CS101
Data Structures and Algorithms

Lecture 1

Data Representation – Part 1

Binary encoding

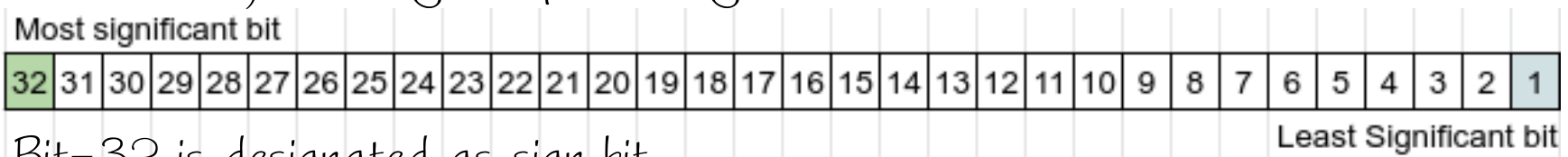
- Bit is the unit of binary data
- Bit → Binary Digit (1 / 0)
- In the H/W bit-1 is +5Volts, and bit-0 is -5Volts.
- This is the most fundamental encoding
- A sequence of 8 bits is called Byte
- Numbers, Text, Image, Video etc all have binary representation.

1: Integers

- A sequence of bits is essentially a binary number.
- Every binary number has a decimal value
- Eg., 101101 is 44 and 1110 is 12?
- Thus non-negative integers have a natural representation in the digital h/w as binary number
- What do we do for representing negative integers?

Negative Integers

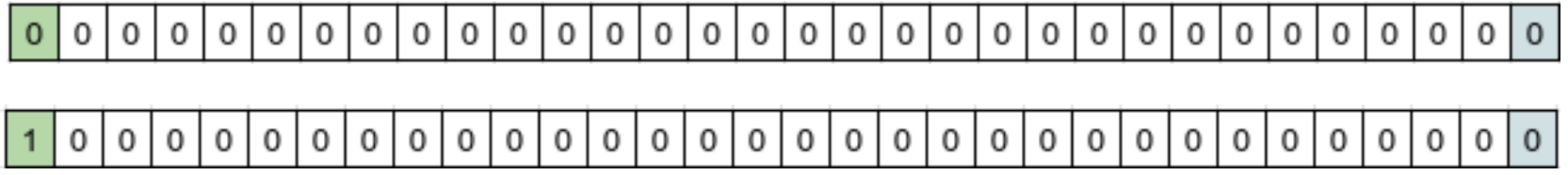
- First Encoding Attempt : Dedicate a bit for sign: 0 => +, 1 => -
- Which bit is the question?
- Thus we need to first fix the width of the integer representation.
- The width of an integer in previous generation is 32 and now is 64,



- Bit-32 is designated as sign bit
- Bits 1->31 is treated as magnitude in the first encoding attempt.

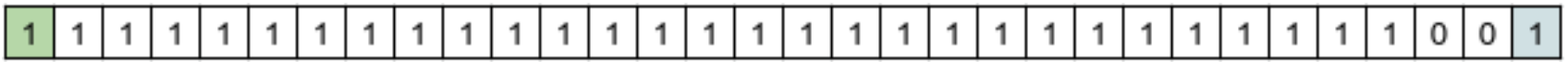
Sign-magnitude representation

- Magnitude can be from 0 to (2^31 – 1)
- Bit-32 as sign bit we get both +ve and -ve nos.
- The downside of this representation is that ZERO has two representation +0, and -0

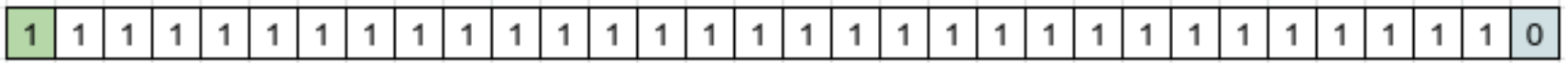


2's compliment representation

- Positive numbers have straightforward binary number.
- For negative numbers, first compliment the number and then add 1.
- eg., Given 7 (00000...0111) to compute -7.
- Compliment 7, and then add 1. The result is,



- 7-8 = 7 + (2's Compliment of 8) + 1 and the result is



- Verify that 0 and -0 has the same representation

2: Real Numbers

- 213 . 6454 has integral and fractional part
- 1110011 . 0000100101 also has integral and fractional part.
- This format however is not applicable for real number in binary format
- Why?
- IEEE-754 format for floating point representation is the adopted standard

IEEE-754 Floating Point Format

- For a 32 bit real number (single precision)
  - 23 bits are for normalized mantissa
  - 8 bits are for exponent
  - 1 bit for sign (The most significant bit)
  - This representation assumes that the integral part is always 1, and hence the phrase "normalized mantissa"
  - Check yourself for double precision, More nuances about this format can be read here in wiki.
- We cannot represent every real no. in this format, and hence rounding

3: Text Representation

- To take stock of atoms
  - 26 alphabets, 10 digits, operators, punctuation, braces, limited set of symbols like $#@! &~
  - The above is collectively called as list of "characters".
  - Yes, all that we can type from the regular PC keyboard.
- We need an encoding for all of them
- Meaning, we need to assign a unique binary sequence to each of these, as computers can understand only binary

3: Text Representation

- ASCII – is the standard that is adopted
- See the ASCII chart here or elsewhere
- Note that it also has representation for a few control characters like bell, and non-printable characters like tab, newline etc, and special characters like NULL etc
- Any length text is a sequence of ASCII encoded text characters.

