Bitwise operators/bitmanipulation/bitmasking – They are operations that affect bits

* \*Another quick reminder; Graphical user interface, text, application, email

  Description automatically generated
  + This same function that transform from decimal to binary and from binary to decimal!
* \*Quick reminder : 0 is false, 1 is true
* And operator (&) and Or (|)

A picture containing diagram

Description automatically generated - XOR operator (~) if one of the bits are different, then it returns true. (1)

A picture containing calendar

Description automatically generated

* Thus, if we have 5&7
  + The value of 5 in binary is 101
  + The value of 7 in binary is 111
    - If we do 101&111:
      * Icon

        Description automatically generated with medium confidence
  + PROOF in vscode:
    - Text

      Description automatically generated with medium confidence
* Left shift operator (<<)
  + Text

    Description automatically generated
    - The left shift can also be seen as the multiplication of 2the value that we are left shifting. Ex: 5<<1 is 5 \* 21.
      * Formula: a<<n 🡪 a\*(2n)
* The right shift operator (>>)
  + Formula: a>>n 🡪 floor(a/(2n))
    - Text

      Description automatically generated
* Not operator (~)
  + This flips all the 0 to 1 and 1 to 0
    - Here is also why it gives a negative number!
      * Because the bit that represents the negative number value is turned into a 1!
    - [Representation of Negative Binary Numbers - GeeksforGeeks](https://www.geeksforgeeks.org/representation-of-negative-binary-numbers/#:~:text=Representation%20of%20Negative%20Binary%20Numbers.%201%20Signed%20Magnitude,in%20case%20of%20negative%20numbers.%20We%20...%20)
    - Graphical user interface, text, application, email

      Description automatically generated
    - A screenshot of a computer

      Description automatically generated with medium confidence