# Decimal & Binary.

$$(x)_{10} \qquad (x)_2$$

**Q.** n = 10 to binary.

| 2 | 10 | |
|---|----|---|
| 2 | 5  | 0 |
| 2 | 2  | 1 |
|   | 1  | 0 |

$(10)_2 \Rightarrow (1010)$

**Q.** Num is odd or even

**Ans.** $n \& 1$ if 1 then odd
0 then even

→ because it converts n into binary. like
000101 then do and with 1 which
will be with last bit.

```
    000101
&   000001
    ─────────
    000001  → that mean it is odd
```

**Q.** Reverse the number in binary form.
ex. n = $(110)_2$ then ans = $(011)_2$ (input like 6)

**Ans:**
```
        ans = 0
    while( n! = 0 ) {
        bit = n & 1;
        n = n >> 1;
        ans = (ans × 10) + bit
```

$y$

at last it will return ans in binary form.

Q. Give the binary representation of decimal number
(input like 12)

```
ans=0; i=0;
while(n!=0) {
    bit= n & 1;
    ans= ( bit x pow(10,i)) +ans;
    i++;
    n =n>>1;
}
```
$y$
```
cout << ans <<endl;
```

Q. binary of negative number.(input like -6)

→ ignore -ve
→ take out it's binary value.
→ make 1's comp.
→ add 1 in it

Q Binary to decimal (input is in binary)
→ take its digits     d= n% 10;
→ ans= ( d x pow(2,i) )+ ans
→ n /=10.