98/100

CSCI-21 Midterm #1 Name Ismael Cortez

Please use PENCIL and erase mistakes. Closed books and notes, except for the 'cheat sheet' I posted on my Web site. Write your name on it (and NOTHING else), and turn in the cheat sheet with your exam. No electronic devices of any kind. For arithmetic answers, ***show your work*** and please put a box around your answer so I can tell it from your scratch work. I will not give full credit without showing work. Turn your cell phone OFF and put it away where you can't easily get to it. If your phone makes any noise whatever, I will take 5 points off your exam score. 100 total points.

14

1. What are the two primary pieces of information that the *base* of a number system gives you? Illustrate both parts of your answer using base 12. (4 points)

1. It tells you about the set of digits and its cardinality.
For base 12: $S = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B\}$

4

$|S| = 12$

2. It tells you the place values for any number expressed in that base, which are always integer powers of '6'.
For base 12: 4A5 has place values 12^2 12^1 12^0

2. Convert the following numbers from the indicated bases to decimal. (2 points each)

a. 254_7

dec = $(2 \times 7^2) + (5 \times 7^1) + (4 \times 7^0)$
= $(2 \times 49) + (35) + 4$ = $98 + 35 + 4$

137

1
98
35
4
137

b. $1A2_{12}$

dec = $(1 \times 12^2) + (10 \times 12^1) + (2 \times 12^0)$
= 144 + 120 + 2

266

144
120
2
266

10

c. 251_6

dec = $(2 \times 6^2) + (5 \times 6^1) + (1 \times 6^0)$
= 72 + 30 + 1

103

72
30
1
103

d. 1234_4

dec = $(1 \times 4^2) + (2 \times 4^1) + (3 \times 4^0)$
= 16 + 8 + 3

27

1
16
8
3
27

e. $A8_{15}$

dec = $(10 \times 15^1) + (8 \times 15^0)$ =
150 + 8 = 158

158

8

3. Using the repeated division method, convert the following decimal numbers to unsigned 8-bit binary. (2 points each)

a. 146

146	0
73	1
36	0
18	0
9	1
4	0
2	0
1	1
0	

1001 0010

4

b. 69

69	1
34	0
17	1
8	0
4	0
2	0
1	1
0	

0100 0101

zero on front

4. Using the double-and-add method, convert the following 8-bit unsigned binary numbers to decimal. (2 points each)

a. 01011011

1, 2, 5, 11, 22, 45, 91

4

b. 11011010

1, 3, 6, 13, 27, 54, 109, 218

12

5. Using the place and digit values, convert the following 8-bit unsigned binary numbers to decimal. (2 points each)

a. 11110110

128	64	32	16	8	4	2	1
1	1	1	1	0	1	1	0

128 + 64 + 32 + 16 + 6 = 128 + 118

125
118
246

246 ✓

4

b. 00111010

128	64	32	16	8	4	2	1
0	0	1	1	1	0	1	0

32 + 16 + 10 = 58 ✓

6. Convert the following 32-bit binary numbers to hexadecimal. (2 points each)

a. 10010101|1100|0110|0000|1111|0111|0010
9 5 C 6 0 F 7 2

0x95C60F72 ✓

4

b. 11100001|1010|1011|0011|0100|1101|1000
E 1 A B 3 4 D 8

0xE1AB34D8 ✓

7. Convert the following 32-bit binary numbers to octal. (2 points each) why the ( )?

a. 10|010|101|110|001|100|000|111|101|110|010
2 2 5 6 1 4 0 7 5 6 2

(22561407562)$_8$ ✓

4

b. 11|100|001|101|010|110|011|010|011|011|000
3 4 1 5 2 6 3 2 3 3 0

(34152632330)$_8$ ✓

12

8. Convert the following hexadecimal numbers to binary. Leave a gap between nibbles. (2 points each)

a. 0xD21A7FE8

1101 0010 0001 1010 0111 1111 1110 1000

4

b. 0xB03C6459

1011 0000 0011 1100 0110 0100 0101 1001

9. Add the following unsigned binary numbers. For each, tell me if overflow occurs. (2 points each)

a.
```
  1 1 1 1 1 1
  0 1 0 1 1 1 1 0
 +0 0 1 1 1 0 1 1
-----------------
 1 0 0 1 1 0 0 1
```

1001 1001

No overflow

64 + 16 + 14 = 94
32 + 16 + 11 = 59
128 + 16 + 9 = 153

4

b.
```
    1 1 1 1 1
  1 1 1 0 1 1 0 1
 +0 0 1 1 1 1 0 0
-----------------
1|0 0 1 0 1 0 0 1
```

Because a 1 carried out at the MSB, there is overflow.

0010 1001

just "yes" worked too?

