



# Pointers - Class 1 [Join Here]

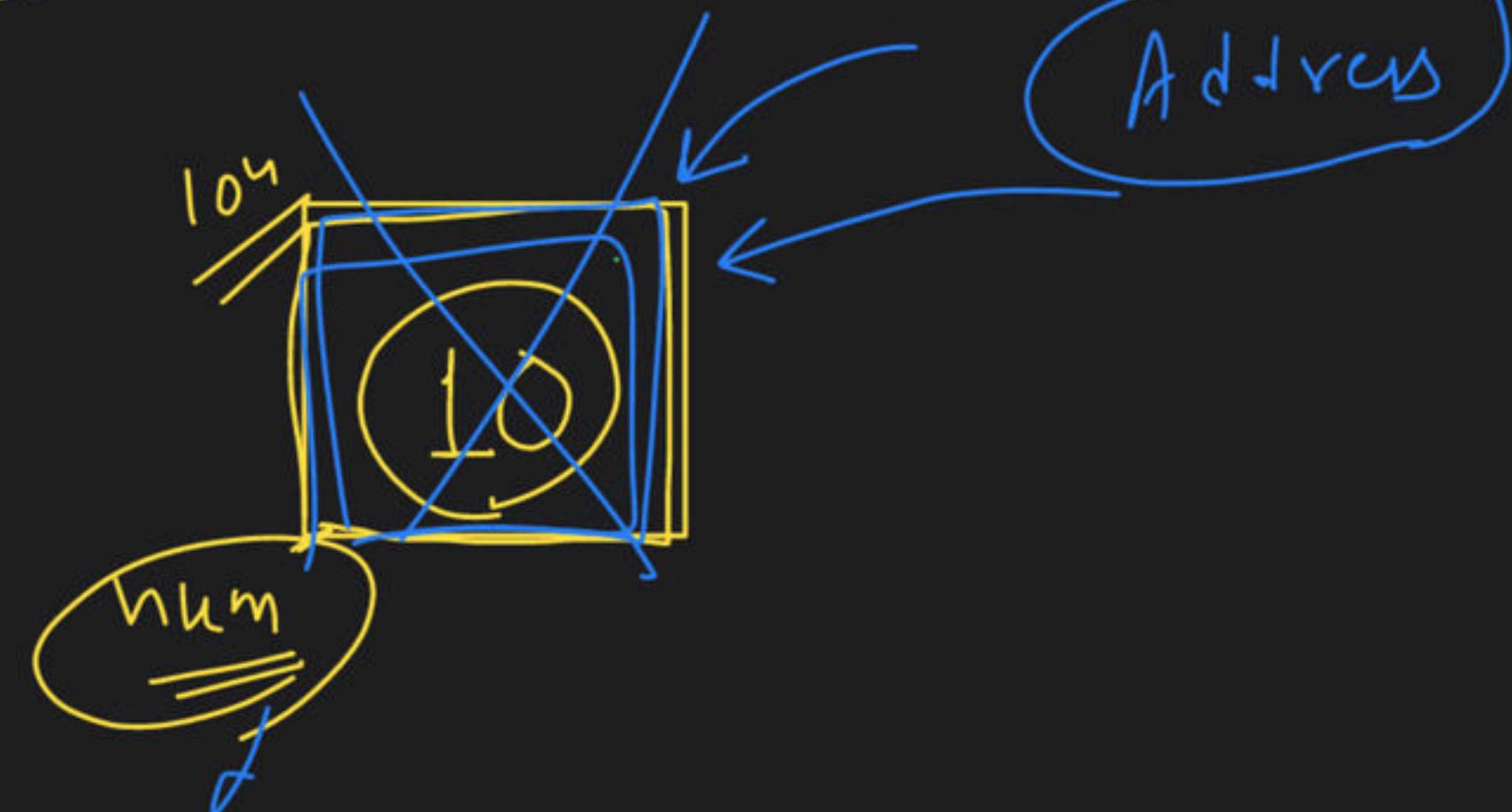
Special class

# Pointers:-

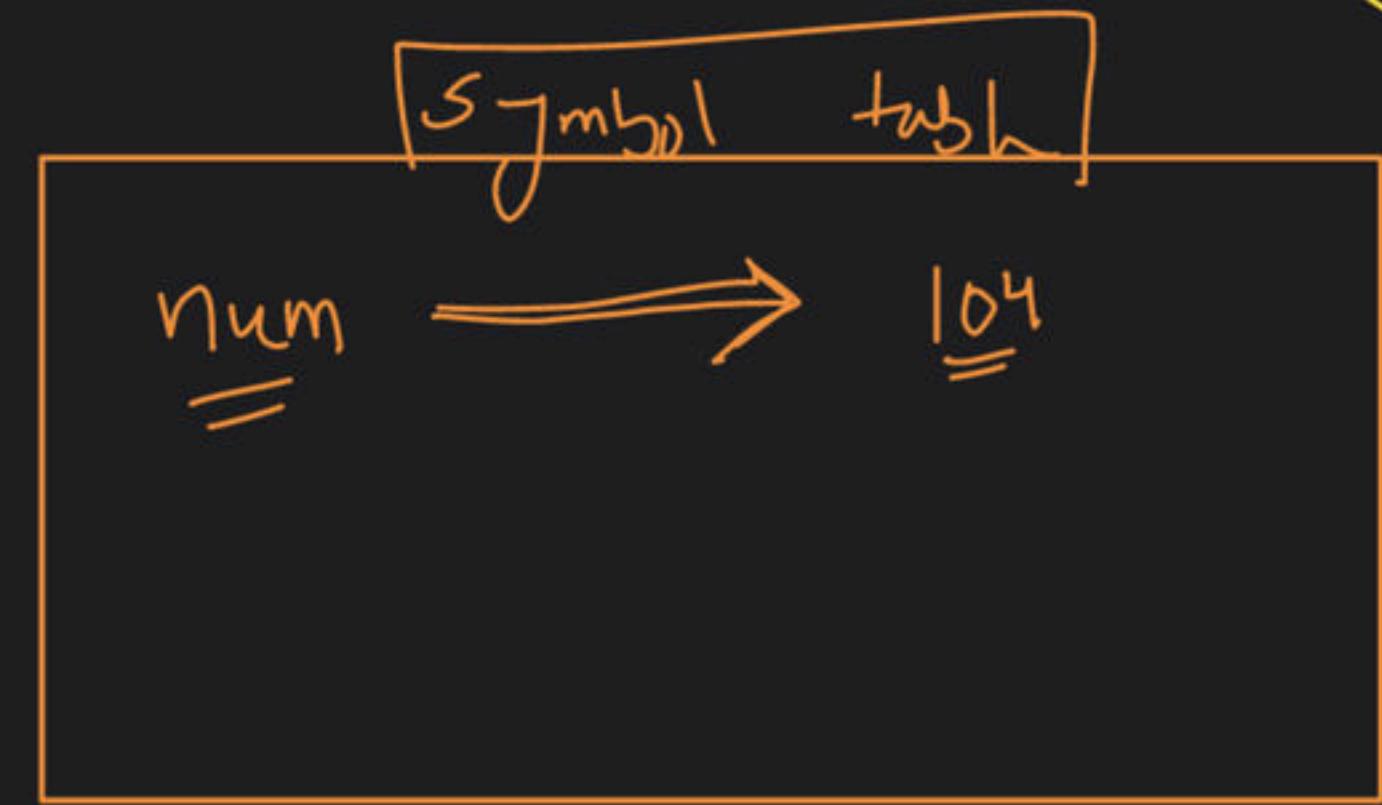
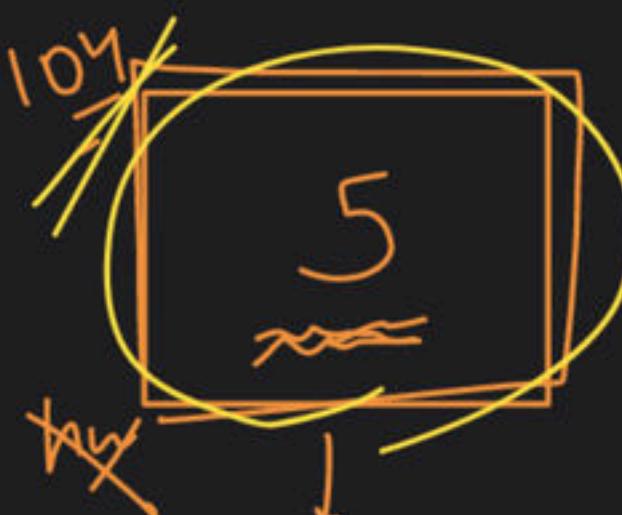
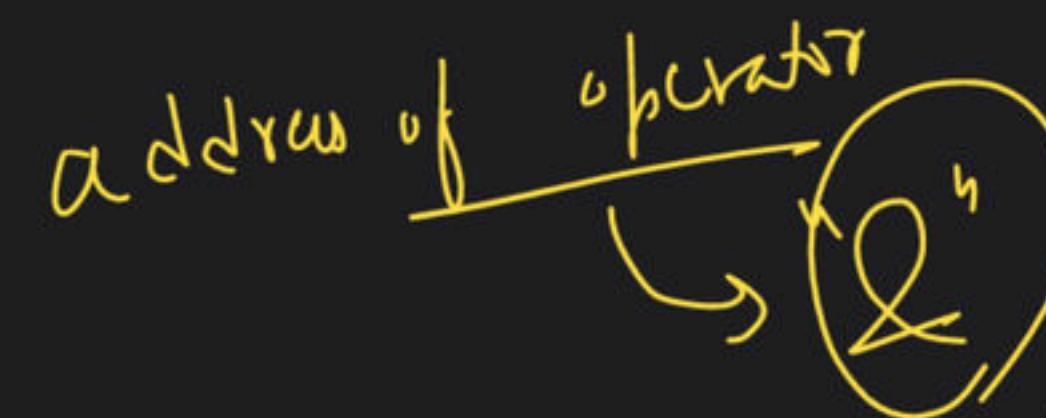
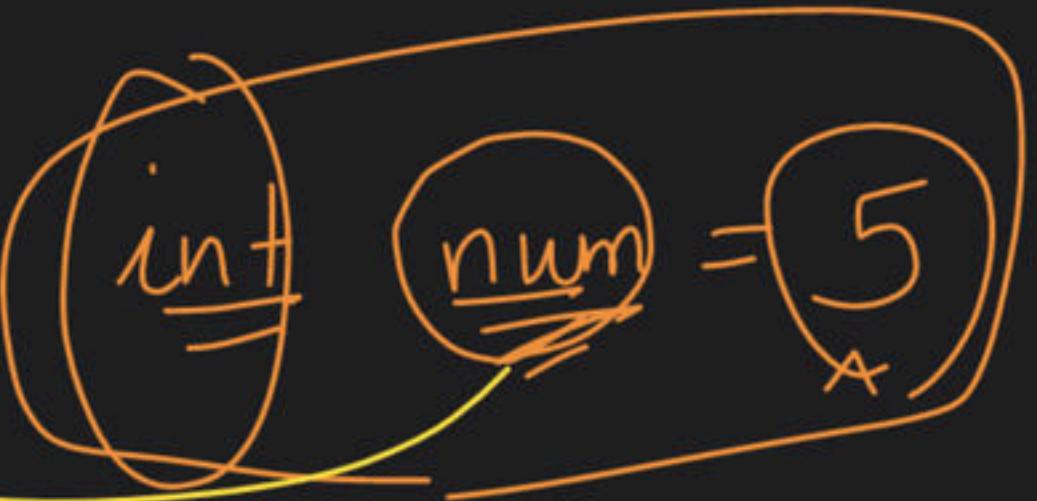
variable  $\longleftrightarrow$  store address of

