NOTE: It is my policy to give a failing grade in the course to any student who either gives or receives aid on any exam or quiz.

INSTRUCTIONS: For multiple choice questions, circle the letter of the *one best choice* for each question. There is no penalty for guessing. Answer short answer questions in the spaces provided. For other questions, follow the directions in the question. All questions count equally unless otherwise indicated.

1. Are you allowed to use a calculator or any other electronic device for this exam?
   A. No

2. Is your cell phone turned off?
   A. Yes

3. List the two elements of the single-cycle MIPS datapath (ignoring data memory) that are connected to the system clock, **and** name the two stages of the pipelined datapath that these two elements correspond to.

4. What does `2'b0` represent in Verilog?
   A. A value that is 2 big 2 be true.
   B. A branch target address
   C. A 2×0 multiplexer
   D. A 0×2 multiplexer
   E. Two binary zeros

5. What are the three uses of the immediate field of an I-format instruction? Give the instruction name of one example of each.

   A. ____

   B. ____

   C. ____

6. What is the difference between *addi* and *andi*?
   A. Nothing except that the first one adds two values and the second one does a bitwise AND.
   B. The first one adds and the second one subtracts.
   C. The first one is an R-format instruction and the second one is an I-format instruction.
   D. The first one does sign extension of the immediate operand and the second one does zero extension.
   E. The first one does sign extension and the second one does not extend the immediate operand at all.

7. Write the *complete* Verilog expression for computing the *effective address* of a *load word* instruction. (Only part of the expression is on the Green Card.)

8. Write the *complete* Verilog expression for computing the *target address* of a *branch equal* instruction.

9. Is `0xCAFEBABE` a valid MIPS *word address*?
   A. No, because the rightmost two bits are not zero.
   B. No, because it is not a 32-bit value.
   C. No, because it is too big.
   D. No, because it is not a number.
   E. Yes

10. What does a *branch equal* instruction do?
   A. It is used to call a subroutine.
   B. It is used to compute a new value for register 0.
   C. It changes the PC if two registers contain the same value.
   D. It causes the PC and a register to have equal values.
   E. It is used to increment an index register.

11. What does `R[rt]` mean?
   A. The contents of the register specified by bits 20:16 of an instruction.
   B. The number of the register specified by bits 20:16 of an instruction.
   C. The instruction located in bits 20:16 of the PC.
   D. The PC of the immediate operand.
   E. The immediate operand, shifted left five bit positions.

12. When is a *load upper immediate* instruction used?
   A. When the left side of a register holds the value to be put in the immediate field of a J-format instruction.
   B. When a value that cannot be represented in 16 bits needs to be loaded into a register. (The next instruction would be *ori* to put the other half of the value into the register.)
   C. When the immediate operand is negative.
   D. When there are not enough bits in an R-format instruction.
   E. When the *slti* instruction is broken.

13. Which of the following sets of instructions do *not* write anything to the register file?
   A. *addi, andi, add*
   B. *slti, lw, sll*
   C. *beq, sw, j*
   D. *sub, ori, or*
   E. *slt, jal, nor*

14. Which of the following describes the internal structure of a 2×5 multiplexer?
   A. Five 2×1 multiplexers with the same *select* input connected to them all.
   B. Two 5×2 multiplexers, each with a different select input.
   C. Two 1×5 multiplexers, with OR gates connecting the outputs together.
   D. Three decoders and one full adder, with the same 4-bit function code going to each one.
   E. Two tri-state buffers and five AND gates.

15. How many wires (inputs plus outputs) connect to a 2×32 multiplexer?
   A. 34
   B. 35
   C. 64
   D. 97
   E. 108

16. What are the two possible sources of information that might be written to the register file when an instruction completes?
   A. The PC and the ALU
   B. The instruction memory and the data memory
   C. The instruction and the PC
   D. The ALU and the data memory
   E. The adder and the subtracter.

17. When is the *rt* field used to select the register to be used for writing to the register file?
    A. When it is a *jump* instruction
    B. When it is an immediate instruction
    C. When it is an R-format instruction
    D. When it is a floating-point instruction
    E. When the register file is full

18. Where is the *op code decoder* and what are its inputs and its outputs?

19. What value does the ALU calculate for *beq* instructions?
    A. The number of lines of code to be executed
    B. The address of the next instruction
    C. The address of the previous instruction
    D. The difference between the two registers being compared
    E. The number of parameters to pass to the subroutine

20. What determines the maximum clock speed that can be used in the single cycle processor design?
    A. The number of registers in the register file
    B. The contents of the PC
    C. The contents of the PC+4
    D. The number of propagation delays needed to fetch and execute the slowest instruction.
    E. The number of gates needed to fetch and execute the fastest instruction.

