CS 631-01 Conversions & Evaluation

Lab 01 & Lab 02

Project 01 starter

Conversions

321

Binary - base 2

$3 \times 10^2 + 2 \times 10^1 + 1 \times 10^0$

0b 0101

0b 1101
32 16 8 4 2 1

$$(1) 2^3 + (1) 2^2 + 0 (2^1) + 1 (2^0)$$
$$8 + 4 + 0 + 1 = 13$$

Hexadecimal (Hex) - base 16

0 - 9   A - F   a - f
       10   15

0x F3B

$$F \times 16^2 + 3 \times 16^1 + B \times 16^0$$
$$15 \times 16^2 + 3 \times 16 + 11 \times 1$$
$$15 \times 256 + 48 + 11$$
$$3840 + 48 + 11 = 3899$$
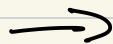
Converting HEX to BIN and BIN to HEX

HEX   0 x F3B

↓ ↑

BIN   0x 1111 0011   1011
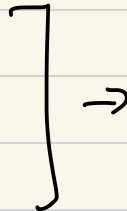
Conversions

"214"
"0b1101"   ⟶   uint32_t
"0xF3B"

parse_operand()

$\left.\begin{array}{l} \text{TK\_INTLIT} \\ \text{TK\_BINLIT} \\ \text{TK\_HEXLIT} \end{array}\right\}$ →

" 1011 "
_____

$s[0]$, $s[1]$ $s[2]$

char *s = " 1011 "
uint32t tmp = 8; 23 10
uint32t value = 0;

char d ;

'0' = 48
'1' = 49

d = s[0];

tmp   = $\boxed{d - '0'}$; binary
                             decimal
value = tmp;   Hexadecimal

d = s[1]

tmp = d - '0';

value = (value << 1)

      = value * $\boxed{\underset{base}{2}}$ + tmp

Nalue = 2

$d = s[2]$

$tmp = d - '0'$

$tmp = 1$

$value = value * \boxed{2} + tmp$

$\qquad\quad 4 \qquad + \quad 1$

$\qquad = \quad \underline{5}$

$d = s[3]$

$tmp = d - '0'$

$tmp = 1$

$value = value \times \boxed{2} + tmp$

$\qquad\quad 10 \qquad + \quad 1$

$\qquad = \enclose{circle}{11}$

---

ntlang

nt -b 10 "241"

$\underline{241}$

scan

parsed $\rightarrow$ uint32_t

$\downarrow$

str

uint32_t to str
base 10

uint32_t value;

char out[64];

Out[0] = '2'
out[1] = '4'
out[2] = '1'
out[3] = '\0'   0

240①
↑
uint32_t tmp

241 / 10 = 24
241 % 10 = 1

tmp = value % (10)

out[0] = '0' + tmp   (1)
value = value / (10)   base
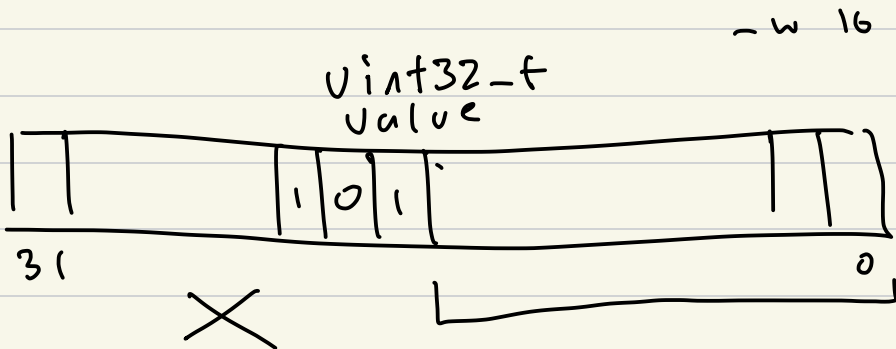tmp = value % (10)
out[1] = '0' + tmp   (4)

⠇

width   -w   4 , 8, 16, 82
only for   -b 2   -b 16

nt -b 16 -w 4 -e " 0xAC "
0xc

nt -b16 -w $\frac{4}{8}$ -e " 0xAC >> 4 "
0xA
0x0A (8)
nr -b 2 -w 4 -e " 0xAC >> 4 "
___

0b 1010

                                    -w 16
                    vint32_t
                    value

31   0

        ✗

value = value & 0xFFFF

Mashing

~w4   0xF   0b1111

-w8   0xFF   0b1111 1111

-w16   0xFFFF   0b1111 1111 1111 1111

Compute the mask

width = 4

mask =   width

                  dec   bin

0x10   =   16   0b10000

-    1   -1   -1

0x0F   15   0b01111

mask = (width << 0b1) - 1;

signed output

nt -e "-3"
-3

nt -b 2 -w 4 "=3"
0b 1101   0b0011
                              inv 0b1100
                                + 0b 1101

nt -b 2 -w 8 "-3"
0b 1111 1101

nt -b 10 -w 4 "0b1101"
-3

nt -b 10 -w 8 "0b1101"
13

nt -b 10 -e "0xFFFFFFFF"
-1

n t -6 10 -c "0xFFFFFFFF" -0

42 .. ---

1 + 2 + 3