other variable

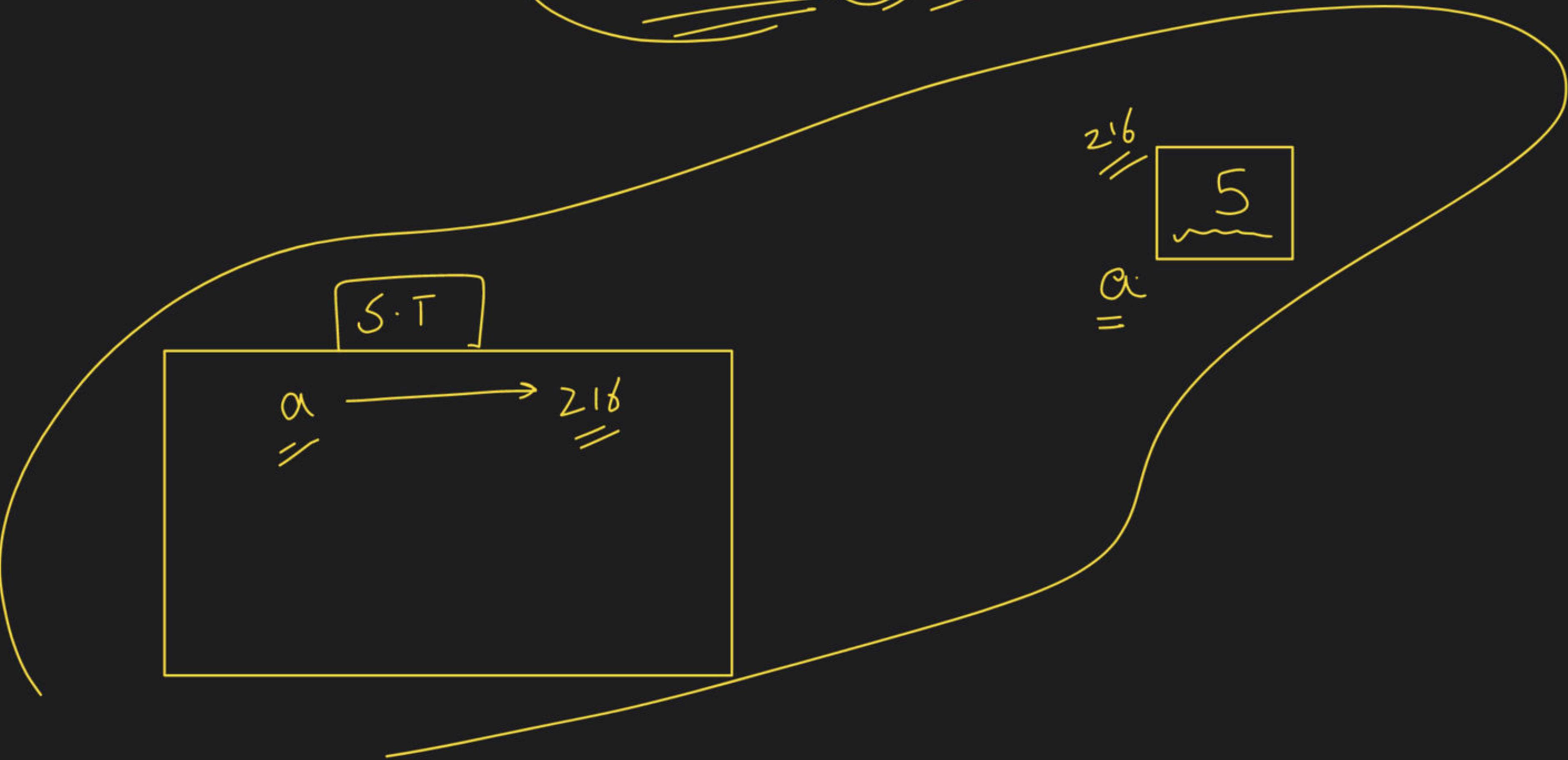
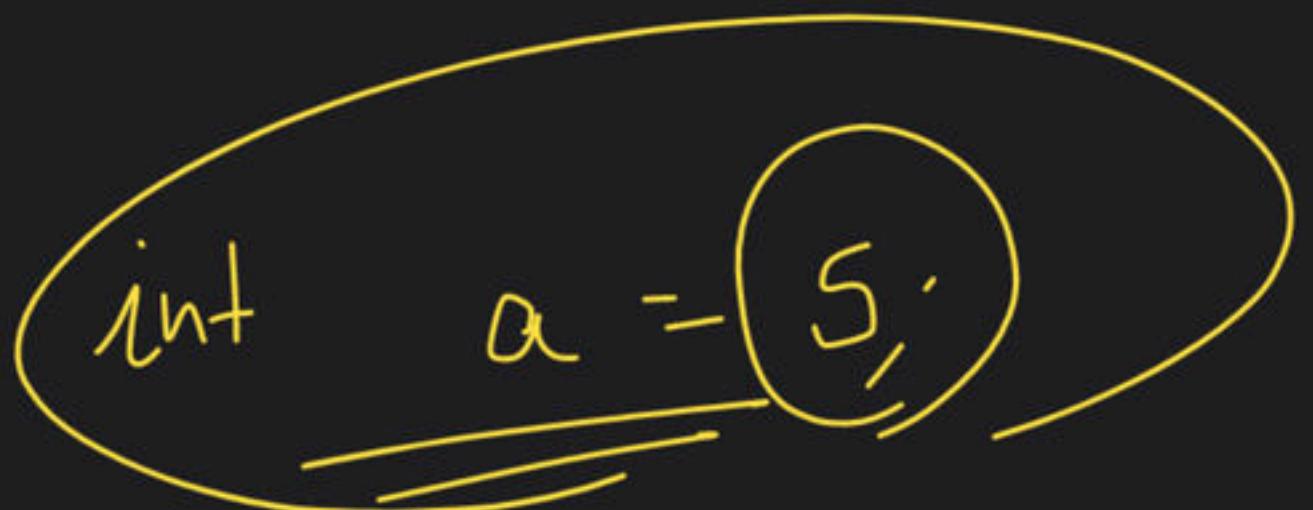
int num = 10;



