한국외국어대학교
HANKUK UNIVERSITY OF FOREIGN STUDIES

# Introduction to Computers & Lab

## # Lab 08

2021.04.29

Prof. Muhammad Bilal
TA. Sohee Jang

# Index

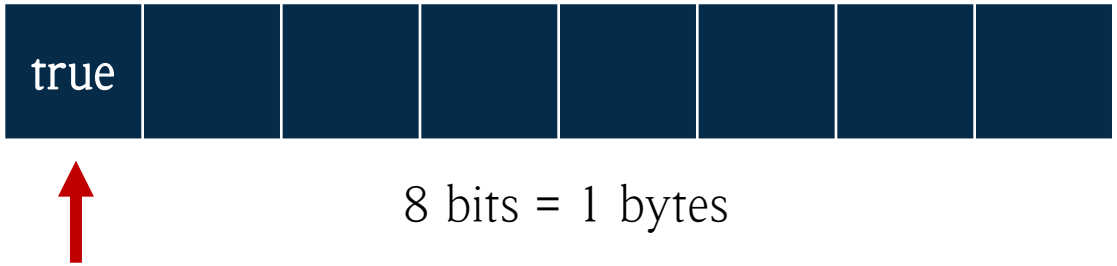
# Bitwise operators

한국외국어대학교
HANKUK UNIVERSITY OF FOREIGN STUDIES

- In the past, memory was so expensive that computers didn't have much memory. Therefore, there has been an attempt to use up all available memory.

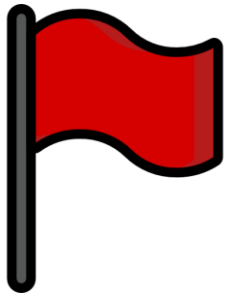
- Example

boolean

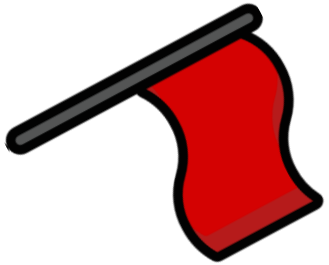



: True and false use only one bit, but account for one byte (8 bits). The smallest unit of memory is one byte, which uses one bit and wastes seven bits.

