I'm Gaurav, 7 yrs exp, backend development, **XYZ** company, worked with product based and startups.

- x -> y
- x -> a -> y
- exp ?
- ctc ?
- Getting interview ?
- Preparation ?

Code reviews ?
Do I need to rememember langauge syntax ?
Which language to use ?
Do you like WFH or Office ?

## Welcome

1. Intro
2. Git repo
3. Create a Leetcode account

```
In [1]: print('abcd')
```

```
abcd
```

```
In [ ]:
```

## Programming basics

1. Memory and storage
2. Bit Byte
3. Data types: primitive
4. Basic Operators
5. Modulus and XOR operator
6. Compile, Interpreter

Memory - in terms of programming Register, Cache, RAM, Disk

Primary memory - RAM

- Random Access Memory

boxes: 1 2 3 4 5 6 get by pos or no. : O(1) ball of different colors: O(N)

**Bit**

```
Binary language: 0 and 1
0-9 -> 1000 -> 91

Bit: 0/1
Byte: 8 Bits
KB: 1024 Bytes = 2^10 bytes
MB: 1024 KB
GB:
TB:


Number systems:
Binary, Octal, Decimal, Hexadecimal
2(0,1)  8(0-7) 10(0-9)  16(0-9ABCDEF)

Bi: (101)2
Oct: (5)8
Dec:  (5)10
Hex:  (5)16

Bi: (1100)2
Oct: (14)8
Dec: 12
Hex: (c)
```

In [8]:

```
print(oct(12))
print(hex(12))
print(bin(12))
```

```
0o14
0xc
0b1100
```

**Data types**

```
- C++, JAVA: data type -> memory

What operations you can do on it.
- int vs string
 4 + 4 -> 8
 "4" + "4" -> "44"
 4 - 3 -> 1
 "4" - "3" -> X


 - int: 4, 8
 - float: 4, 8
 - char: 1
 - string: length
 - bool
 - bytes/bytearray



1 bit: 0,1 0->0 1->1
2 bit: 00(0), 01(1), 10(2), 11(4) -> 4
3 bit: 8
8 bits -> 1 byte -> 0-255 -> 256
       -> -128 to +127
2 int -> -32768 to +32767
4 bytes ->

32000 + 32000
```

In [ ]:

```

```

**Variables**

```
// C++, JAVA
int a = 10;
a = 20; // correct
a = 4.5; // ? yes
a = "abcd"; // ? no
a = 'a'; // ? yes

// Python
a = 10
a = "abcd"
```

```
LHS <- RHS

20 = a; // wrong

/*
C++ thing
*/
if a = 20 {
}

if 20 = a {
}
```

**Constant**

```
const int a = 10;
a = 20; // error
```

In [ ]:

**Operators**

```
Arithmetic
+, -, *, /, %, (// , ** python)

Shift operators: bits
>>, <<

Assignment
=, +=, -=, *=

Relational
==, <, >, >=, <=, !=  (=== JS)

Logical
&&, ||, !  (and, or, not python)

Bitwise
| , &, ~, ^


?: -> if else
```

In [13]:

```python
a = 4 # 0b 0000 0100
print(a >> 1) #0b 0000 0010
print(a << 1) #0b 0000 1000
print(a << 2) #0b 0001 0000
```

```
2
8
16
```

In [ ]:

**modulus**

```
1 % 2 -> 1
2 % 1 -> 0
5 % 2 -> 1
```

x % y -> [0 to (y - 1)]

Even or odd ?  
do % 2 -> if 0-> even 1-> odd

extract last digit of a number

- 123
- 123%10 -> 3
- 123/10 -> 12
- 12 %10 -> 2
- 12/10 -> 1
- 1%10 -> 1
- 1/10 -> 0

0*10 + 3 -> 3*  
*(3 * 10) + 2 -> 32*  
*(32*10) + 1 -> 321

In [ ]:

**xor**

```
Truth Table
0  0 -> 0
0  1 -> 1
1  0 -> 1
1  1 -> 0
```

In [21]:

```
a = 5 # 101
b = 4 # 100
print(a^b)

print()
a = 1 ^ 3
print(a)
print(a^1)
print(a^3)
```

```
1

2
3
1
```

```
int main() {
    int a = 1 ^3;
    cout << a << endl;
    cout << a ^ 1;
}
```

In [ ]:

In [1]:

```
% pip install pandoc
```

```
UsageError: Line magic function `%` not found.
```

## Some math

1. Sum of first n numbers
2. Mean, median, mode
3. Permutation Combinations
4. Modulo arithmetic
   (A + B) mod C = (A mod C + B mod C) mod C
   (A * B) mod C = (A mod C * B mod C) mod C
   A^B mod C = ( (A mod C)^B ) mod C

In [ ]:

### Common Terms

1. Pallindrome
2. Factorial
3. Fibonacci Series
4. Sorting

In [ ]: