

## Bit Manipulation - II

**Prerequisites:** knowledge of binary number system

### Count set bits

$n \& (n - 1)$  sets the first set-bit to zero.

Explanation:  $n = \text{XXX}100$

$n - 1 = \text{XXX}011$

$n \& (n - 1) = \text{XXX}000$

```
int numberOfones(int n) {  
    int count = 0;  
    while (n) {  
        n = n & (n - 1);  
        count++;  
    }  
    return count;  
}
```

### Power of two

From our past knowledge of the binary number system,

Numbers of the type  $2^n$  have only 1 set bit.

Explanation:  $n = 000100$

$n - 1 = 000011$

$n \& (n - 1) = 000000$

$!(n \& (n - 1)) = 000001$

If the number only had one set bit, then  $n \& (n - 1)$  would be zero.

```
bool ispowerof2(int n) {  
    return (n && !(n & n - 1));  
}
```

## Generate Subset

Explanation: if the  $j^{\text{th}}$  bit is set, then we take the  $j^{\text{th}}$  element.

There are a total of  $2^n$  subsets.

```
void subsets(int arr[], int n) {
    for (int i = 0; i < (1 << n); i++) {
        for (int j = 0; j < n; j++) {
            if (i & (1 << j)) {
                cout << arr[j] << " ";
            }
        } cout << endl;
    }
}
```

Practice Questions:

1. [Counting bits](#)
2. [Power of four](#)