21. Is the *control unit* used in the single cycle design combinational logic, sequential logic, or a combination of both?
    A. Combinational only
    B. Sequential only
    C. Some of both
    D. All of the above
    E. None of the above

22. Circle the letters of **all** the following instructions that cause the *MemRead* control signal to be true:
    A. *lw*
    B. *sw*
    C. *beq*
    D. *addi*
    E. *j*

23. Circle the letters of **all** the following instructions that will fail if the *RegWrite* control signal gets stuck at one.
    A. *addi*
    B. *lw*
    C. *sw*
    D. *beq*
    E. *add*

24. How long would it take to execute two million instructions on a processor with a 2GHz clock, and an average CPI of 2? (Show work for possible partial credit.)

25. Processor *A* takes ten seconds to execute the same program that processor B executes in eleven seconds. Write two sentences that express the relative performances, one as a percentage and one as a ratio. The first sentence will say, “Processor [A|B] is [X] percent [faster|slower] than processor [B|A],” and the second one will say, “Processor [A|B] is [X] times [faster|slower] than processor [B|A].”

A. ____

B. ____

26. A single-cycle CPU design with a 300MHz clock is converted to a perfectly balanced pipeline design with five stages. What will be the new clock speed?
A. 200 MHz
B. 300 MHz
C. 500 MHz
D. 700 MHz
E. 1.5 GHz

27. What is the *latency* of a pipelined processor?
A. How long it takes to fetch an instruction from memory
B. How long it takes to read from the disk
C. The reaction time of the slowest stage
D. The time it takes an instruction to pass through all the stages of the pipeline
E. The rate at which instructions are retired.

28. What is the major disadvantage of a deep pipeline?
A. The clock cannot keep up with the memory
B. The memory cannot keep up with the clock
C. The latency is too wide, so the cache depth gets compromised
D. Branch instructions disrupt the flow too much
E. It makes the computer harder to sell.

29. Give the full names of the pipeline stages used in the textbook, and state briefly what happens in each stage.

A. IF: ____

B. ID: ____

C. EX: ____

D. MEM: ____

E. WB: ____

30. List the *names* (only the names) of all the pipeline registers used in the book.

31. During which pipeline stage is the register file read from?
A. IF
B. ID
C. EX
D. MEM
E. WB

32. During which pipeline stage is the register file written to?
A. IF
B. ID
C. EX
D. MEM
E. WB

33. During which pipeline stage is the ALU used?
   A. IF
   B. ID
   C. EX
   D. MEM
   E. WB

34. Which of the pipeline registers hold information for controlling the WB stage? List as many of the names from your answer to Question 30 as needed:

35. What is a *data hazard?*
   A. When an instruction stores data into memory before the cache is ready for it.
   B. When an instruction reads data from memory, but the data is not in the cache.
   C. When an instruction in the ID stage needs a register value that is in one of the later pipeline stages.
   D. When an instruction in the WB stage needs a register value that is in one of the earlier pipeline stages.
   E. When a *jump* instruction needs an address that is more than 26 bits wide.

36. What is the term for the mechanism that reduces or eliminates data hazards?
   A. Branch prediction
   B. Accelerated load
   C. Delayed store
   D. Grocery store
   E. Register forwarding

37. What is a *bubble*?
   A. A speedup mechanism that floats instructions through the pipeline faster
   B. A speedup instruction that adds extra stages to the pipeline
   C. A wasted set of pipeline cycles due to a hazard that could not be eliminated
   D. When two instructions are in the same stage of the pipeline at the same time
   E. When a particular instruction does nothing during a pipeline stage, such as an *add* instruction in the MEM stage.

38. A computer has 2 GB of main memory and 1 MB of cache; what proportion of main memory can be in cache at a time?
   A. 10%
   B. 20%
   C. 5%
   D. $2^{20} \div (2 \times 2^{30})$
   E. $(2 \times 2^{30}) \div 2^{20}$

39. What is *temporal locality*, and why is it important for cache performance?

40. How long does it take a 7,200 RPM disk to make one revolution?
   A. 7200 Hz
   B. 7200 milliseconds
   C. 7200 ÷ 60 = 120 seconds
   D. 1 ÷ 120 = 0.00833 seconds
   E. 320 KB/sec

41. What is *bandwidth*, and what is its unit of measure?
    A. The speed of a clock, measured in Hz
    B. The speed of a clock, measured in seconds
    C. The capacity of a disk, measured in gigabytes
    D. The number of wires in a bus, measured in bits
    E. The rate at which information moves, measured in bits per second

42. Is a cache memory implemented in hardware, software, or a combination of both?
    A. It’s all hardware
    B. It’s all software
    C. It’s a combination of both
    D. It depends on the operating system
    E. It depends on the bandwidth

43. How many main memory *blocks* are there if there are 32 words per *cache line* and 2^{30} words of cache memory? (Show work for possible partial credit.)

44. How do you calculate the average memory access time of a computer that has a cache? Give the formula and explain the terms in it.