# Bit flags



Flag on = 1



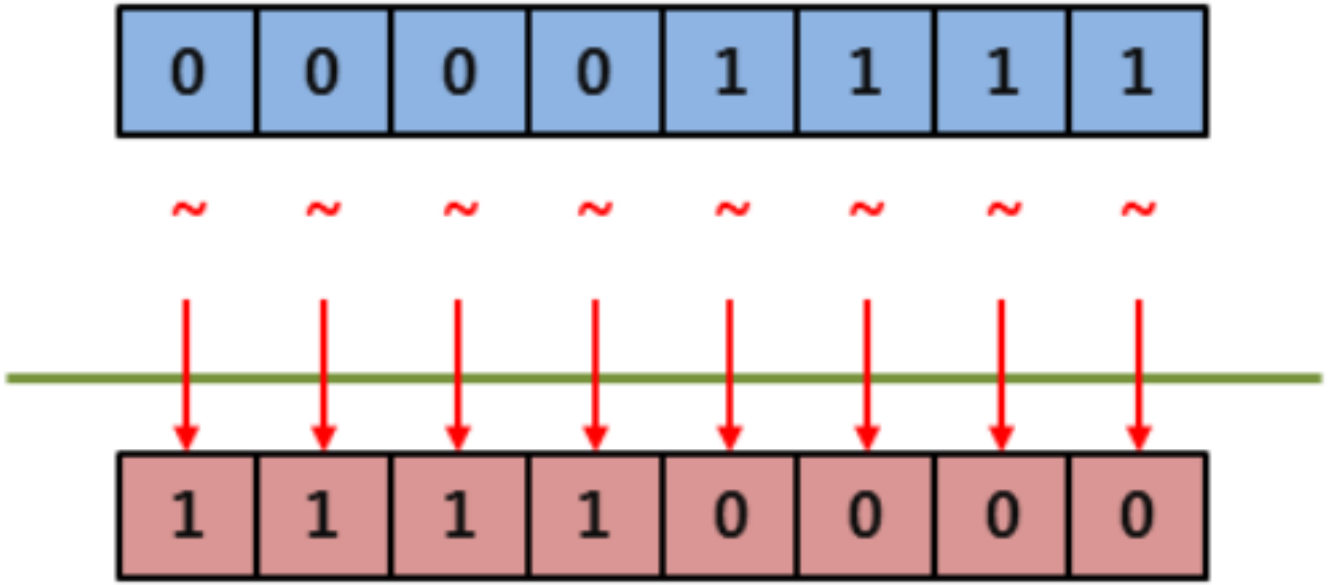
Flag off = 0

The individual bits of a byte are called bit flags.

