18 57 63 00 82 FF 60 00 A2 20 1C A8 00 10 60 B2 FE FF 07 00 68 F3 26 DA 8B A3 F8 F8 00 00 00 00

000

1100\_0010\_ 1011\_1011 0001\_1000\_0000\_0100

0\_0010

0 – Not last buffer

00 – No compression

0x18: 0001 1000

0x57: 0101 0111

FCHECK: 0b10111 – 23

FDICT: 0b0 – 0

FLEVEL: 0b01 – 1

CM: 0b1000 – 8

CINFO: 0b0001 – 1

NOTES: Read from least significant bit

{7 6 5 4 3 2 1 0}

Start from 0 and go upward.

This is the zlib format, so it has its own header. The first two bytes do that.

Next is the compressed data and then a checksum.

0x63 – 0110 0011

BFINAL: 0b1 – 1 (TRUE)

1111100011111000101000111000101111011010001001101111001101101000000000000000011111111111111111101011001001100000000100000000000010101000000111000010000010100010000000000110000011111111100000100000000001100011

1

01

00110

00000

0000

10000011111111100000110000000000100010100000100001110000001010100000000000010000000011001001101011111111111111111100000000000000001011011001111011001000101101111010001110001010001111100011111

10010001

00111010

00000000

Decompressing Deflate with fixed length codes

All values between 0 – 143 are 8 bits and are between these values:

0b00110000 - 48

To

0b10111111 – 191

All values between 144 – 255 are 9 bits and are between these values:

0b110010000 - 400

To

0b111111111 – 511

All values between 256 – 279 are 7 bits and are between these values:

0b0000000 - 0

To

0b0010111 – 23

All values between 280 – 287 are 8 bits and are between these values:

0b11000000 - 192

To

0b11000111 – 199

You should build a tree for this however, you don’t have to since codes have a bit of consistency

A few if statements would fix that.

Some constants for you

0 – 255 (Literals)

256 (Stop code)

257 – 285 (Back Reference) (Starts with 1)

0 – 29 (Copy Distance) \*Always after back reference

(Starts with the distance 3)

Note that back references and copy distances may contain extra bits after the initial code.

You treat the extra bits as a separate number and then add later.

Example

Code 277 has a default length of 67 with 4 extra bits.

If the 4 bits are 0b1110

Then you add 14 to 67 since 0b1110 is 14.

Your final length is 81.