CS 312	**Computer Organization & Architecture**	Test 1
Name:	Fall 2002	Max Pts: 39

Important: This is an open book test. You can use any books, notes, or paper, but not exchange anything with other students. You are not allowed to use any electronic/communication devices. *Do not log into the computer during the test.* Any calculations and rough work can be done on the back side of the test pages. You will lose five points for not writing your name.

1. [5 pt] Explain the difference between *instruction set* and *instruction set architecture*.

2. [5 pt] What is a *die*? What is *yield* in chip manufacturing process? If I double the area of a die, does that guarantee the yield to be doubled as well? Explain your answer.

3. [9 pt] We wish to compare the performance of two different machines: m_1 and m_2. The following measurements are made:

Program	Time on m_1	Time of m_2
1	4 sec	5 sec
2	2 sec	4 sec

(a) Which machine is faster for each program and by how much?

(b) We made some more measurements using program 1. m_1 executed 32 million instructions while m_2 executed 81 million instructions for this program. What can you conclude about MIPS rating for each machine for this program.

(c) If the clock rate of m_1 is 1.6 GHz and the clock rate of m_2 is 500 MHz, find the CPI for program 1 for both machines.

4. [10 pt] Suppose we enhance a machine to make all integer instructions run 1.56 times faster, and all floating point instructions run 1.71 times faster. If the execution time of some benchmark before speedup is 10 seconds, what will the speedup be if floating point instructions constitute 0.18 of all instructions, with the rest of the instructions being integer instructions.

5. [10 pt] Consider the following code:

```
main:    add   $t0, $zero, 5
         add   $t1, $zero, $t0
         add   $t2, $zero, 1
         add   $t3, $zero, $t0

loop:    sub   $t1, $t1, $t2
         beq   $t1, $zero, finish
         add   $t3, $t3, $t1
         j     loop

finish:
```

What does this code do? Assume that `add` and `sub` require 1 instruction cycle, and `j` and `bne` require 2 instruction cycles. How many seconds will it take to run this code on a 20 MHz machine?