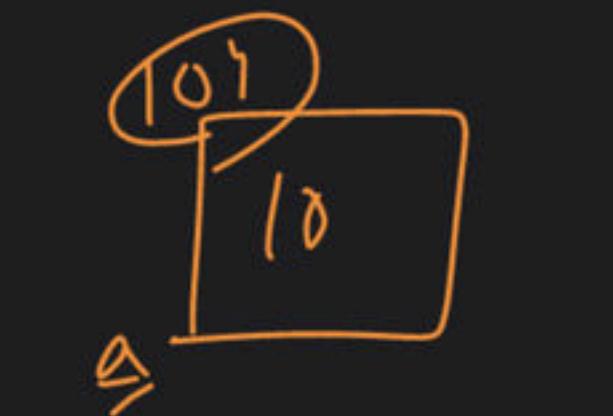
$p^{int^{\perp}}$  addres



104 hexadec



```
int a = 10;  
cout << a  
cout << 10
```



? → 10  
? → 101 =

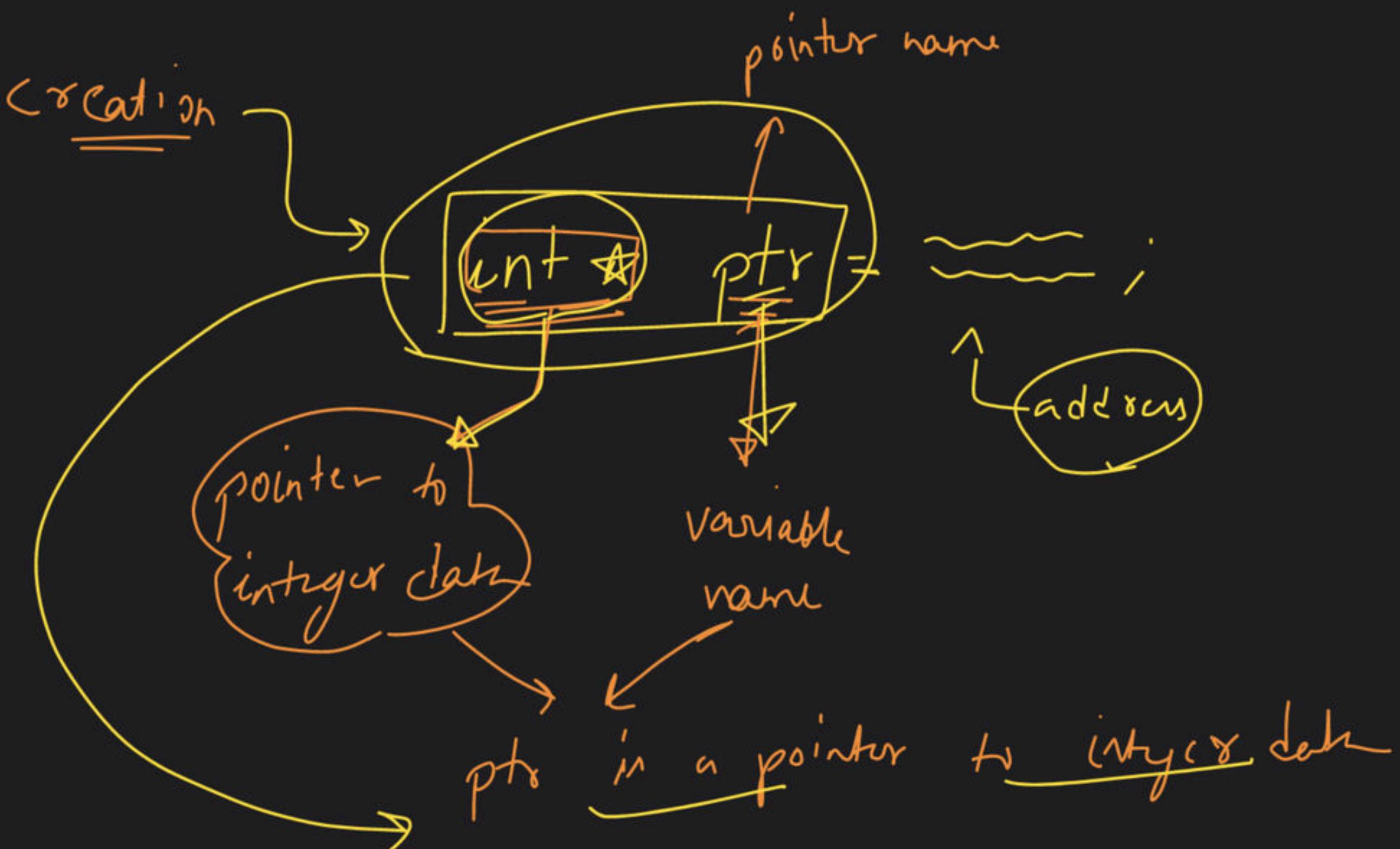
Pointer

char \*  
↓  
pointer to  
char data

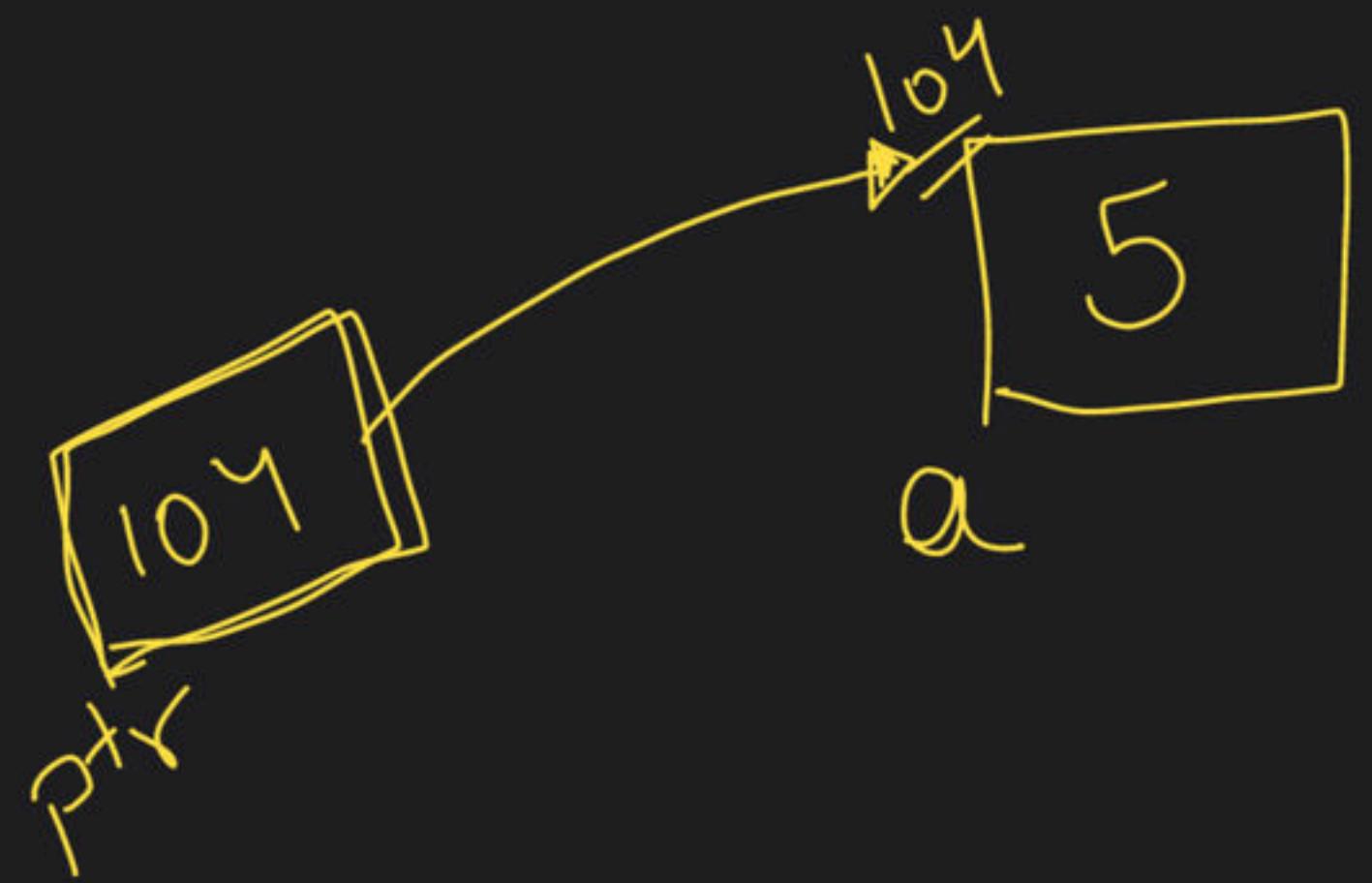
long \*  
↓  
shool \*

bool \*

creation



```
int a = 5;
```



```
int * ptr = &a;
```

lvalue →  $\text{int } \star \text{ptr} = \&a;$

lvalue → value stored at address stored in ptr  
 ↳ dereference  $\rightarrow \star \rightarrow \text{cout } \star \text{ptr}$

2016

ptr

1008

1008  
10  
a

10

why?

