Assignment 1

Thursday, January 13, 2022 2:57 PM

Encoding Numbers

example: $1101101_2 = 1\cdot2^0+0\cdot2^1+1\cdot2^2+0\cdot2^3+1\cdot2^4+1\cdot2^5 = 53$

(1)

→ The base-10 rep. of the integer whose binary rep. is 01011101_2 is 45.

$1\cdot2^0+0\cdot2^1+1\cdot2^2+1\cdot2^3+0\cdot2^4+1\cdot2^5+0\cdot2^6$

$1+0+4+8+0+32+0=45.$

→ The binary rep. of the integer whose base-10 is 154 is 10011010_2.

10011010_2

$154-128-16-8-2$

(2)

→ The base-10 rep. of the number whose binary rep. is 0101.101_2 is 5.625

first,

$1\cdot2^0+0\cdot2^1+1\cdot2^2+0\cdot2^3 = 5$

and

$1\cdot2^{-1}+0\cdot2^{-2}+1\cdot2^{-3} = 0.625$

→ The binary rep. of the rational number whose base-10 rep is 12.625 is 1100.101_2

from above, we know the decimal, 0.625, is written as $.101_2$. Now, to get 12...

$12-8=4$

$2^3 \quad 2^2$

$4-4=0.$

(3) **(3) (5 points)** Find an ASCII table online and use it to complete the following:

- Find the text corresponding to the following ASCII-encoded bytes:

77	97	116	104	33
01001101	01100001	01110100	01101000	00100001

- Find the binary representation of the ASCII-encoded bytes corresponding to the first four letters of your last name.

→ The ASCII-encoded bytes corresponds to Math!

→ The binary representation of "Howa" is

H	o	w	a
↓	↓	↓	↓
72	111	119	65
↓	↓	↓	↓
1000111_2	1101111_2	1110111_2	1000001_2

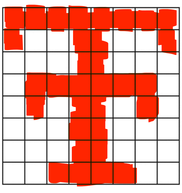
(4) **(4) (5 points)** A simple 8×8 monochrome image can be encoded in binary by indicating shaded pixels with a 1 and non-shaded pixels with a 0. Each row of pixels can then be represented in computer memory by a single byte, and the entire 8×8 image is represented by the sequence of those bytes, from the top row to the bottom row. For example, an image of the letter A might be encoded as:

Binary	hex	decimal
00011000	0x18	24
00100100	0x24	36
00100100	0x24	36
01000010	0x42	66
01111110	0x7E	126
10000001	0x81	129
10000001	0x81	129
10000001	0x81	129

So the image is encoded by the 8 bytes whose values in base-10 are:

24, 36, 36, 66, 126, 129, 129, 129

Draw an 8×8 pixel image of a monochrome 'double T' logo. Find the binary, hexadecimal, and decimal values of the bytes that encode it in the same way as above.



Binary	Hex	Decimal
11111111	0xFF	255
10011001	0x99	153
00011000	0x18	24
01111110	0x7E	126
01011010	0x5A	90
00011000	0x18	24
00011000	0x18	24
00111100	0x3C	60

(5)

→ The number of bytes in a

- kilobyte is 1024 bytes
- megabyte is 1048576 bytes
- gigabyte is 1073741824 bytes
- terabyte is 1099511627776 bytes
- petabyte is 1125899906842624 bytes

→ An Mb is a Megabit (1000000 bits) while a MB is a Megabyte (1000000 bytes). There are 8 bits in every byte so, a MB is 8 times the number of bits and thus 8000000 bits. (Indiana University IT Services)

→ first we should make a few assumptions... I will claim that the average word is 5 characters (according to Google). We should add 1 more character for spaces between. Now, if we say there are about 500 words per page, thus 100,000 words in 200 pages. Then that makes 600,000 characters. Now, according to Google there is typically 1 byte (8 bits) in 1 character so we'd have 600,000 bytes for a typical 200 page novel.

I just realized I neglected characters like ,.;:"'-, etc. say we'll add about 50 per page so another 10,000 characters which brings us to 610,000 bytes are required to ASCII-encode a typical novel of 200 pages.

Extra Credit:

5 is easy: 101.

0.4 isn't...

in binary...

0.1 is too big for 0.4 so

$0\cdot2^{-1}+1\cdot2^{-2}+1\cdot2^{-3}+0\cdot2^{-4}+0\cdot2^{-5}+1\cdot2^{-6}+1\cdot2^{-7}+0\cdot2^{-8}+0\cdot2^{-9}+1\cdot2^{-10}+1\cdot2^{-11}$

$= 0 + .25 + .125 + 0 + 0 + .015625 + .0078125 + 0 + 0 + .00976625 + .00048828125$

$= 0.39990234375 \approx 0.4$

(final calculations with some help from Desmos Calculator (Scientific))

101.01100110011_2 is the (approx.) binary representation of 5.4

if you go further...

2^{-11}+2^{-10}+2^{-7}+2^{-6}+2^{-3}+2^{-2}+2^{-14}+2^{-15}+2^{-18}+2^{-19}+2^{-23}+2^{-24}+2^{-25}+2^{-26}+2^{-27}+2^{-28}+2^{-29}+2^{-30}

^ this is 0.3999999856