Practice Midterm Solutions

24 October 2012

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 • bit • Boolean • byte • hexadecimal • pixel • pseudo-code • Python • unicode

   (a) hexadecimal is base 16 notation, which uses digits 0–9 and letters A–F.

   (b) unicode is an encoding of characters used in all the world's languages.

   (c) pseudo-code is a way of describing a computational technique to other humans using carefully structured English.

   (d) Python is a high-level programming language used to instruct computers.

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.

   `0 0 1 1 1 0 =` **8+4+2 = +14** `0 0 1 0 1 0 =` **8+2 = 10**

   `1 1 1 1 0 0 =` **-32+16+8+4 = -4** `1 0 1 0 0 0 =` **-32+8 = -24**

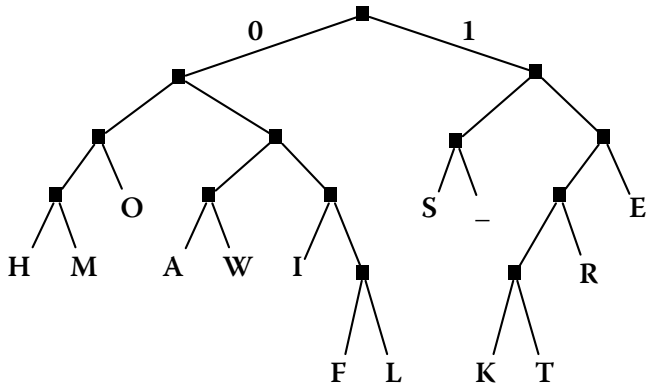
   `0 1 1 0 1 1 =` **16+8+2+1 = +27** `1 1 0 1 0 0 =` **-32+16+4 = -12**

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

```
1 1 1                 1 1 1 1                   1 1
  0 0 1 0 0 = +4        1 1 1 1 1 =  -1         0 1 1 0 0 = +12
+ 1 1 1 0 0 = -4      + 1 0 1 1 0 = -10       + 0 1 1 0 0 = +12
-------------         -------------           -------------
  0 0 0 0 0 =  0        1 0 1 0 1 = -11         1 1 0 0 0 =  -8
```

   Remember the number circle, 12+12 is -8 because there's no way to represent 24 in 5-bit two's complement, and instead we *overflow* from positive numbers into the negative ones.

4. Below is a tree representing a variable-width encoding.



(a) Use the tree to encode the following message in binary:

```
   T      R      E      E      _      F      L      O      W

1 1 0 0 1,1 1 0 1,1 1 1,1 1 1,1 0 1,0 1 1 1 0,0 1 1 1 1,0 0 1,0 1 0 1
```

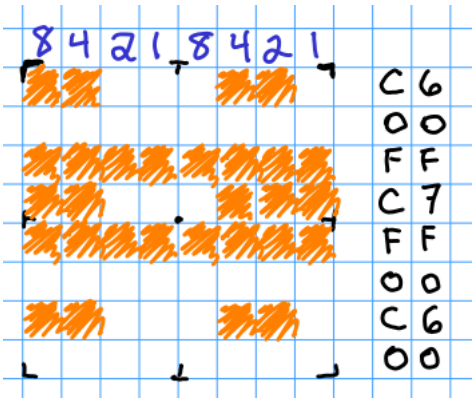
(b) The tree contains 14 distinct characters. If we were using a *fixed-width* encoding of the same characters, how many bits per character would we need? 4 bits ($2^4 = 16$)

(c) The message in part (a) is 9 characters. How many bits did we save by using a variable-width encoding instead of a fixed-width one? Saved one bit
Fixed would be $4 \cdot 9 = 36$ bits, variable was 35 bits.

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



7. 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          Output:
K: 1  2  3  4  5              10
```

8. What is the output of the following algorithm?

```
1. Set N to 13
2. Output N
3. If N = 1 then stop
4. If N is even, then set N to N/2
   Otherwise set N to 3*N + 1
5. Go back to step 2.
```

This algorithm is related to the *Collatz Conjecture*, which states that, no matter what integer N starts with, the sequence will also reach 1. In this case, starting with 13, we get:
13
40
20
10
5
16
8
4
2
1

9. What is the output of the following Python program?

```
alpha = 6
beta = 2
print "alpha + beta"
print alpha - beta
gamma = alpha - 1
print gamma*2
```

```
alpha + beta
4
10
```