Topics to discuss

What is bitmasking ?

Some operations useing bitmaking.

- set a bit
- unset a bit
- Toggling a bit
- checking a bit.

Summary.

Source code.

YouTube @StartPracticing

What is bit masking ?

→ Bit masking is a technique used in computer programming that involves manipulating individual bits within a binary number.

In this technique, specific bits are "masked" or selected using bitwise operations to perform various operations like setting, clearing, toggling specific bits, checking a bit.

YouTube @StartPracticing

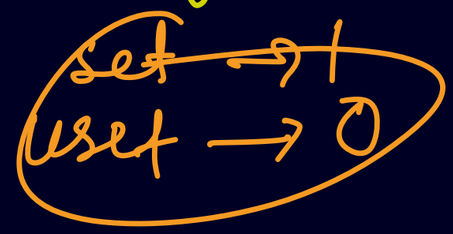
Some common operations using bitmasking.

→ Set a bit

→ Unset a bit / clear a bit

→ Toggling a bit.

→ Checking a bit



YouTube @StartPracticing

1. Set a bit :

Q:- $n = (45)_{10} = (101101)_2$, set the 1st bit and 3rd bit from right

Note: The position of LSB is 0, 2nd bit is 1 and so on.

$n = \underset{5}{1}\ \underset{4}{0}\ \underset{3}{1}\ \underset{2}{1}\ \underset{1}{0}\ \underset{0}{1}$, for 1st bit

$1 << 1 = 0\ 0\ 0\ 0\ 1\ 0$

$n \mid (1<<1) = 1\ 0\ 1\ 1\ 1\ 1$ = Ans.

YouTube @StartPracticing

n = 1 0 1 1 0 1 , for 3rd bit.

5 4 3 2 1 0

$1 << 3$ = 0 0 1 0 0 0

$n \mid (1 << 3)$ = 1 0 1 1 0 1 = Ans

In general,

$n \mid (1 << K)$; $0 \leq K < 32$

YouTube @StartPracticing

2. Unset a bit / clear a bit :

Q:- Given $n = (45)_{10} = (101101)_2$, Unset 2^{nd} and 4^{th} bit

Solution: $n = \underset{5}{1}\ \underset{4}{0}\ \underset{3}{1}\ \underset{2}{1}\ \underset{1}{0}\ \underset{0}{1}$, for 2^{nd} bit

$1 << 2 = 0\ 0\ 0\ 1\ 0\ 0$

$\sim(1 << 2) = 1\ 1\ 1\ 0\ 1\ 1$

$n\ \&\ \sim(1<<2) = 1\ 0\ 1\ 0\ 0\ 1$ = Ans.

YouTube @StartPracticing

$n = \underset{5}{1}\ \underset{4}{0}\ \underset{3}{1}\ \underset{2}{1}\ \underset{1}{0}\ \underset{0}{1}$, for 4th bit.

$1 << 4 = 0\ 1\ 0\ 0\ 0\ 0$

$\sim(1 << 4) = 1\ 0\ 1\ 1\ 1\ 1$

$n\ \&\ \sim(1<<4) = 1\ 0\ 1\ 1\ 0\ 1$ = Ans.

In general, $n\ \&\ \sim(1<<K)$

YouTube @StartPracticing

3. Toggling a bit :

Q: Given $n=(45)_{10} = (10110)_2$, Set the K^{th} bit to 1 if it is 0 and set 0 if it is 1. ($K=3$ & $K=4$)

Solution:

$$n = \underset{5}{1}\ \underset{4}{0}\ \underset{3}{1}\ \underset{2}{1}\ \underset{1}{0}\ \underset{0}{1}$$, for 3rd bit.

$$1<<3 = 0\ 0\ 1\ 0\ 0\ 0$$

$$n \wedge (1<<3) = 1\ 0\ 0\ 1\ 0\ 1 = \text{Ans.}$$

YouTube @StartPracticing

$n =$ 1 0 1 1 0 1 , for 4^{th} bit

(5 4 3 2 1 0)

$1 << 4 =$ 0 1 0 0 0 0

$n \wedge (1 << 4) =$ 1 1 1 1 0 1 = Ans.

In general, $n \wedge (1 << K)$

YouTube @StartPracticing

4. Checking a bit :

Q: Given, $n = (45)_{10} = (101101)_2$, Check the k^{th} bit ($K=4$ & $K=5$)

Solution:-

$n =$ 1 (5) 0 (4) 1 (3) 1 (2) 0 (1) 1 (0) , for 4^{th} bit

$1 << 4 =$ 0 1 0 0 0 0

$n \& (1<<4) =$ 0 0 0 0 0 0 = Ans.

$(n \& (1<<4)) == 0 \longrightarrow$ Bit is unset.

YouTube @StartPracticing

n = 1 0 1 1 0 1 , for 5th bit

5 4 3 2 1 0

$1 << 5$ = 1 0 0 0 0 0

$n \& (1<<5)$ = 1 0 0 0 0 0

$n \& (1<<5) != 0 \longrightarrow$ Bit is set

In general,

$$n \& (1<<K) = \begin{cases} 0, & \text{bit is unset} \\ !=0, & \text{bit is set} \end{cases}$$

YouTube @StartPracticing

Summary

The position of LSB is 0, 2nd bit is 1 and so on.

Set K^{th} bit : $n \mid (1 << K)$; n is a number

Unset K^{th} bit : $n \,\&\, \sim(1 << K)$

Toggling K^{th} bit : $n \wedge (1 << k)$

Checking K^{th} bit : $n \,\&\, (1 << k) = \begin{cases} 0, & \text{bit is unset} \\ !=0, & \text{bit is set} \end{cases}$

YouTube @StartPracticing

Source code

I/p : n = 45

K = 3

O/p : setbit = 45

unsetbit = 37

Toggling = 37

Checkbit = True.

n = 1 0 1 1 0 1

Set = 45

unset = 1 0 0 1 0 1

= 37

Toggling = 1 0 0 1 0 1

= 37

Checkbit = True.

YouTube @StartPracticing

Follow Now

Start Practicing

i._am._arfin

Arfin Parween

arfin-parween

YouTube @StartPracticing