



{2, 3, 65}



最大的数是64 :  $\text{data}[64/32+1]=\text{data}[3]$ :

data[0](int): 0000 0000 0000 0000 0000 0000 0000 1100 0~31 0~8

data[1]: 0000 0000 0000 0000 0000 0000 0000 0000 32~63

data[2]: 0000 0000 0000 0000 0000 0000 0000 0010 64~95

data[3]: 0000 0000 0000 0000 0000 0000 0000 0000

2肯定就在data[0] 里面 :

3

65

$2/32=0$  说明放在data[0]  $2\%32=2$  说明在data[0]的第3个位置 (下标从0开始)

$65/32=2$  说明放在data[2]  $65\% (取余) 32=1$  说明在data[2]的第2个位置

计算空间 : 0~2亿 最大的那个数就是2亿

2亿=M

开2亿个int数组 :  $2\text{亿} \times 4 (\text{byte}) / 1024 / 1024 = 762\text{M}$ , 用下标的形式来判读就需要这么多内存。

如果用bitMap :  $2\text{亿}/32 \times 4 (\text{byte}) / 1024 / 1024 = 762/32=23\text{M}$  (因为最后我们还是会用int来表示bit), 只不过我们是开了 `int data[max/32]`.

判断66存不存在 :  $66/32=2$  我们知道存在data[2]里面  $66\%32=2$  再去data[2]里面找第2位 也就是第3个位置 看是不是1

如果上面的数组 我们不用int来存 用byte数组 那么你们说那里会有变化 ?