

CS 208

F 19 Sep 2025

0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
A	1010
B	1011
C	1100
D	1101
E	1110
F	1111

0xB6F

$\swarrow$ $\swarrow$ $\swarrow$
 1011 0110 1111

101010

32 16 8 4 2 1

$32 + 8 + 2 = 42$

0010 | 1010
 2 A

0x2A

$x = 000 \dots 011000$
16 8 4 2 1

24

$y = x \gg 2;$

$0000 \dots 0110$
4 2 1

6

$\gg 2$
= division by 4

$\gg 3$
= div. by 8

etc.

$\ll 3$... mult. by $8 = 2^3$

$x = 000 \dots 011101$
0 0 0 1 1 1

$x \gg 3 \sim x/8$ (int division)

Suppose I want the remainder?

$x \% 8$

$x \& 0x07$
0b111

"masking"

Shifting left (sometimes) used
for mult. by powers of 2
Shifting right ... division...

& Sometimes used for masking
(i.e. turning some bits into 0's)

Special Bits 0xffca3fff

1111 1111 1100 1010 0011 1111 1111 1111

0xffffffff

```
int x = ~(0);  
return x;
```

Sadly, this
is not
the goal

"Word" means

Heads
-up

32-bit quantity

Constants are OK, but only
1-byte constants like 0xb0