~~Why~~  
Dynamic Memory

int a = 5

~~int~~

int \* ptr = &a;

2016

ptr  
1008

int \* ptr  
=

int \*  
ptr

int \*  
ptr

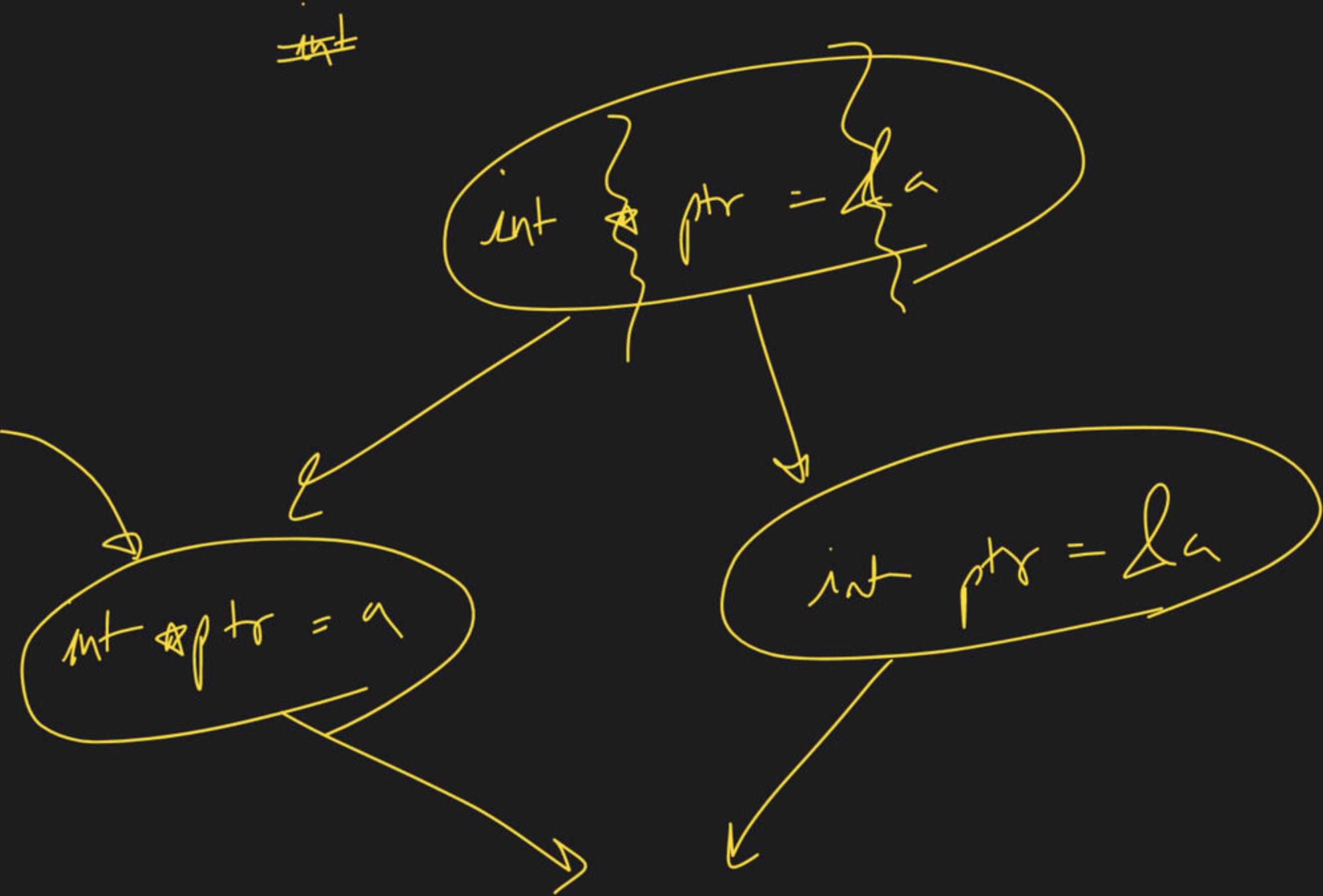
a

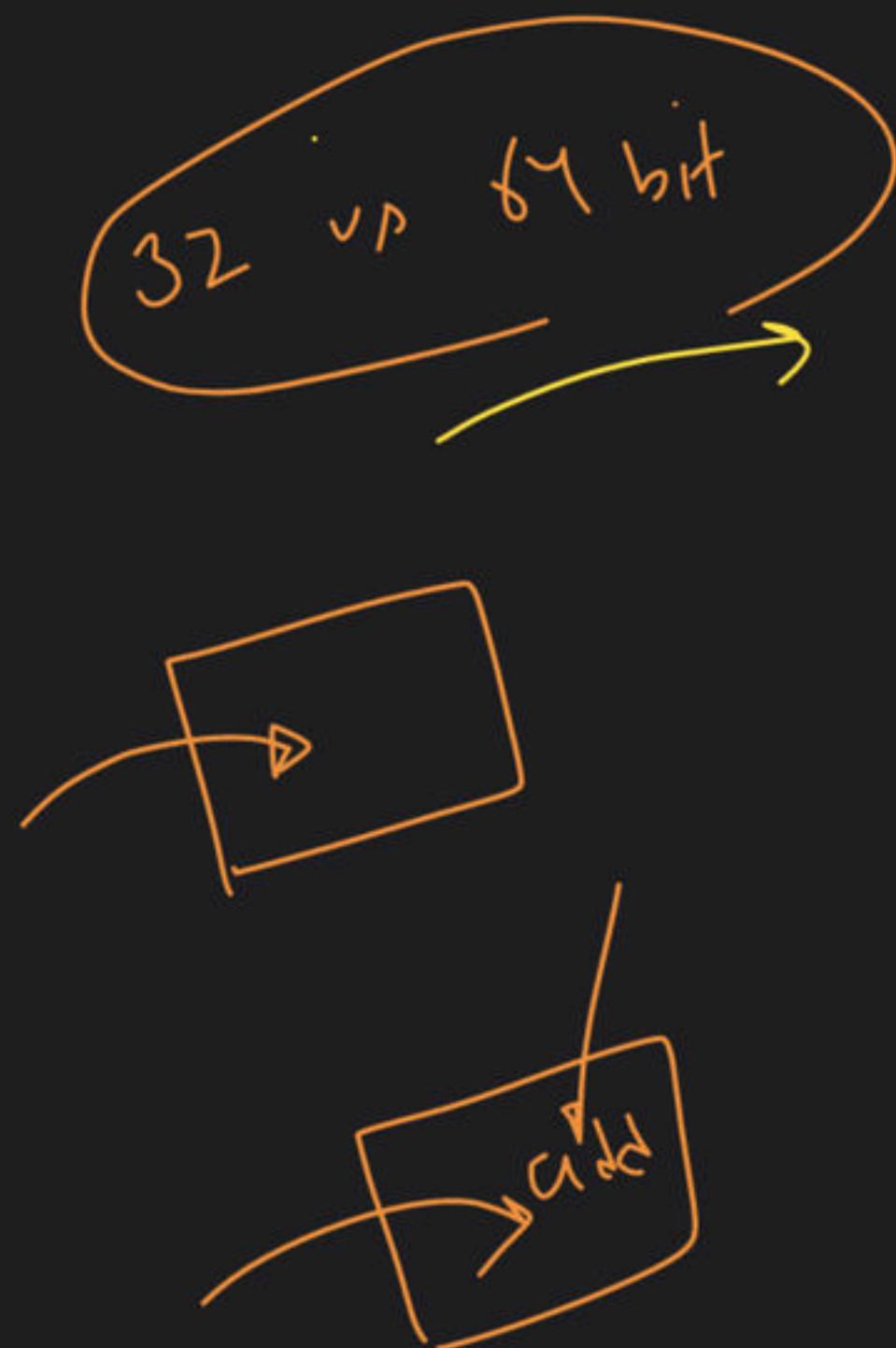
5

a → 5

&a → 1008

\*ptr → (I) address of J2V (II) value rnv  
5





```
int a = 5;
int *ptr = &a;
```

```
char ch = 'K';
char *cptr = &ch;
```

```
long lachi = 10;
long *lptr = &lachi;
```

