Very short answer questions. "True" and "False" are considered very short answers.

(2) Clock rates have grown by a factor of 1000 while power consumed has only grown by a factor of 30. How was this accomplished?

(1) The use of large, multilevel caches can substantially reduce the memory bandwidth demands of a processor.

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

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

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

(1) It is not necessary to simulate very many instructions in order to get an accurate performance measure of the memory heirarchy.

(1) A program's locality behavior often varies over the run of an entire program.

(1) The instruction set architecture has little impact on the ability to pipeline a processor.

(1) Using a write-through cache simplifies the implementation of a cache coherence protocol.

(1) Does peak performance track observed performance?

(1) Which is more effective, dynamic or static branch prediction?

(1) Predicting the direction of a branch is not enough. What else is necessary?

(1) Do benchmarks remain valid indefinitely?

(2) What is Amdahl's law (in words)?

(1) Write down the 3-term CPU performance equation developed in class.

(1) The instruction set architecture has little impact on the implementability of a virtual machine monitor.

(1) Write down the average memory access time equation.

(2) Why are there multiple dies per silicon wafer? Why not just fabricate one huge die per wafer?

(2) What is the relationship between speculation and power consumption?

(2) List 2 techniques that are useful for reducing the Miss Rate.

(2) List 2 techniques that are useful for reducing the Hit Time.

(2) List 2 techniques that are useful for reducing the Miss Penalty.

(3) What is the goal of the memory heirarchy? What two principles make it work?

(1) What type of parallelism do SIMD machines exploit?

(1) What type of parallelism do MIMD machines exploit?

(2) What are the four classifications of cache misses?

(1) Which type of cache miss can be changed by altering the mapping scheme?

(1) Which type of cache miss can be reduced by using longer lines?

(1) Which type of cache miss can be increased by using longer lines?

(1) What hardware structure is necessary in order to make a snooping protocol work?

(2) What is the "threshold voltage"?

(1) What pipeline hazard can be avoided by "throwing money at the problem"?

(1) What pipeline hazard can be avoided using a technique known as value prediction?

(2) Why do CMOS processors avoid accessing items that are located off-chip?

(2) How many entries are there in a (5,3) Gshare branch predictor? How many bits?

(2) What are the two instructions used in RISC machines to support atomicity?

(2) What is the main difference between a commodity cluster and a custom cluster?

(2) Relaxing the requirement that Writes complete before Reads yields a model known as:

Short Answers:

(3) Why is it difficult to come up with good benchmarks for parallel processors?

(3 pts) There is a problem with bus-based coherence protocols. What is the problem, and what is currently done to solve that problem? Briefly describe how the solution works, don't just give the name.

(3) What is a control hazard? Briefly explain how modern processors deal with it.

(3) What are the 4 types of data hazards? Which one is not really a hazard? Why not?

(4) What are the two biggest challenges in parallel processing? In other words, what two things are keeping parallel processors from being the dominant architecture?

(4) What is the primary difference between Scoreboarding and Tomasulo's algorithm? What hardware feature makes Tomasulo's work?

(6) Understanding the hardware can influence how you write programs. Give at least 2 examples of how you might write software differently for a heavily pipelined machine verses a non-pipelined one.

(4) What is the definition of a basic block? Why is there a desire to create larger ones?

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

(4) Supporting precise interrupts in machines that allow out of order completion is a challenge. Briefly explain why, and give two different techniques that can be used to provide precise interrupts.

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

(5) Compare and contrast Superscalar and VLIW. Describe each, and list the advantages and disadvantages of each approach.

(5) Describe the difference between shared memory and message passing machines. Include the impact on design, cost, and programming model.

(3) In an MOS device, there is a gate, drain, and source. Briefly explain how this device works. Pictures are welcome.

(3) Assuming a 16-bit address and a 256-byte Direct Mapped cache with a linesize=8, show how an address is partitioned/interpreted by the cache.

(3) Assuming a 16-bit address and a 160-byte 5-way SA cache with a linesize=4, show how an address is partitioned/interpreted by the cache.

(3) Given a 1 Megabyte physical memory, a 24 bit Virtual address, and a page size of 2K bytes, write down the number of entries in the Page Table, and the width of each entry.

(4) Given a 1 Megabyte physical memory, a 32 bit Virtual address, and a page size of 2K bytes, write down the number of entries in the Page Table, and the width of each entry. Is there a problem with this configuration? If so, how can you fix it?

(4) There are at least two types of control flow changes that standard dynamic branch predictors have trouble with. Explain what the two are and why they are a problem, and give the technique used to successfully deal with one of them.

(4) The memory system presents challenges to ILP designers as well. What is it about the memory system that makes it hard for compilers to optimize code, and also for execution units to achieve maximal performance? (This is not a question about technology, it's a higher-level question).

(10) You are given the following code sequence:

```
ADDF    F3,F4,F5
SUBF    F3,F1,F5
MULTF   F5,F6,F7
DIVF    F5,F5,F7
```

Assume the instructions are MIPS-like (i.e. ADDF F3,F4,F5 means F3=F4+F5), and that there are 8 logical and 16 physical registers. On the left below is the register mapping upon entering the code sequence. Your job is to fill in the mappings after the execution of the DIVF instruction, including what is on the free list. (Assume that during the execution of this code, no registers are released - in other words, the free list will be shorter at the end than at the beginning.)

BEFORE	
Logical	Physical
0	1
1	3
2	5
3	7
4	9
5	11
6	13
7	15

AFTER	
Logical	Physical
0	
1	
2	
3	
4	
5	
6	
7	

Free Pool: 0,2,4,6,8,10,12,14

Free Pool:

Now, rewrite the code sequence below using the actual physical register names instead of the logical ones.

```
ADDF   P__,P__,P__
SUBF   P__,P__,P__
MULTF  P__,P__,P__
DIVF   P__,P__,P__
```

(6) 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	
10										
i	IF	ID	RR	EX	MEM	WB				
i+1		IF	ID	RR	EX	MEM	WB	<- Interrupt detected		
i+2			IF	ID	RR	EX	MEM	WB	<- Instruction Squashed	
i+3				IF	ID	RR	EX	MEM	WB	<- Trap Handler fetched
i+4					IF	ID	RR	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	11
i	IF	ID	RR	EX	MEM	WB					
i+1		IF	ID	RR	EX	MEM	WB				
i+2			IF	ID	RR	EX	MEM	WB			
i+3				IF	ID	RR	EX	MEM	WB		
i+4					IF	ID	RR	EX	MEM	WB	
i+5						IF	ID	RR	EX	MEM	WB

What happens in this case?

	1	2	3	4	5	6	7	8	9	10	11
i	IF	ID	RR	EX	MEM	WB	<- Data write causes Page Fault				
i+1		IF	ID	RR	EX	MEM	WB	<- Illegal Opcode			
i+2			IF	ID	RR	EX	MEM	WB			
i+3				IF	ID	RR	EX	MEM	WB		
i+4					IF	ID	RR	EX	MEM	WB	
i+5						IF	ID	RR	EX	MEM	WB

(2) What is the maximum number of exceptions that could happen at one time in the above machine?

Good questions for future use: 22. (4) Assume a relatively large fully associative write-back cache that contains no valid data. Given the following sequence of 5 memory operations (the address of the operation is in the square brackets):

WriteMem[100]
ReadMem[100]
WriteMem[100]
WriteMem[200]
WriteMem[100]

What are the number of hits and misses when using write allocate versus no-write allocate? 24. (4) Assume that L2 has a block size four times that of L1. Show how a miss for an address that causes a replacement in L1 and L2 can lead to a violation of the inclusion property.