10. Subtract (by borrowing as needed) the following unsigned binary numbers. You may NOT convert using one's or two's complement. (2 points each)

0001 1101

a.
```
    0 1 10 1
  0 0 1  0  0 1 1 1
 -0 0 0  0  1 0 1 0
-------------------
  0 0 0  1  1 1 0 1
```

32 + 7 = 39
10 - 10
29

16 + 13 = 29

4

b.
```
      10 10
    0  1  0  16
  0 1 1 1  1  1  0 1
 -0 0 1 0  1  1  1 1
---------------------
  0 1 0 0  1  1 1 0
```

64 + 32 + 16 + 13 = 112 + 13

32 + 15 = 47

64 + 14 = 78

```
  1 1 1
 +125
  - 47
  78
```

0100 1110

14

11. Subtract the following 8-bit binary numbers by converting using two's complement and adding. For each, tell me if overflow occurs1(3 points each)

a. 0 1 1 1 0 0 1 0
-0 0 1 0 1 1 1 1

1 1 1 1
1 1 0 1 0 0 0 1
+ 0 1 1 1 0 0 1 0
1 0 1 0 0 0 0 1 1

64 + 32 + 16 + 2 = 114
32 + 15 = 47
64 + 3 = 67

0 1 0 0 _ 0 0 1 1

NO overflow.

6

b. 0 1 0 0 1 1 1 0
-0 1 1 1 1 0 1 1

1 0 0 0 0 1 0 1
+ 0 1 0 0 1 1 1 0
1 1 0 1 0 0 1 1

64 + 14 = 78
64 + 32 + 16 + 11 = 112 + 11 = 123

-128 + 64 + 16 + 3
-128 + 83

1101 0011

NO overflow

12. Convert the following 8-bit 2's complement binary numbers to decimal. (2 points each)

a. 10000000

-128

b. 11111111

-1

-128 + 64 + 32 + 16 + 15 = -128 + 112 + 15
-128 + 127

8

c. 10101011

-128 + 32 + 11 = -128 + 43

-85

d. 00101111

32 + 15 = 47

47

14

13. Sign extend the following hexadecimal numbers to 32 bits (write them in hexadecimal) (1 point each):

a. 0x6F 8 bits

0x0000006F

2

b. 0x8C 8 bits

0xFFFFFF8C

14. Zero extend the following hexadecimal numbers to 32 bits (write them in hexadecimal) (1 point each):

a. 0xF3 8 bits

0x000000F3

2

b. 0x2B 8 bits

0x0000002B

15. Convert the following decimal values to the indicated bases (2 points each):

a. 73 = 243 $_5$ = 67 $_{11}$ = 12A $_{13}$ ~~133~~

73 | 3
14 | 4
2 | 2

73 | 7
6 | 6

73 | A
9 | 2
1 | 1

4, 3, 2, 1

49 + 14 + 10 = 73

10

b. 56 = 62 $_9$ = 2002 $_3$ = 44 $_{13}$

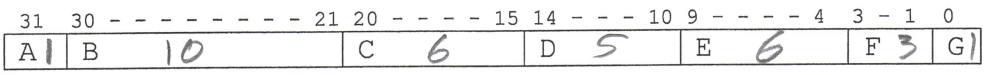
56 | 2
6 | 6

56 | 2
18 | 0
6 | 0
2 | 2

56 | 4
4 | 4

16. Assume that a 32-bit word held in $t0 has the following fields (9 points):

9



So, field A is one bit wide in bit 31 (the high-order bit), and field E is 6 bits wide in bits 9 – 4, etc. Each of the following can be done in two or three instructions.

a. Write MIPS instructions to convert field B into an unsigned 32-bit value in register $t0. (3 points)

```
sll  $t0, $t0, 1.
srl  $t0, $t0, 22.
```

3

b. Write MIPS instructions to test the bit in field G and branch to LABEL if it is 1. (3 points)

```
andi  $t1, $t0, 1.
bne   $t1, $zero, LABEL
```

3

c. Write MIPS instructions to convert field C into a two's-complement signed value in register $t0. Assume field C is signed two's-complement already. (3 points)

```
sll  $t0, $t0, 11.
sra  $t0, $t0 26
```

3

0x69 = 0110 1001

17. Write a complete MIPS program to put the following bit pattern into register `$t0`: `0x69696969`. Start by putting a 1 in the low order bit of register `$t0`. Using only bitwise logical operations on that 1 bit, and on bits derived from that 1 bit, build the value for register `$t1`. (Fewer than 15 instructions) (15 points)

```
.text
.globl main

main:
    ori  $t0, $zero, 1      # $t0 = 0001
    sll  $t1, $t0, 1        # $t1 = 0010
    or   $t2, $t0, $t1      # $t2 = 0011
    sll  $t2, $t2, 1        # $t2 = 0110
    sll  $t1, $t0, 3        # $t1 = 1000
    or   $t1, $t0, $t1      # $t1 = 1001
    sll  $t2, $t2, 4        # $t2 = 0110 0000
    or   $t1, $t1, $t2      # $t1 = 0110 1001
    sll  $t2, $t1, 8        # $t2 = 0x6900
    or   $t1, $t1, $t2      # $t1 = 0x6969
    sll  $t2, $t1 16        # $t2 = 0x69690000
    or   $t1, $t1, $t2      # $t1 = 0x69696969
```

15

Syscall w/ code 10 ?