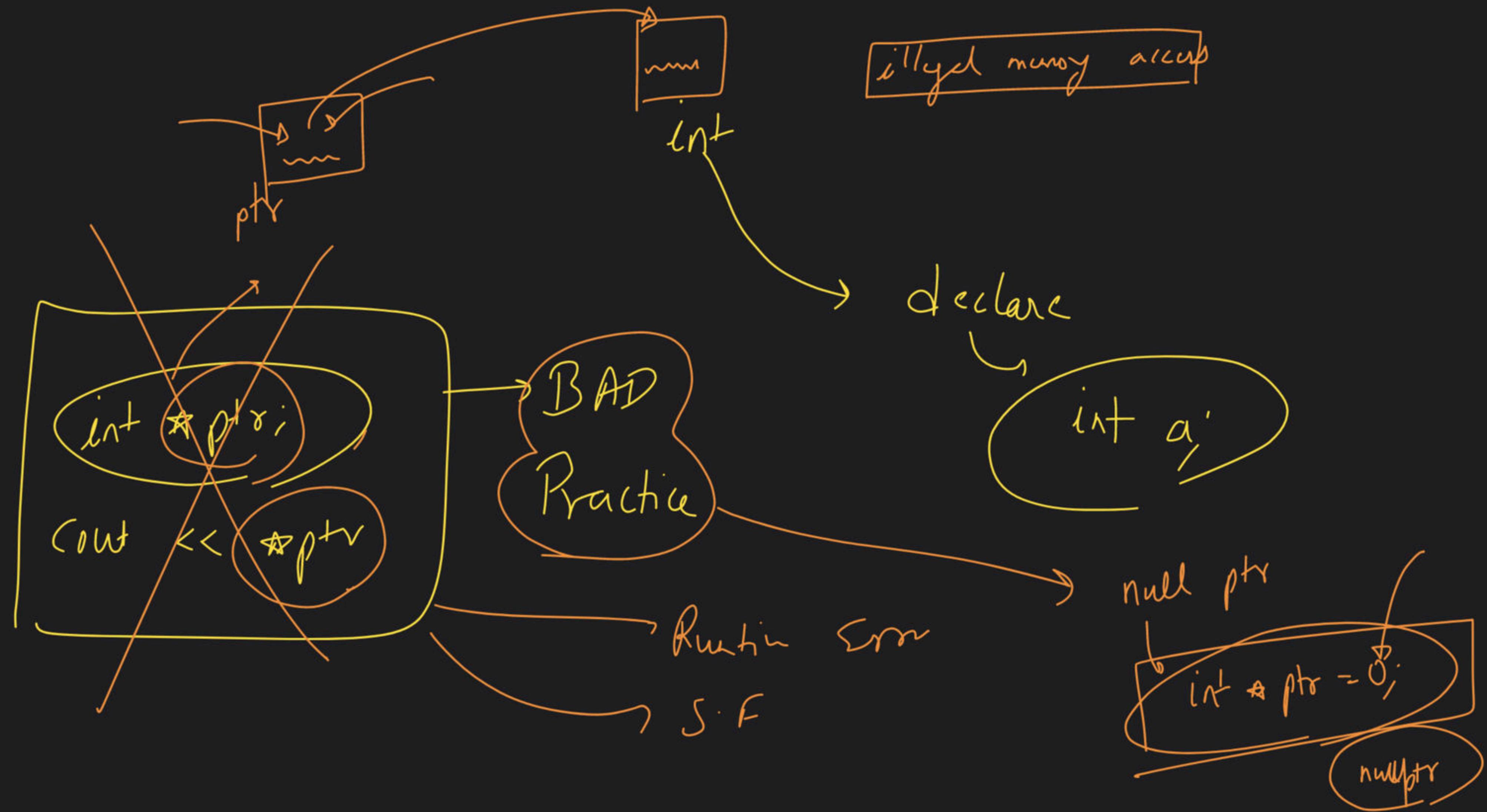
why

32 vs 64 bit  $\rightarrow$  what

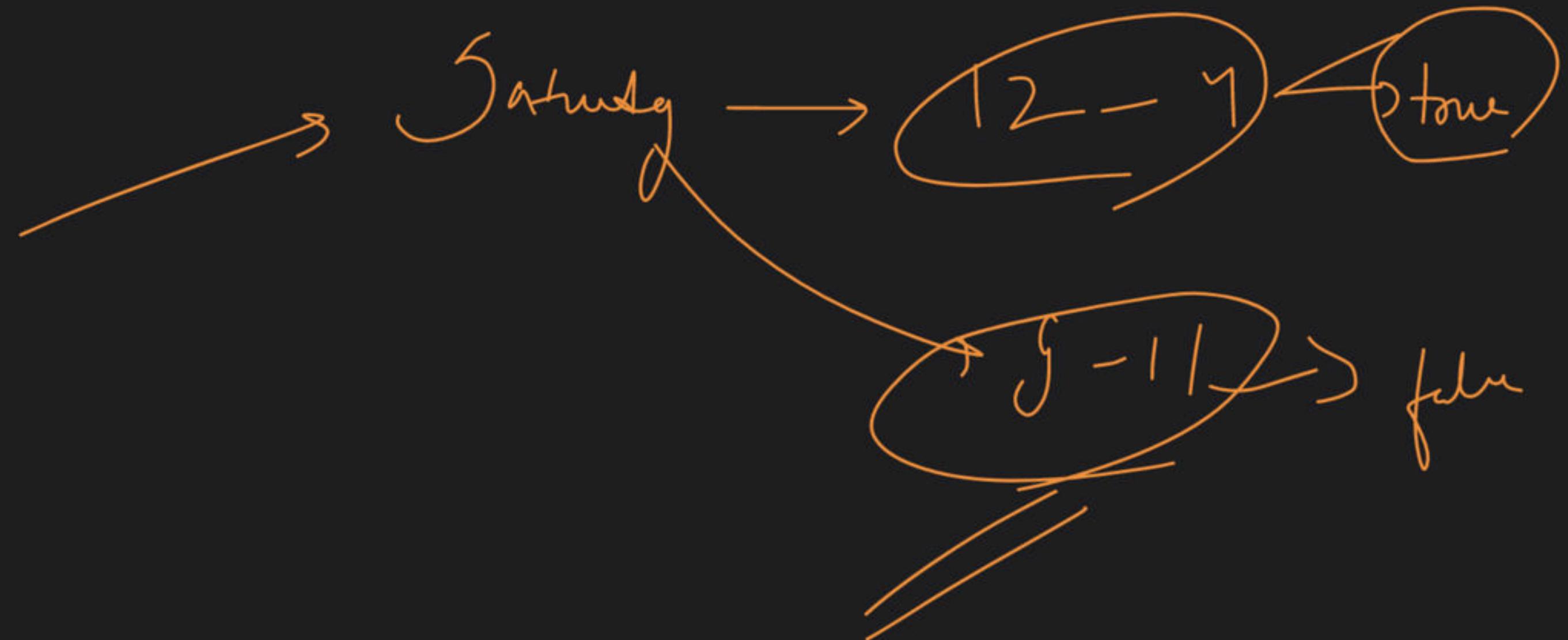
why  $\xrightarrow{\text{ptr}} \delta/\gamma?$

why

dynamic memory allocation



Expr<sup>c</sup>  
union



int \*ph

$$\text{int } *ph \rightarrow \text{int } *ph = 0$$

①

```
int a = 100; =  
int *ptr = &a =
```

a = a + 1;

ptr = ptr + 1;

a → ?

ptr → ?