3: Text File on Disk

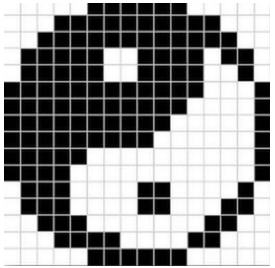
- On disk the files are stored in pages, each of size 4KB (in general, but reconfigurable while building the kernel)
- Text files (or in general files) are stored in unit of pages on disk.
- The pages need not be contiguous.
- Then what decides the end of file?
- The NULL character in the ASCII code sequence marks the end of file.
- Applications (like Libreoffice, MSWord, vi, notepad) respect this EOF while performing R/W operations on disk.

3: Text File on Disk

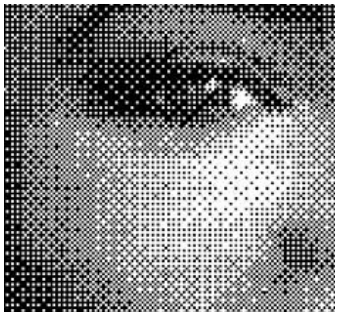
- Question: Why dont we need a begin-of-file marker/symbol/character?
- Disk file manager maintains a mapping of all files to its starting location or address on disk
- Typical location on disk will be in terms of Cylinders, Tracks/Head, and Sectors (CHS).
- In modern disks we have LBA (Logical Block Addressing).
- A simple formula $A = (c \cdot Nheads + h) \cdot Nsectors + (s - 1)$, translates CHS to LBA.

4. Binary Image

- How to encode this 16x16 binary image?

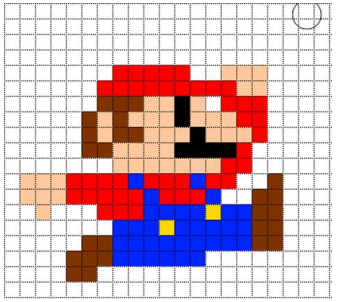


bitsofbytes.co



quora

- Increasing the number of colors, How to encode?



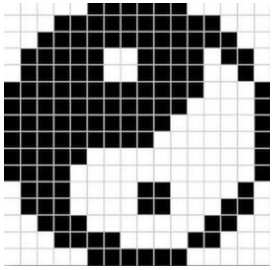
bitsofbytes.co

- How to encode True-color images (TCI)?

NASA

- What about size of TCI?

4. Binary Image



- How to encode this 16x16 binary image?
- A text file (bitmap file) containing the size in the first line, and rowwise bit data as hexadecimal values as shown here for the eighth row.

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

`0xFFC1`

5. Video file

- A naive construction of a video is from a sequence of image files
- As True color images (TCI) are themselves large in size in naive encoding, Video files will explode.
- This means TCIs and videos need to be compressed.
- Discrete Cosine Transform technique is employed for compression
- AVC, HEVC, HEIF are popular video encoding formats
- JPEG, MPEG, MP4 are all container formats.

Applications

- The applications Libreoffice, Image viewer, VLC, Photoshop are all codecs in one or the other sense.
- CODEC is the "portmanteau" for Coder / Decoder.

Operators

- Arithmetic operators +,-,*,/,=
- Bitwise Operators: AND (&), OR (|), XOR (^)

```
1101 & 0111 = 0101
0101 | 1101 = 1111
1011 ^ 0101 = 1110
```

- Logical Operator: AND (&&), OR(||), NOT (!)

  X && Y = 1 if both X,Y are non-zero, 0 if either one is zero

  X || Y = 1 if either of X,Y is non-zero, 0 if either one is zero

  !X = 0 if X is non-zero, 1 if X is 0

Operators

- Shift Operator: <<, >> applicable "only to" integers
  - Right Shift: `x << 2` : 2-MSBs will be lost
  - Left Shift : `y >> 3` : 3-LSBs will be lost
  - Right shift of `x` by one bit location is akin to `(2x mod 2^31)`
  - Left shift of `y` by one bit location is akin to `floor(y/2)`
- Relational operators can be meaningfully applied to the character class `'k' < 'n'`
  - Other relational operators: `<, >, >=, <=, ==, !=`

Address and Data Bus

- CPU is 32-bit processor means the following -
- The address of memory locations is 32 bits
- That is, the width of the address bus (lines or wires) is 32
- 32-bits processors can work with only 4GB RAM (why?)
- Typically the data bus width is also 32 bits for a 32-bit processor.
- CPU is byte addressable means the address points to one byte data which can be read and written in memory

Intuition for Address

- For example, 1GB RAM provides 2^30 memory location
- Each memory location has eight bits, or otherwise called 1 Byte
- Byte addressable CPU can read or write only one memory location at a time
- Hence CPU should be able to uniquely "refer" a memory location
- The "reference" to memory location is technically called an Address

Intuition for Address

- Therefore Address must have at least 30 bits for 1GB RAM.
- In other words Address bus must have at least 30 lines (wires)
- 32-bit processors can address 2^32 memory locations, and hence has 32 address lines
- 64-bit processors will have 64 address lines.
- What is the maximum size of RAM usable with 64-bit processors?

Data Types

- Basic Data types
  - Char / Byte
  - Integer / Long
  - Float / Double
- Derived Data Types
  - String
  - Record