MIPS Reference Data

CORE INSTRUCTION SET

NAME, MNEMONIC		FOR-MAT	OPERATION (in Verilog)	OPCODE / FUNCT (Hex)
Add	add	R	R[rd] = R[rs] + R[rt]	0 / 20$_{hex}$
Add Immediate	addi	I	R[rt] = R[rs] + SignExtImm	8$_{hex}$
Add Imm. Unsigned	addiu	I	R[rt] = R[rs] + SignExtImm	9$_{hex}$
Add Unsigned	addu	R	R[rd] = R[rs] + R[rt]	0 / 21$_{hex}$
And	and	R	R[rd] = R[rs] & R[rt]	0 / 24$_{hex}$
And Immediate	andi	I	R[rt] = R[rs] & ZeroExtImm	c$_{hex}$

Branch On Equal	beq	I	if(R[rs]==R[rt])
			PC=PC+4+BranchAddr
				4$_{hex}$

Branch On Not Equal	bne	I	if(R[rs]!=R[rt])
			PC=PC+4+BranchAdd
				5$_{hex}$

Jump	j	J	PC=JumpAddr	2$_{hex}$
Jump And Link	jal	J	R[31]=PC+8;PC=JumpAddr	3$_{hex}$
Jump Register	jr	R	PC=R[rs]	0 / 08$_{hex}$

Load Byte Unsigned	lbu	I	R[rt]={24'b0,M[R[rs]
			+SignExtImm](7:0)}
				24$_{hex}$

Load Halfword Unsigned	lhu	I	R[rt]={16'b0,M[R[rs]
			+SignExtImm](15:0)}
				25$_{hex}$

Load Linked	ll	I	R[rt] = M[R[rs]+SignExtImm]	30$_{hex}$
Load Upper Imm.	lui	I	R[rt] = {imm, 16'b0}	f$_{hex}$
Load Word	lw	I	R[rt] = M[R[rs]+SignExtImm]	23$_{hex}$
Nor	nor	R	R[rd] = ~ (R[rs] | R[rt])	0 / 27$_{hex}$
Or	or	R	R[rd] = R[rs] | R[rt]	0 / 25$_{hex}$
Or Immediate	ori	I	R[rt] = R[rs] | ZeroExtImm	d$_{hex}$
Set Less Than	slt	R	R[rd] = (R[rs] < R[rt]) ? 1 : 0	0 / 2a$_{hex}$
Set Less Than Imm.	slti	I	R[rt] = (R[rs] < SignExtImm)? 1 : 0	a$_{hex}$
Set Less Than Imm. Unsigned	sltiu	I	R[rt] = (R[rs] < SignExtImm) ? 1 : 0	b$_{hex}$
Set Less Than Unsig.	sltu	R	R[rd] = (R[rs] < R[rt]) ? 1 : 0	0 / 2b$_{hex}$
Shift Left Logical	sll	R	R[rd] = R[rt] << shamt	0 / 00$_{hex}$
Shift Right Logical	srl	R	R[rd] = R[rt] >> shamt	0 / 02$_{hex}$
Store Byte	sb	I	M[R[rs]+SignExtImm](7:0) = R[rt](7:0)	28$_{hex}$

Store Conditional	sc	I	M[R[rs]+SignExtImm] = R[rt];
			R[rt] = (*atomic*) ? 1 : 0
				38$_{hex}$

Store Halfword	sh	I	M[R[rs]+SignExtImm](15:0) = R[rt](15:0)	29$_{hex}$
Store Word	sw	I	M[R[rs]+SignExtImm] = R[rt]	2b$_{hex}$
Subtract	sub	R	R[rd] = R[rs] - R[rt]	0 / 22$_{hex}$
Subtract Unsigned	subu	R	R[rd] = R[rs] - R[rt]	0 / 23

BASIC INSTRUCTION FORMATS

R	opcode	rs	rt	rd	shamt	funct
	31 26	25 21	20 16	15 11	10 6	5 0
I	opcode	rs	rt	immediate		
	31 26	25 21	20 16	15 0		
J	opcode	address				
	31 26	25 0				

SIZE PREFIXES (10^x for Disk, Communication; 2^x for Memory)

SIZE	PRE-FIX	SIZE	PRE-FIX	SIZE	PRE-FIX	SIZE	PRE-FIX
$10^3, 2^{10}$	Kilo-	$10^{15}, 2^{50}$	Peta-	10^{-3}	milli-	10^{-15}	femto-
$10^6, 2^{20}$	Mega-	$10^{18}, 2^{60}$	Exa-	10^{-6}	micro-	10^{-18}	atto-
$10^9, 2^{30}$	Giga-	$10^{21}, 2^{70}$	Zetta-	10^{-9}	nano-	10^{-21}	zepto-
$10^{12}, 2^{40}$	Tera-	$10^{24}, 2^{80}$	Yotta-	10^{-12}	pico-	10^{-24}	yocto-

The symbol for each prefix is just its first letter, except μ is used for micro.