1 byte

bool → 1 bit  
1 byte only

100 8

ptr

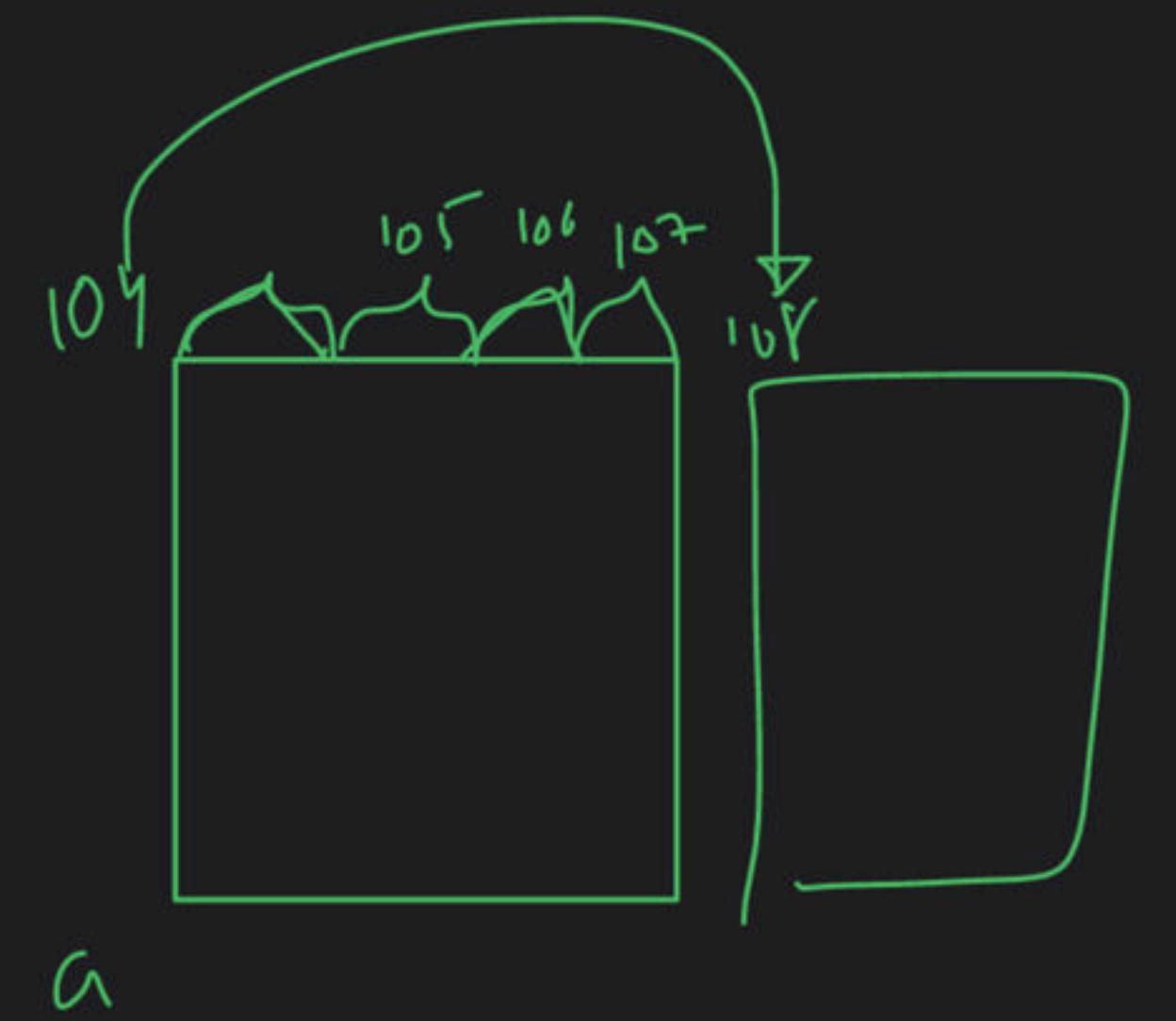
104H

a = 4 bytes



$$\begin{aligned}104 + 1 \\= 108\end{aligned}$$

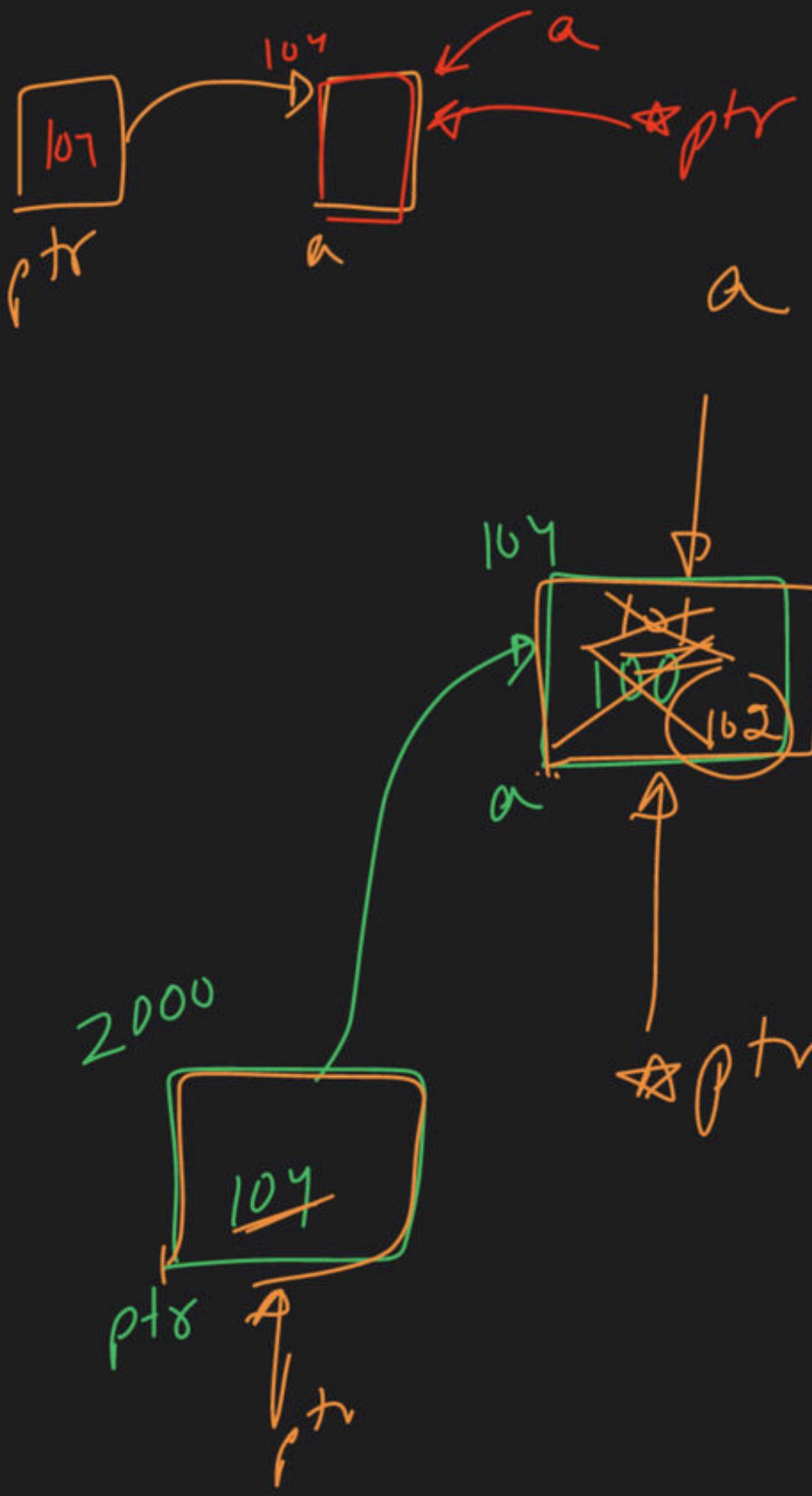
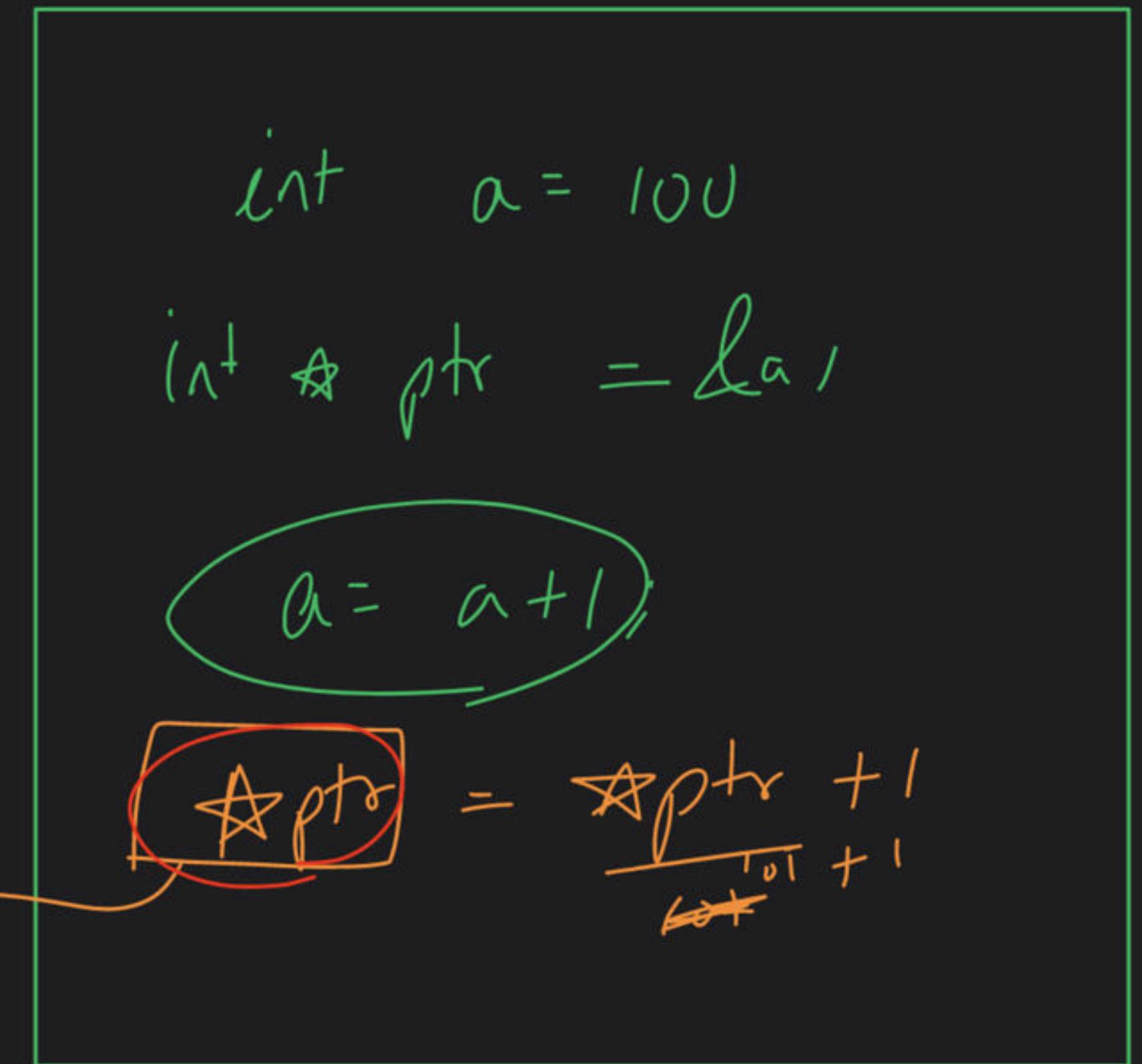
15 sec



