Midterm Examination

Solution and Marking Scheme

1. d (4 points)

2. a (4 points)

3. e (4 points)

4. (14 points total)

   a) For a given program, let N denote the total number of instructions, N_A the number of class A instructions, N_B the number of class B instructions, and N_C the number of class C instructions.

   CPU time = CPU clock cycles / clock rate

   CPU time on M1 using C1 = $(4\ N_A + 6\ N_B + 8\ N_C)$ / 400 msec
   = (4 x 0.3 + 6 x 0.5 + 8 x 0.2) N / 400 msec
   = 5.8 N / 400 msec (1.5 points)

   CPU time on M2 using C1 = (2 x 0.3 + 4 x 0.5 + 3 x 0.2) N / 200 msec
   = 3.2 N / 200 msec (1.5 points)

   Thus, using C1, M1 is 6.4 / 5.8 or 32 / 29 times as fast as M2. (1.5 points)

   b) CPU time on M1 using C2 = (4 x 0.3 + 6 x 0.2 + 8 x 0.5) N / 400 msec
   = 6.4 N / 400 msec (1.5 points)

   CPU time on M2 using C2 = (2 x 0.3 + 4 x 0.2 + 3 x 0.5) N / 200 msec
   = 2.9 N / 200 msec (1.5 points)

   Thus, using C2, M2 is 6.4 / 5.8 or 32 / 29 times as fast as M1. (1.5 points)

   c) CPU time on M1 using C3 = (4 x 0.5 + 6 x 0.3 + 8 x 0.2) N / 400 msec
   = 5.4 N / 400 msec (1.5 points)

   C3 is the best choice since the CPU time for the program on M1 using C3 is the shortest. (1 point)

   d) CPU time on M2 using C3 = (2 x 0.5 + 4 x 0.3 + 3 x 0.2) N / 200 msec
   = 2.8 N / 200 msec (1.5 points)

   C3 is the best choice since the CPU time for the program on M2 using C3 is the shortest. (1 point)

5. (10 points total)

Let N denote the total number of instructions in the program.

CPU time = CPU clock cycles / clock rate

CPU time on M1 = N / 400 msec (4 points)

CPU time on M2 = (5 x 0.2 + 4 x 0.1 + 4 x 0.5 + 3 x 0.15 + 3 x 0.05) N / 200 msec
= N / 50 msec (4 points)

Thus, M1 is 400 / 50 or 8 times as fast as M2. (2 points)

6. (10 points total)

This pseudoinstruction

```
ble  $s0, $s1, L
```

can be implemented as

```
slt  $t0, $s1, $s0
beq  $t0, $zero, L
```

7. (15 points total)

```
      add  $t1, $zero, $zero    # $t1 = i = 0
      add  $t2, $t0, $zero      # $t2 = address of a[0]
Loop: slti $t3, $t1, 10         # $t3 = 1 if i < 10
      beq  $t3, $zero, Exit     # Exit loop if i >= 10
      addi $t4, $t1, 1          # $t4 = i + 1
      sw   $t4, 0($t2)          # a[i] = i + 1
      addi $t1, $t1, 1          # $t1 = i + 1
      addi $t2, $t2, 4          # $t2 = $t2 + 4
      j    Loop
Exit:
```

8. (15 points total)

a)

Inputs			Outputs	
a	b	CarryIn	CarryOut	Sum
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

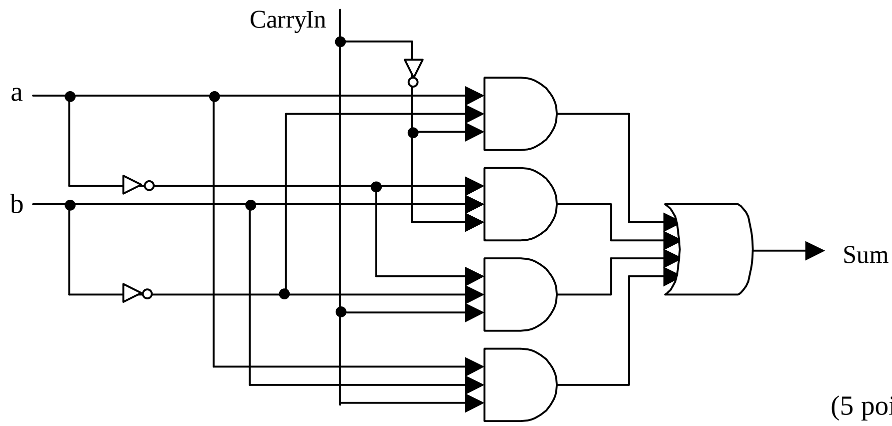
(5 points)

b) Sum = $(a \cdot \underline{b} \cdot \underline{\text{CarryIn}}) + (\underline{a} \cdot b \cdot \underline{\text{CarryIn}}) + (\underline{a} \cdot \underline{b} \cdot \text{CarryIn}) + (a \cdot b \cdot \text{CarryIn})$

(5 points)

[underline denotes NOT]

c)



(5 points)

9. (12 points total)

a) Bit 31 : 0 (1 point)
Bits 30-23 : 01111110 (2 points)
Bits 22-0 : 11111111111111111111111 (2 points)

b) x = -24 (3 points)

c) With the biased notation, floating-point numbers can be compared in magnitude (e.g, for sorting) simply by performing integer comparison without having to compute the actual values of the floating-point numbers. (4 points)

10. (12 points total)

a) $2^{31} + 2^{27} + 2^{26} + 2^{25} + 2^{24} + 2^{23} + 2^{22} + 2^{21} + 2^{19} + 2^{18} + 2^{17} + 2^{16} + 2^{15} + 2^{14}$ (4 points)

b) $-2^{31} + 2^{27} + 2^{26} + 2^{25} + 2^{24} + 2^{23} + 2^{22} + 2^{21} + 2^{19} + 2^{18} + 2^{17} + 2^{16} + 2^{15} + 2^{14}$ (4 points)

c) Sign bit = 1
Exponent field = $00011111_{two} = 31_{ten}$
Significand field = 11011111100000000000000

Decimal value = $-(1 + 2^{-1} + 2^{-2} + 2^{-4} + 2^{-5} + 2^{-6} + 2^{-7} + 2^{-8} + 2^{-9}) \times 2^{31-127}$ (4 points)