Basic\_of\_c\_2

Bitwise not operator(~) - it is also a unary operator (i.e those operator which need only operant to perform their work)

As in the 32 or 64 bits -

Integer value will take 4 bytes - i.e 32 bits

Thus

Binary of 5 is 00000000000000000000000000000101

i.e (<0\*29>101) which we normally represented by 101

And by (~) bitwise not operator all the zeros are converted to 1 and all the 1 are converted to 0

I.e (~5 = 11111111111111111111111111111010)

So tabhi only (101) galat hai kyu ki us me bahot sare zero hote h jin ko 1 banana hai bitwise not operator me

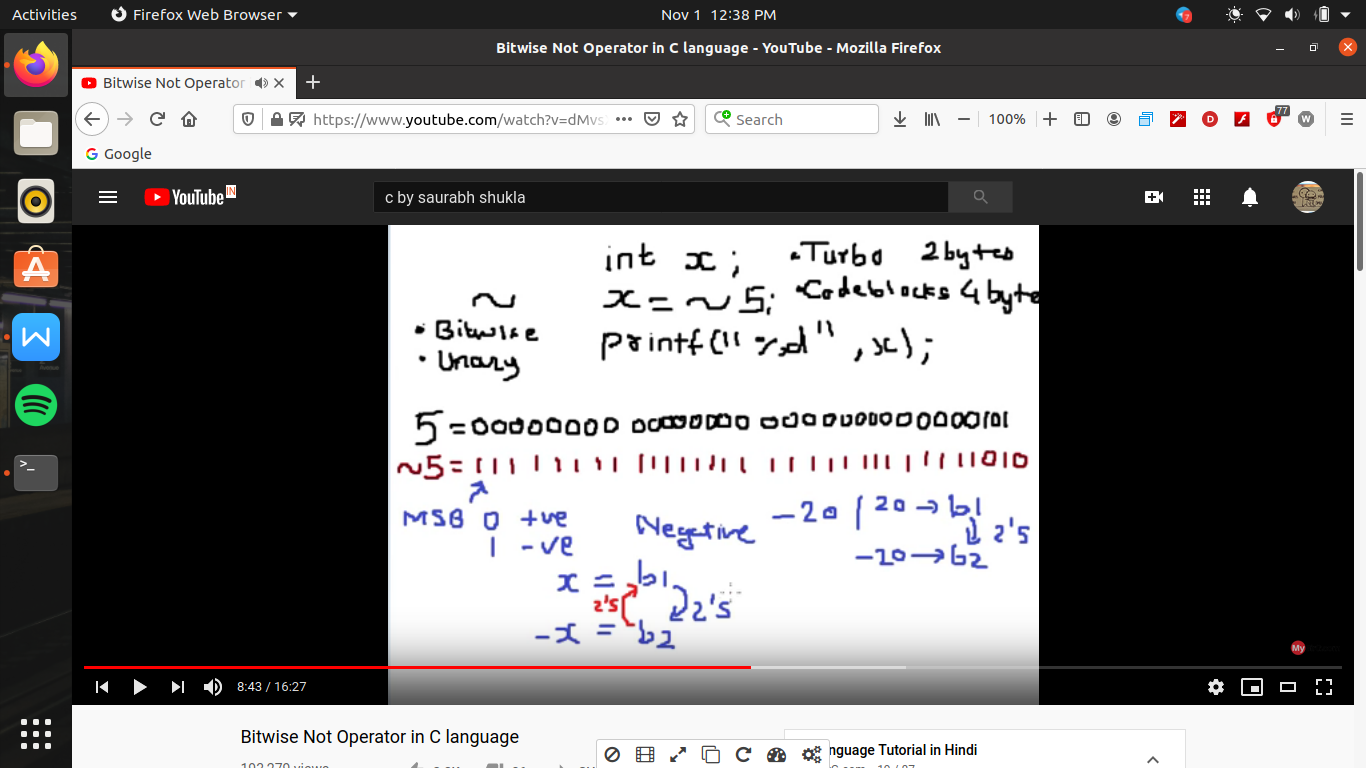
Here jo

Pahali wali bit hoti hai use kahate h most significant bit

And if first bit is 0 then it means no. Is positive(+)