a



Value present at  
address stored in  
 $\star \text{ptr}$

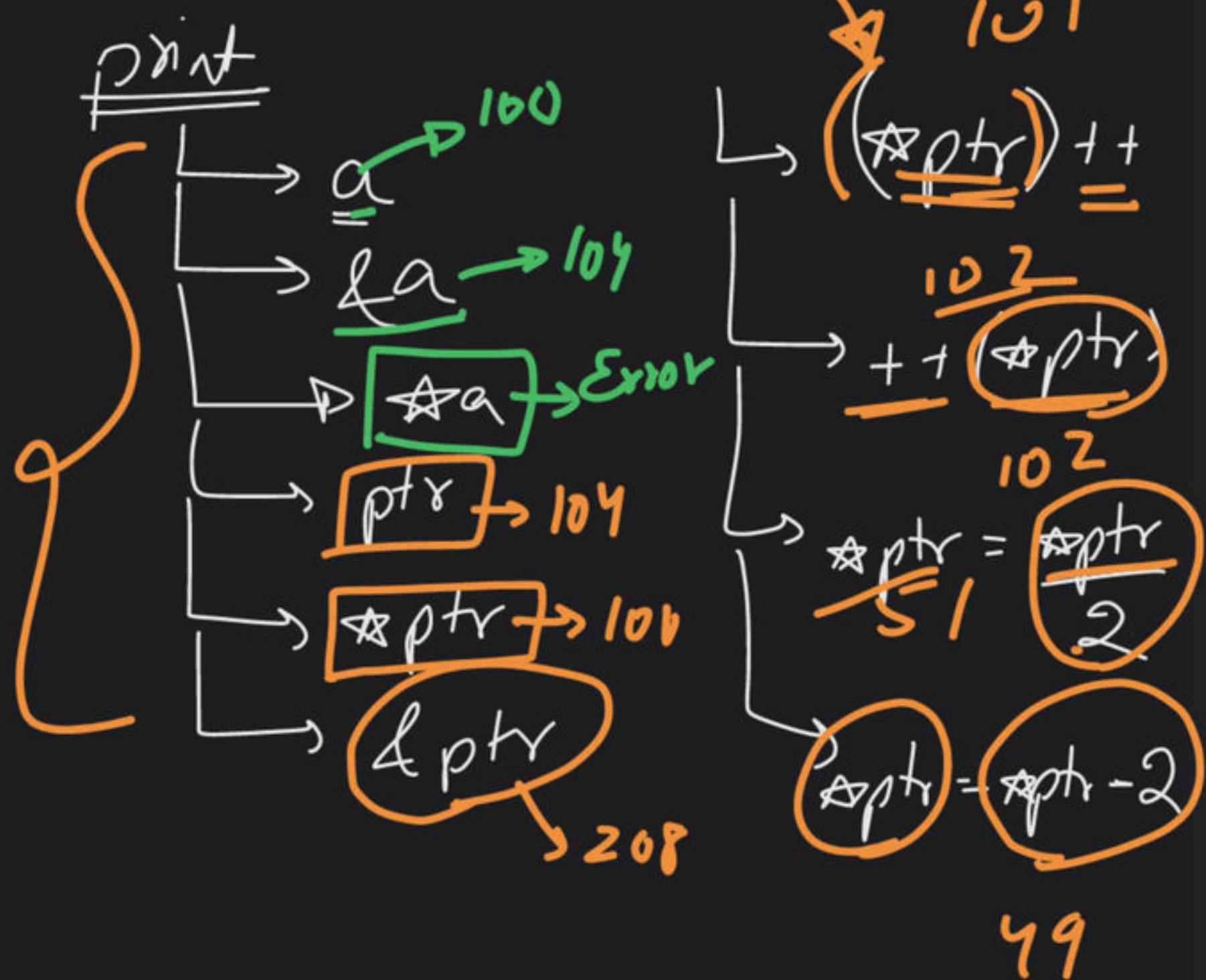
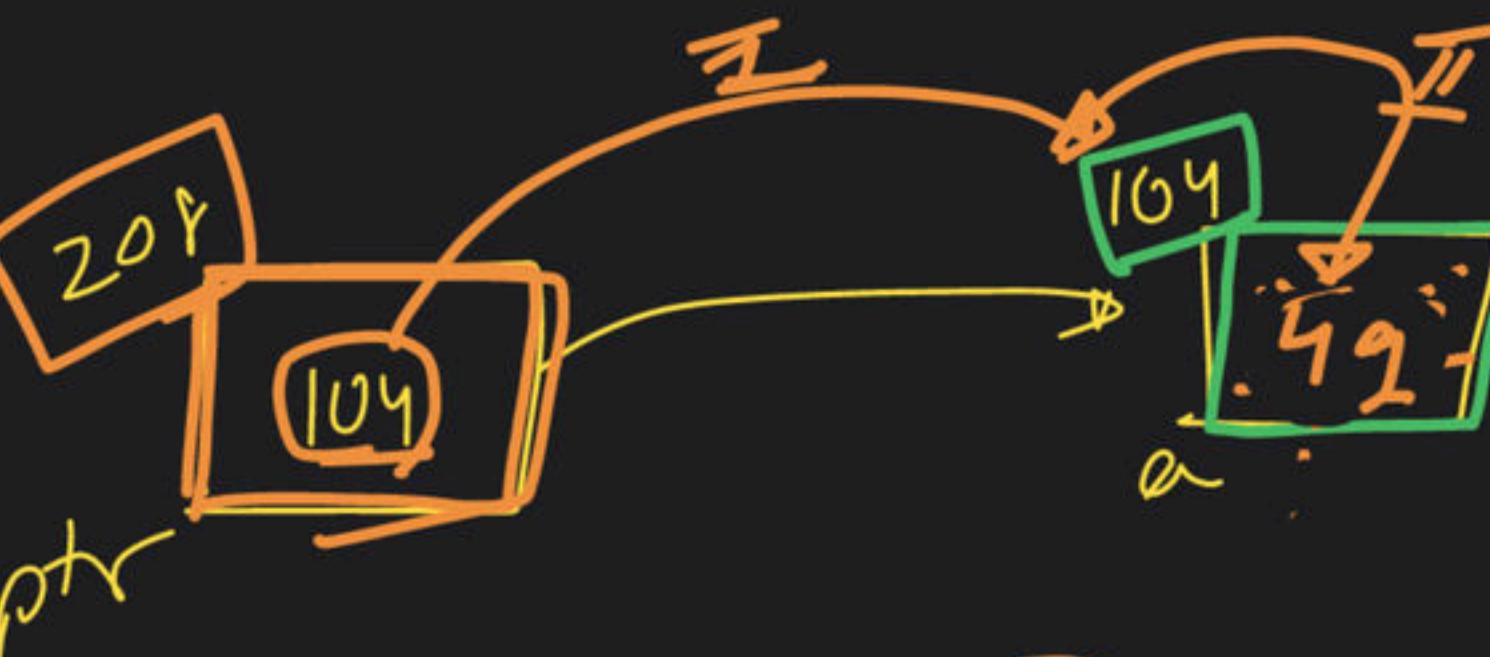


