

Practice midterm solutions

19 March 2014

You have up to 1 hour, 45 minutes. You may use a calculator, but no text book or notes.

1. For each statement below, fill in the blank with the *best* term from the following list. Some terms might be used more than once; some might not be used at all.

• algorithm • ASCII • binary • bit • Boolean • byte • hexadecimal • pixel
• pseudo-code • two's complement • Unicode

- (a) hexadecimal is the name of a numbering system in which each digit corresponds to exactly four bits.
- (b) pseudo-code is a notation for specifying algorithms. It is more natural for humans than a programming language.
- (c) A(n) byte is exactly 8 bits.
- (d) ASCII is an encoding of characters used in American English using 7 bits per character.

2. Write down the decimal (base 10) equivalents for the following 6-bit signed (two's complement) binary numbers. (That means the answers might be negative!)

The values of the columns in 6-bit two's complement are -32, 16, 8, 4, 2, 1.

$$1\ 1\ 0\ 0\ 1\ 0 = \underline{-32+16+2 = -14} \quad 1\ 1\ 1\ 1\ 0\ 1 = \underline{-32+16+8+4+1 = -3}$$

$$1\ 1\ 0\ 1\ 1\ 0 = \underline{-32+16+4+2 = -10} \quad 0\ 1\ 0\ 0\ 0\ 1 = \underline{16+1 = +17}$$

$$0\ 0\ 0\ 1\ 0\ 1 = \underline{4+1 = +5} \quad 1\ 1\ 1\ 1\ 1\ 1 = \underline{-32+16+8+4+2+1 = -1}$$

3. Add the following pairs of 5-bit signed (two's complement) binary numbers. Your answers must be in binary, but you should check your work by converting to decimal. Remember, values can be negative!

The values of the columns are -16, 8, 4, 2, 1. Remember the number circle, 4+12 is -16 because there's no way to represent +16 in 5-bit two's complement, and instead we *overflow* from positive numbers into the negative ones. A similar effect happens with -10+-15, only this time there's an extra carry bit that gets thrown away because it doesn't fit into the 5-bit answer.

$$\begin{array}{r}
 1\ 1 \\
 0\ 0\ 1\ 0\ 0 = +4 \\
 +\ 0\ 1\ 1\ 0\ 0 = +12 \\
 \hline
 1\ 0\ 0\ 0\ 0 = -16
 \end{array}
 \qquad
 \begin{array}{r}
 1 \\
 1\ 0\ 1\ 0\ 1 = -11 \\
 +\ 0\ 0\ 1\ 0\ 0 = +4 \\
 \hline
 1\ 1\ 0\ 0\ 1 = -7
 \end{array}
 \qquad
 \begin{array}{r}
 1 \\
 1\ 0\ 1\ 1\ 0 = -10 \\
 +\ 1\ 0\ 0\ 0\ 1 = -15 \\
 \hline
 0\ 0\ 1\ 1\ 1 = +7
 \end{array}$$

4. Suppose we want to design encodings just for the five letters A, H, M, N, and T.
- How many bits would we need to represent each letter in a **fixed-width** encoding? 3, because $2^3 = 8$ is enough letters.
 - Using the fixed-width encoding in the previous question, how many bits would we need to represent the nine-letter word MANHATTAN? $27 = 9 \text{ letters} \times 3 \text{ bits per letter}$
 - Draw a tree to represent a **variable-width** encoding of these five letters. Use your tree to encode the word MANHATTAN. How many bits did you need? How many bits did you *save*, compared to the fixed-width encoding?

There are many different trees with 5 leaves. Here are two:

you want to place the more frequent letters on the shorter paths.

MANHATTAN

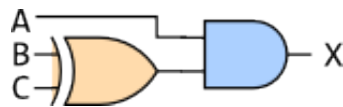
Letter	Frequency	Ordered by freq: ANTMH
M	1	Tree 1: MANHATTAN bits: $4+1+2+4+1+3+3+1+2 = 21 \text{ bits}$ (6 shorter)
A	3	
N	2	
H	1	
T	2	

Tree 2: MANHATTAN
bits: $3+2+2+3+2+2+2+2 = 20 \text{ bits}$
(7 shorter)

5. Create a truth table to show the value of $X' + (X \cdot Y)$ for all possible inputs of X and Y.

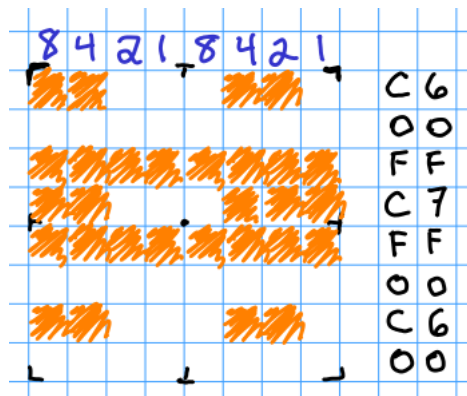
X	Y	X'	X·Y	X' + X·Y
0	0	1	0	1
0	1	1	0	1
1	0	0	0	0
1	1	0	1	1

6. Which Boolean expression is equivalent to the following circuit diagram?



- (a) $A \oplus (B \cdot C)$
- (b) $A \cdot (B \oplus C)$ **(This one)**
- (c) $A \cdot (B + C)$
- (d) $(A \cdot B) + C$
- (e) $A + (B \oplus C)$

7. Decode the following hexadecimal notation into an 8×8 icon, using 1 bit per pixel.



8. Convert the following binary number into hexadecimal and octal.

1 0 1 0 1 1 1 0 0 1 1 0 0 1 0 0

For hexadecimal, group it into fours starting from the right:

(1 0 1 0) (1 1 1 0) (0 1 1 0) (0 1 0 0)

8 4 2 1 8 4 2 1 8 4 2 1 8 4 2 1

Answer: A E 6 4

For octal, group it into threes starting from the right:

(1) (0 1 0) (1 1 1) (0 0 1) (1 0 0) (1 0 0)

1 4 2 1 4 2 1 4 2 1 4 2 1 4 2 1

Answer: 1 2 7 1 4 4

9. What is the output of the following algorithm? Remember to indicate clearly what is *output* and what is scratch work.

1. Set N to 0
2. Set K to 1
3. If $K > 4$ then output N and stop.
4. Set N to $N + K$
5. Set K to $K + 1$
6. Go back to step 3.

N: 0 1 3 6 10

K: 1 2 3 4 5

Output:

10