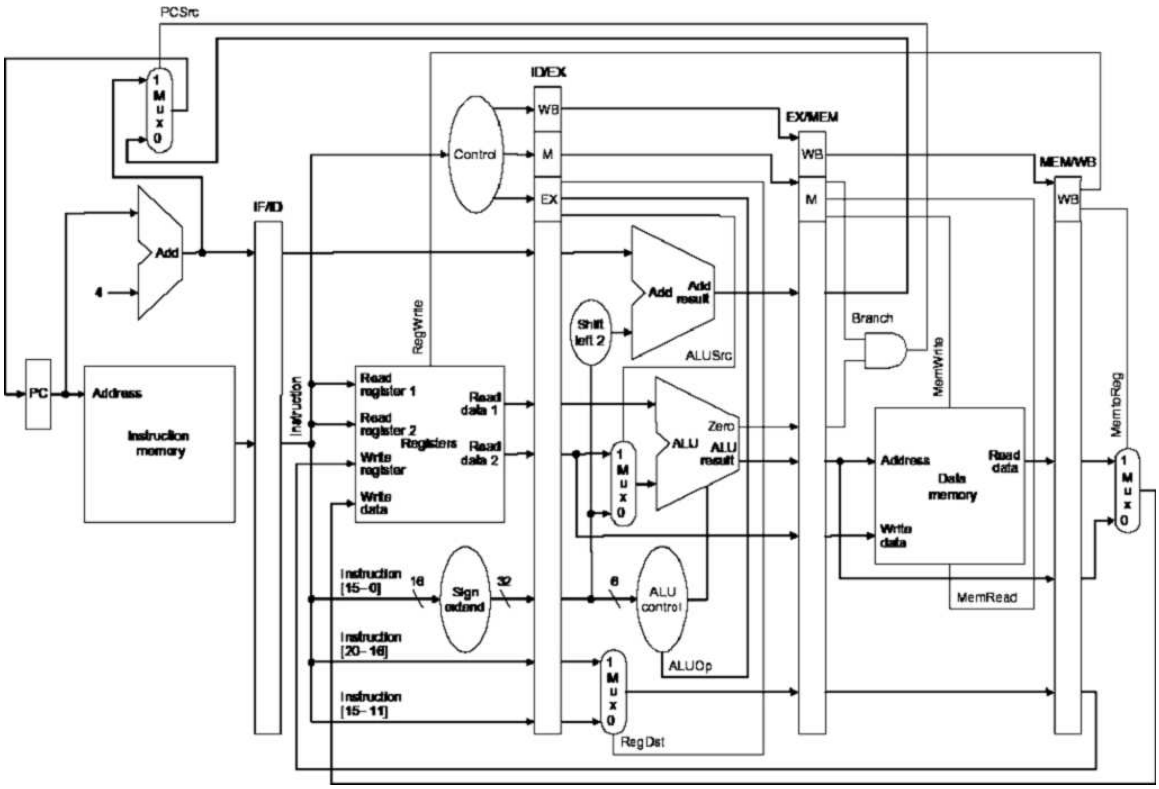
Once you have torn page 11/12 off your exam, this should be back page of the exam as you turn it in. You can keep or discard page 11/12.

Reference Information

You may separate this sheet from your exam and do not need to submit it.

The following two figures show single-cycle and pipelined versions of the same basic MIPS subset implementation, which are not exactly the same ones we used in class.





The following diagram shows the internal structure of an Intel Nehalem processor.