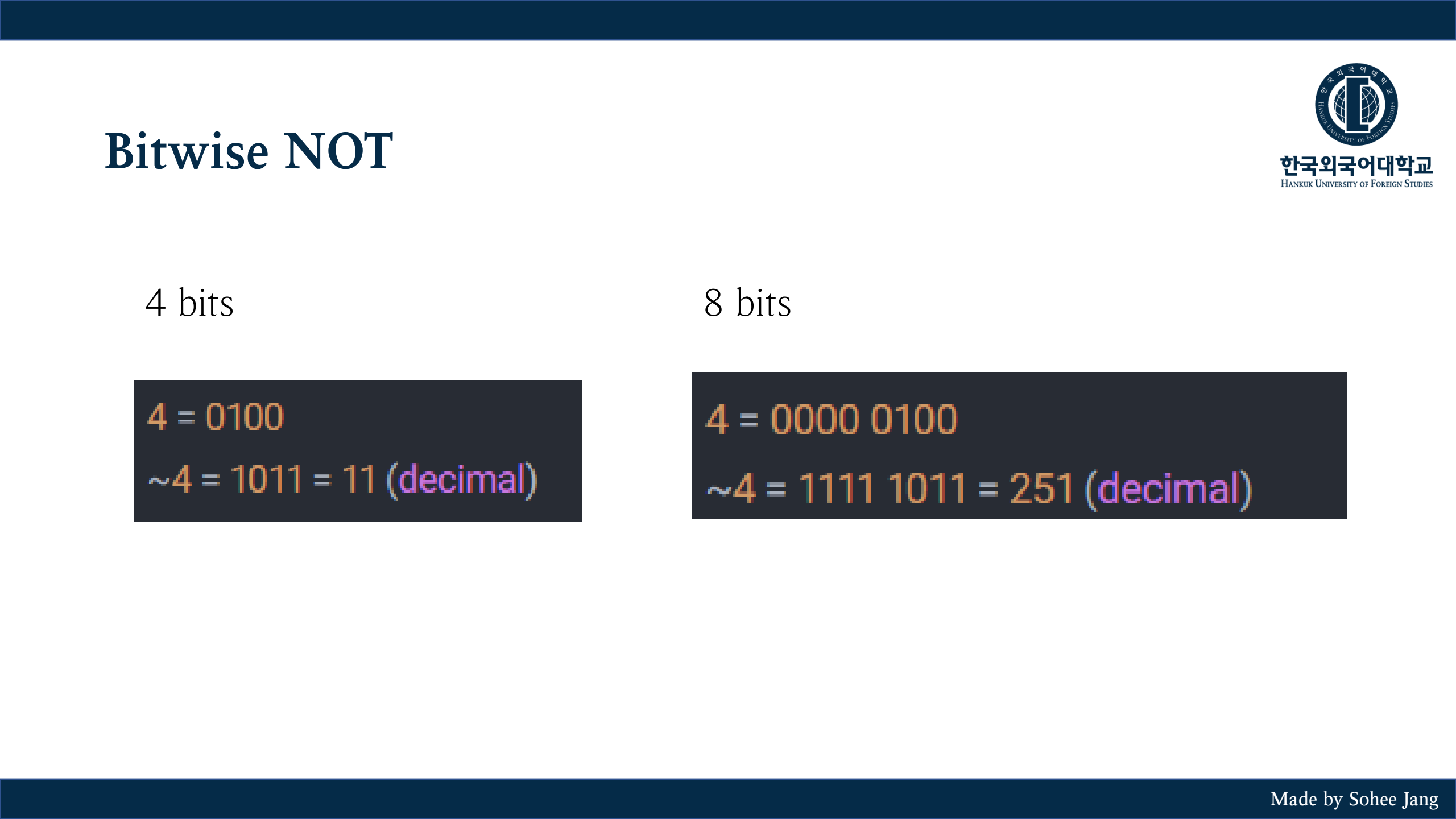
# Bitwise operators


| Operator | Symbol | Form |
|---|---|---|
| Bitwise NOT | ~ | ~x |
| Bitwise AND | & | x & y |
| Bitwise OR | \| | x \| y |
| Bitwise XOR | ^ | x ^ y |
| Left shift | << | x << y |
| Right shift | >> | x >> y |

