Short Answer:

(1) (2) As minimum feature sized decrease, what happens to wire resistance?

(2) (4) If you are mainly worried about program size, what type of instruction set would you use? What if performance was your primary concern?

(3) (2) Is MOS technology current-based or charge-based?

(4) (2) Do benchmarks need to change over time?

(5) (2) Pipelining increases instruction ______________ but also increases instruction ______________. (Fill in the blanks.)

(6) (4) What is a Structural Hazard, and how do I deal with it?

(7) (4) Why have designers been reducing "Vdd"?

(8) (2) Are wire delays or transistors more likely to be the most significant limit on clock frequency in the future?

(9) (3) What is the "threshold voltage"?

(10) (3) How does reducing the minimum feature size affect power density?

(11) (3) What is Amdahl’s law (in words)?

(12) (4) What is the difference between static and dynamic leakage?

(13) (4) What is a benchmark program? What is the **perfect** benchmark?

(14) (6) Register allocation is a very effective and useful compiler optimization. However, it cannot be used in certain situations for correctness reasons - explain why.

(15) (8) What are the three types of data hazards? Briefly outline how to deal with one of them.

(16) (8) What is a control hazard? Briefly explain how to deal with it.

(17) (8) In an MOS device, there is a gate, drain, and source. Briefly explain how this device works.

(18) (4) There is a "wall" between the internal processing elements on a chip and the outside world. Explain why that is.

(19) (3) Why do most current chips feature multiple cores on the same die?

(20) (6) Why is branch prediction important? What performance enhancing techniques have made it so? List 3 examples of existing Static Branch Prediction strategies in order of (average) increasing effectiveness.

(21) (10) Suppose floating point instructions are responsible for 55% of the execution time of a particular benchmark, and the floating point square root (FPSQ) by itself is responsible for 23% of the execution time. You have a choice between speeding up the FPSQ instruction by a factor of 9, or making all floating point instructions run faster by a factor of 1.65. Which should you do? (Show how you set up the equations, although you do not need to actually solve them.)

(22) (8) In class, we talked about the cycle by cycle steps that occur on different interrupts. For example, here is what happens if there is an illegal operand interrupt generated by instruction i+1:

	1	2	3	4	5	6	7	8	9
i	IF	ID	EX	MEM	WB				
i+1		IF	ID	EX	MEM	WB	<- Interrupt detected		
i+2			IF	ID	EX	MEM	WB	<- Instruction Squashed	
i+3				IF	ID	EX	MEM	WB	<- Trap Handler fetched
i+4					IF	ID	EX	MEM	WB

Fill out the following table if instruction i+1 experiences a fault in the EX stage (arithmetic exception, for example):

	1	2	3	4	5	6	7	8	9	10
i	IF	ID	EX	MEM	WB					
i+1		IF	ID	EX	MEM	WB				
i+2			IF	ID	EX	MEM	WB			
i+3				IF	ID	EX	MEM	WB		
i+4					IF	ID	EX	MEM	WB	
i+5						IF	ID	EX	MEM	WB

What happens in this case?

	1	2	3	4	5	6	7	8	9	10
i	IF	ID	EX	MEM	WB	<- Data write causes Page Fault				
i+1		IF	ID	EX	MEM	WB	<- Illegal Opcode			
i+2			IF	ID	EX	MEM	WB			
i+3				IF	ID	EX	MEM	WB		
i+4					IF	ID	EX	MEM	WB	
i+5						IF	ID	EX	MEM	WB