And if 1 -> (-ve)

So bitwise not operator ke use karane se +ve value negative ho gayi



Type of bitwise operator ->

1. Bitwise AND &
2. Bitwise OR |
3. Bitwise XOR ^ // i.e excluseive or
4. Bitwise NOT ~
5. Right shift >>
6. Left shift <<

As bitwise operator are work in a binary no’s

So those operants are 0 and 1

&operator ->

And operations is works as same as and gate

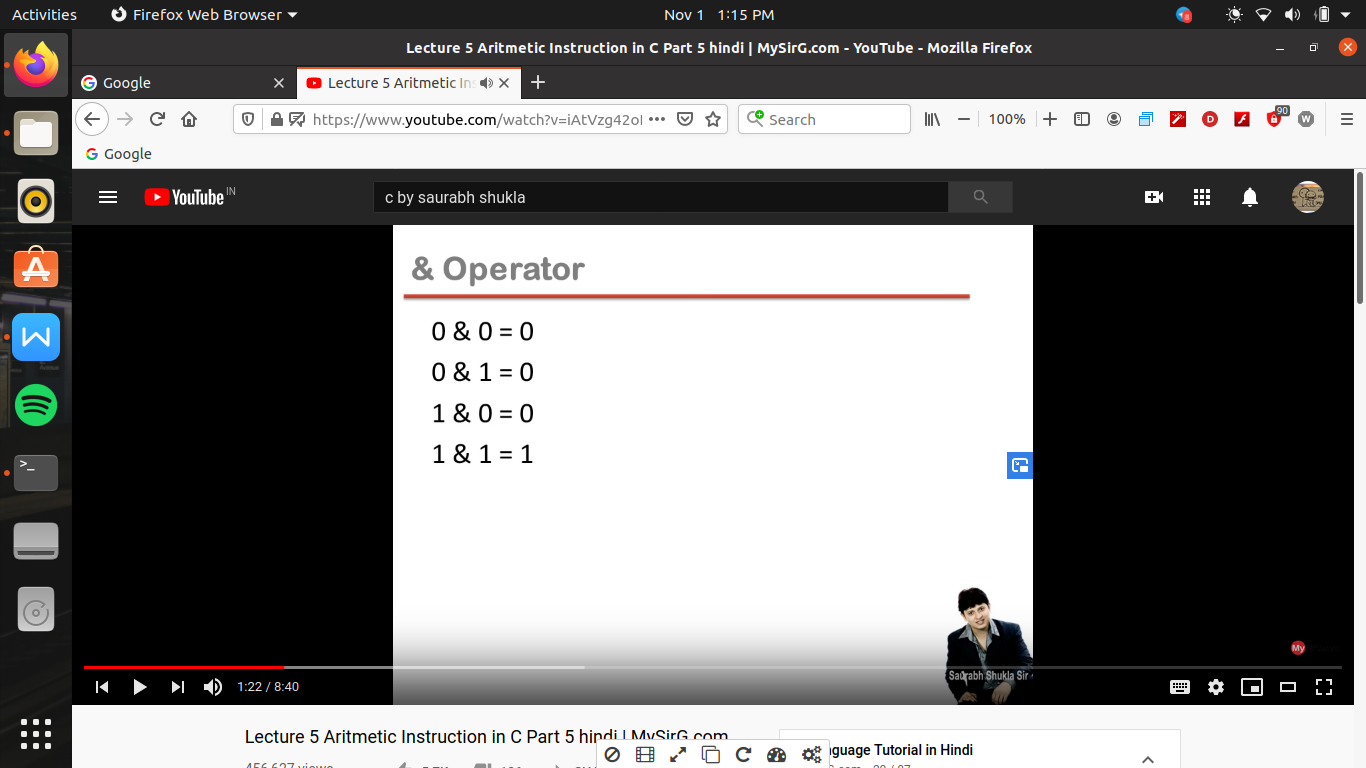
1. e agar dono me se koe 0 hoga toh us ka result bhi 0 ayega)

0&0=0

0&1= 0

1&0 =0

1&1 =1



Ex-