# Bitwise NOT



# Bitwise NOT



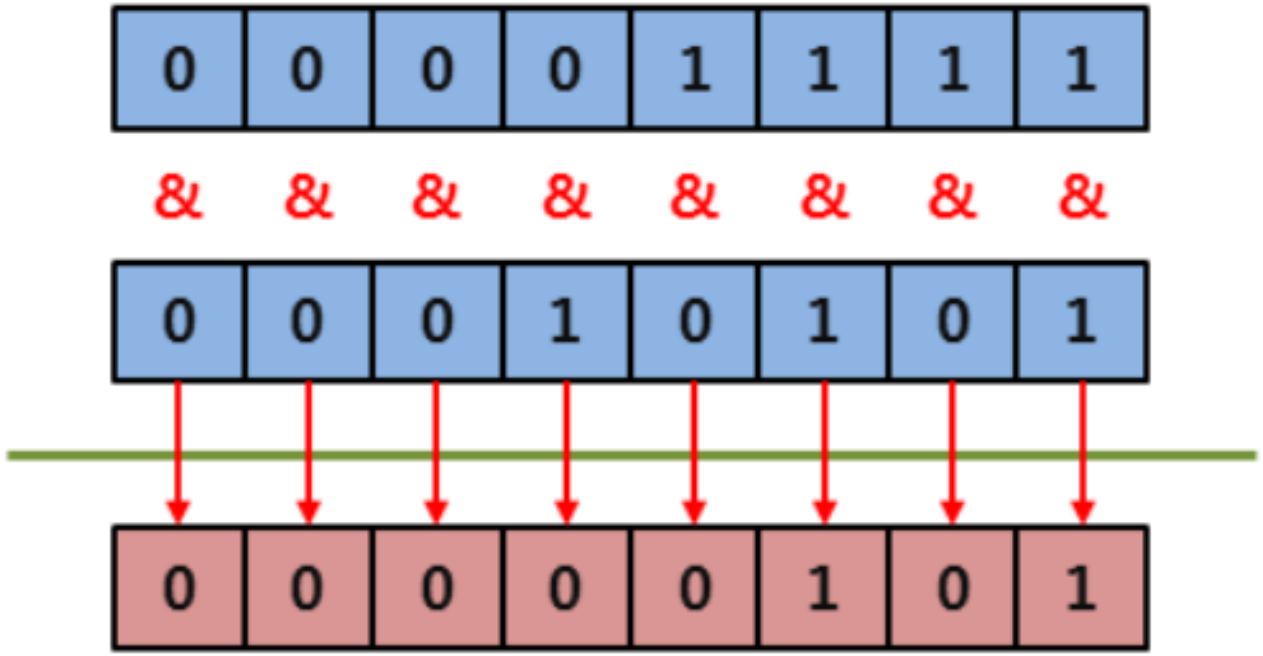

4 bits

```
4 = 0100
~4 = 1011 = 11 (decimal)
```

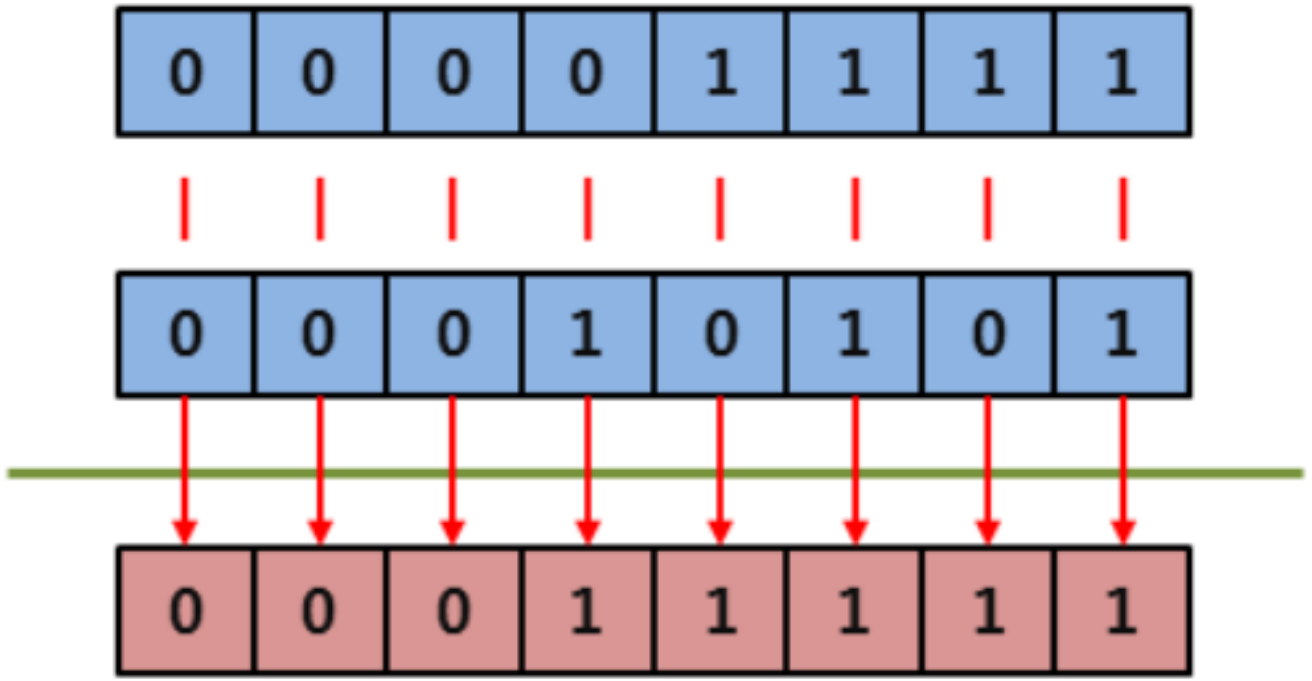
8 bits

```
4 = 0000 0100
~4 = 1111 1011 = 251 (decimal)
```

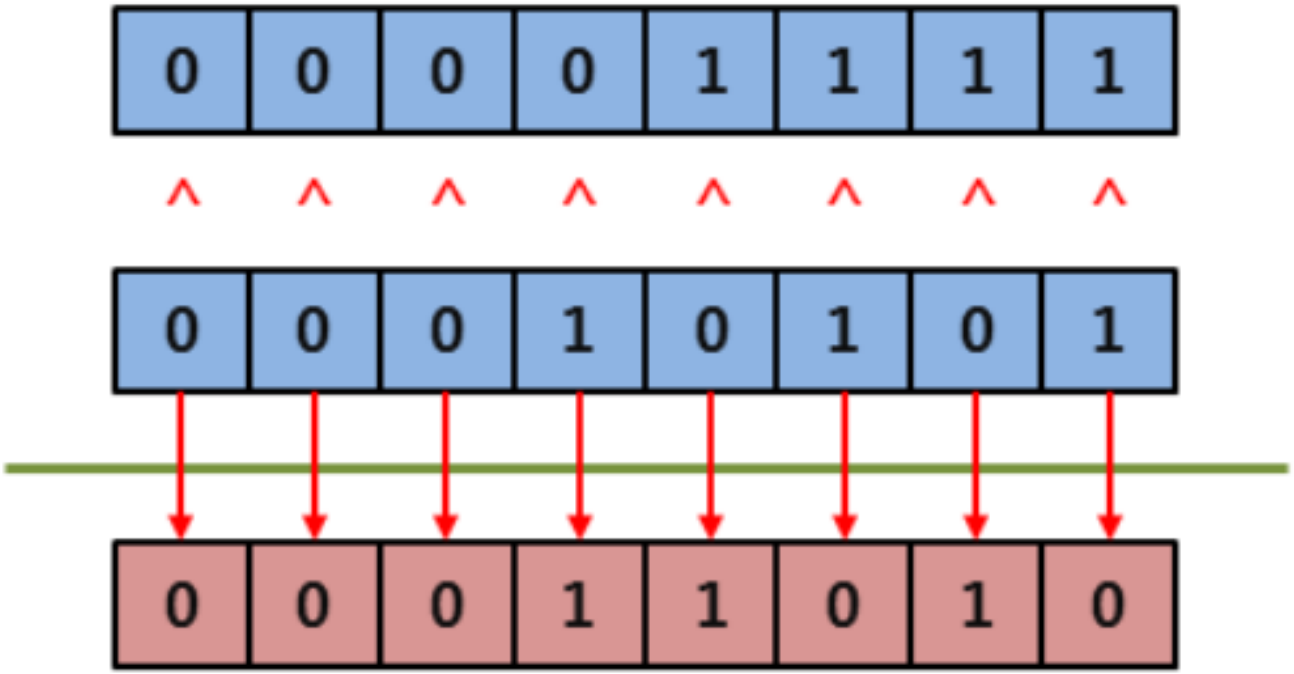
# Bitwise AND



# Bitwise OR



# Bitwise XOR



# Bitwise AND, OR, and XOR

한국외국어대학교
HANKUK UNIVERSITY OF FOREIGN STUDIES

0 1 0 1 (5)
0 1 1 0 (6)

5 & 6

0 1 0 1 (5)
0 1 1 0 (6)
---

5 | 6

0 1 0 1 (5)
0 1 1 0 (6)
---

5 ^ 6

0 1 0 1 (5)
0 1 1 0 (6)
---

# Left shift & Right shift



## Left shift

```
3 = 0011
3 << 1 = 0110 = 6
3 << 2 = 1100 = 12
3 << 3 = 1000 = 8
```

## Right shift

```
12 = 1100
12 >> 1 = 0110 = 6
12 >> 2 = 0011 = 3
12 >> 3 = 0001 = 1
```

# Task 1 : Flip bits



Write a program to flip bits of a binary number using bitwise operators.

★ 0 become 1, and 1 become 0

★ using bitset library -> flip()

# Task 2 : zero count



Write a program to count trailing zeros in a binary number.

★ Number of consecutive zeros from end

★ Left shift ( >> ) by 1 while n & 1 == 0

# Task 3 : Rotate



Write a program to rotate bits of a given number.

★ left shifted, right shifted => print both values

# Task 4 : Even/Odd



Write a program to check whether a number is even or odd using bitwise operators.

★ using AND operator -> n & 1

Example

(11) 1011 -> 1 & 1 = 1 (odd)

(10) 1010 -> 0 & 1 = 0 (even)