COMP180: Computer Organization
Spring 2001, Lecture Section 1, Dit-Yan Yeung
Quiz #3
9 April 2001

Name: _____   Student ID: _____   Lab: _____

Answer all questions in the space provided. Time allowed: 20 minutes

1. For each of the following statements, write down 'true' if it is true and 'false' otherwise.

    (a) Two's complement is not used in the IEEE 754 floating-point representation for
        representing signed integers.   Ans: _____

    (b) The equation "dividend = quotient × divisor + remainder" holds for both signed
        and unsigned integers.   Ans: _____

    (c) In a floating-point representation, overflow/underflow occurs if the significand
        field is not large enough to hold the value.   Ans: _____

    (d) Immediate addressing is an addressing mode for loading a constant value from
        memory as an operand.   Ans: _____

    (e) The following x86 assembly code iterates 10 times before exiting from the loop.
        ```
        Again:  mov     cx, 10
                loop    Again
        ```
                                                                                    Ans: _____

    (f) Real mode addressing is available on both 16-bit and 32-bit x86 architectures.   Ans: _____

    (g) "`mov ax, 80000`" is a valid x86 assembly instruction.   Ans: _____

    (h) "`mov al, bx`" is a valid x86 assembly instruction.   Ans: _____

    (i) "`mov cx, bx`" is a valid x86 assembly instruction.   Ans: _____

    (j) "`mov 18, ax`" is a valid x86 assembly instruction.   Ans: _____

2. Using the integer multiplication algorithm discussed in class, show all the steps involved when two 2-bit binary numbers 11 (multiplicand) and 10 (multiplier) are multiplied to obtain the product. You can show the steps by completing the table below.

Iteration	Multiplicand	Product	Remark

3. Consider the following x86 assembly code, where A, B, C, D and E refer to the memory locations of five variables in the corresponding high-level language assignment statement:

```
        mov     eax, B
        cmp     eax, C
        mov     bl, 0
        jle     N1
        mov     bl, 1
N1:     mov     eax, D
        cmp     eax, E
        mov     bh, 0
        jne     N2
        mov     bh, 1
N2:     or      bl, bh
        mov     A, bl
```

Give a C/C++ assignment statement that corresponds to the assembly code. The statement evaluates an expression (on the right-hand side) and assigns its value to variable A.

4. Give four different methods to clear register ax (i.e., set ax to 0). Each method should use only one x86 assembly language instruction.

(a) _____

(b) _____

(c) _____

(d) _____