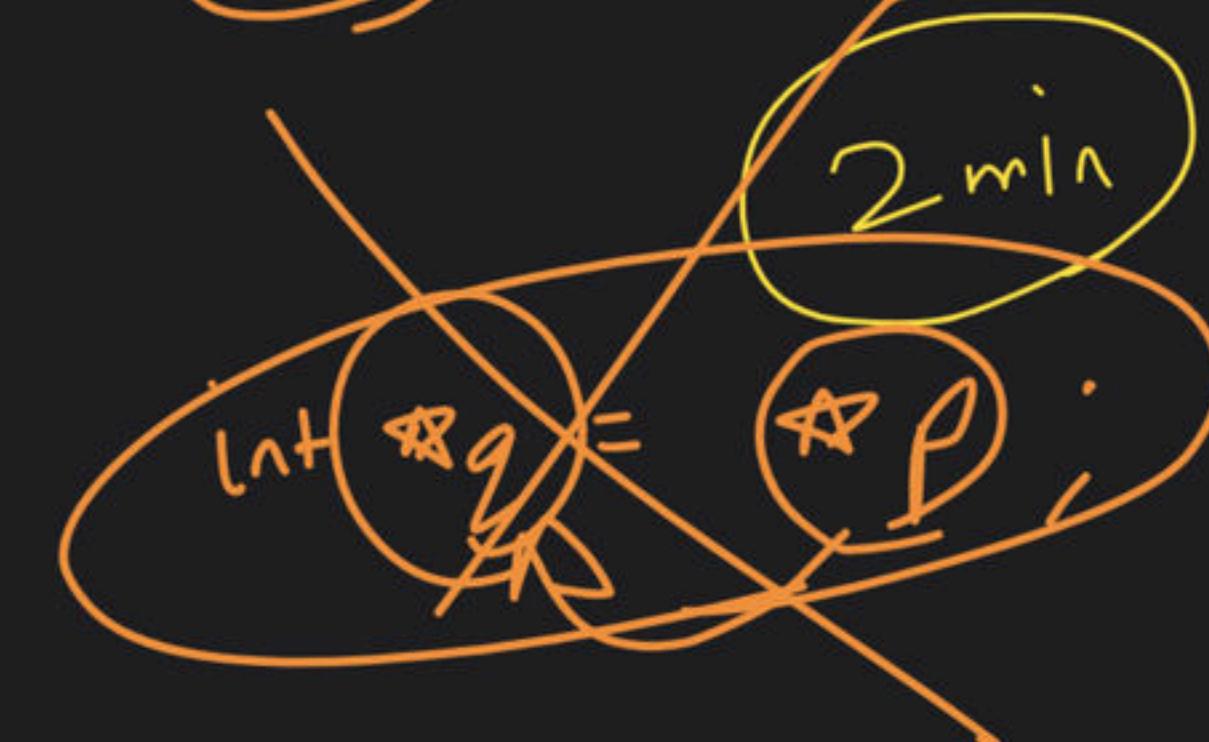
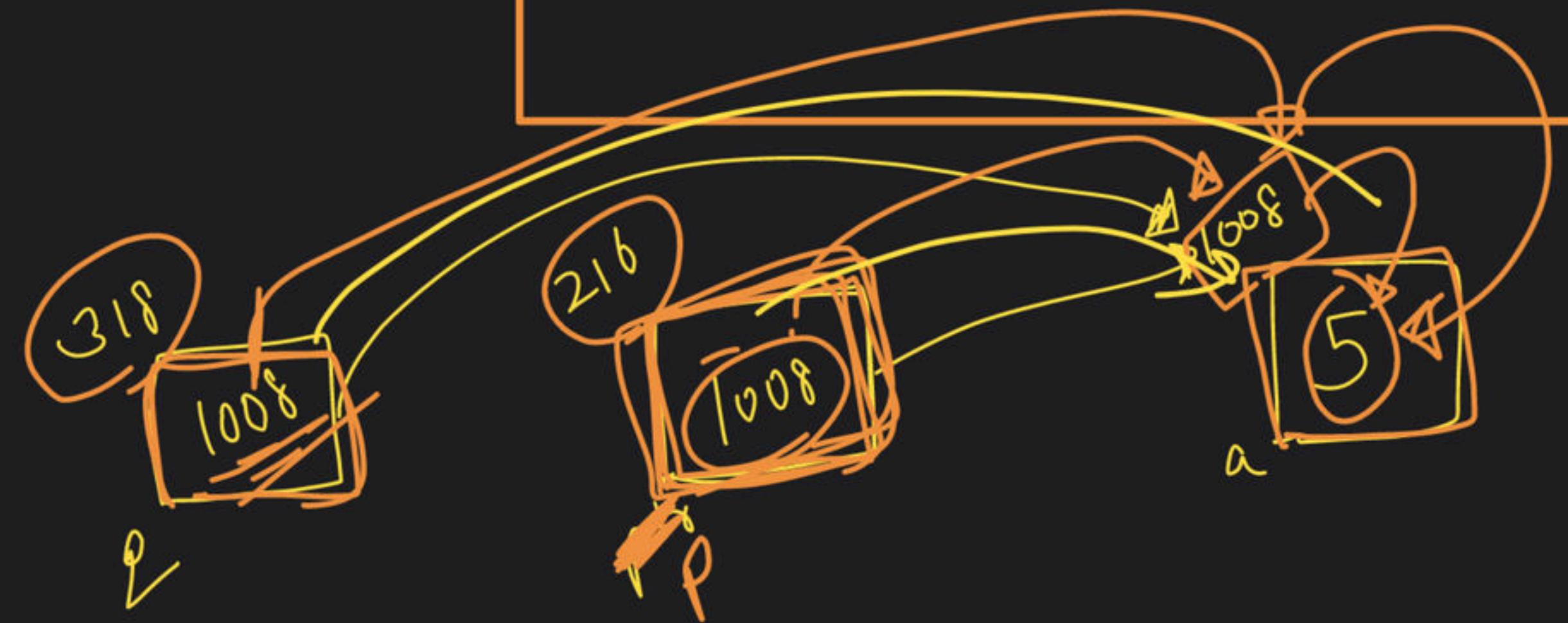
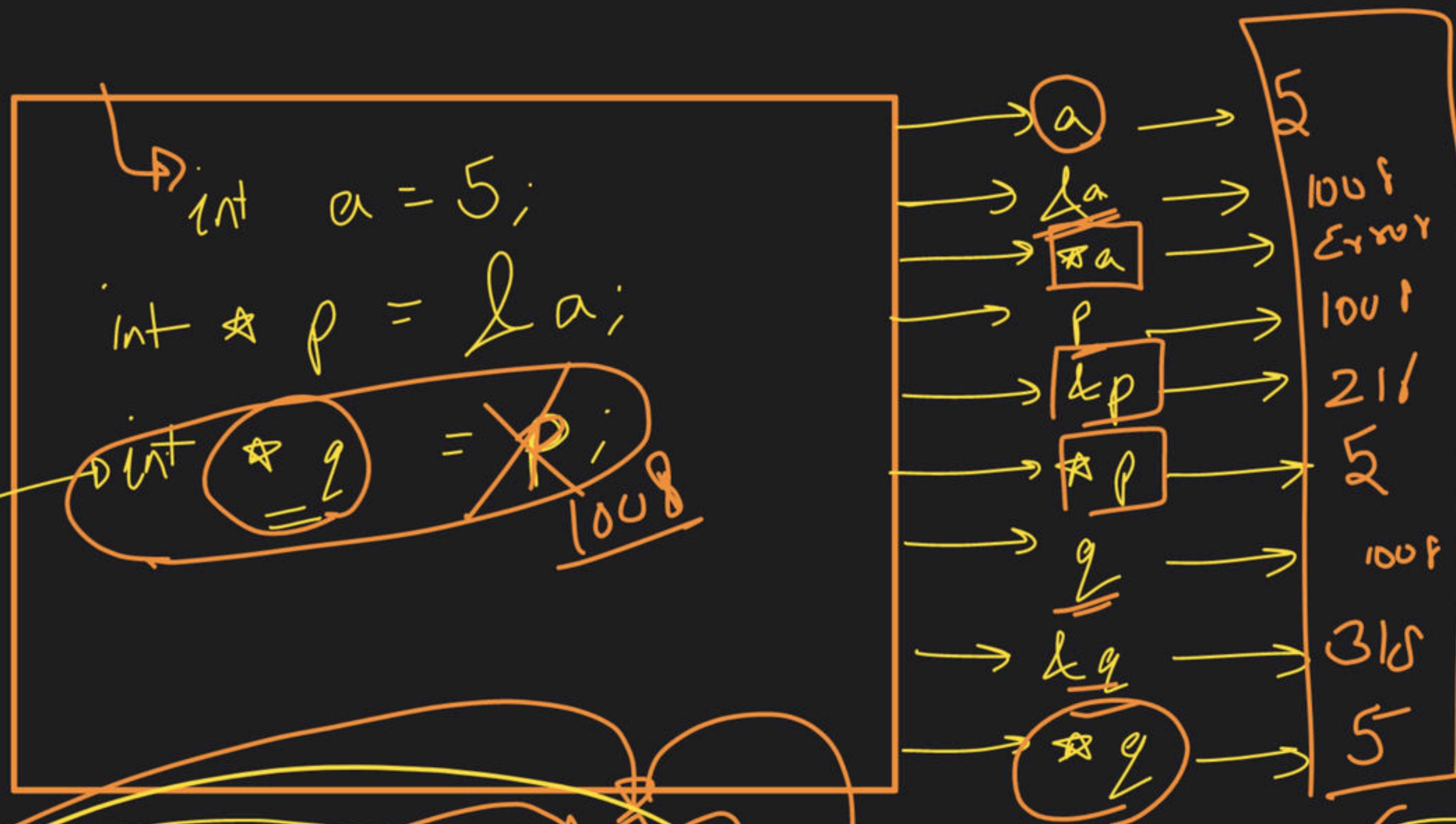
```

int a = 100;
int *ptr = &a;

```

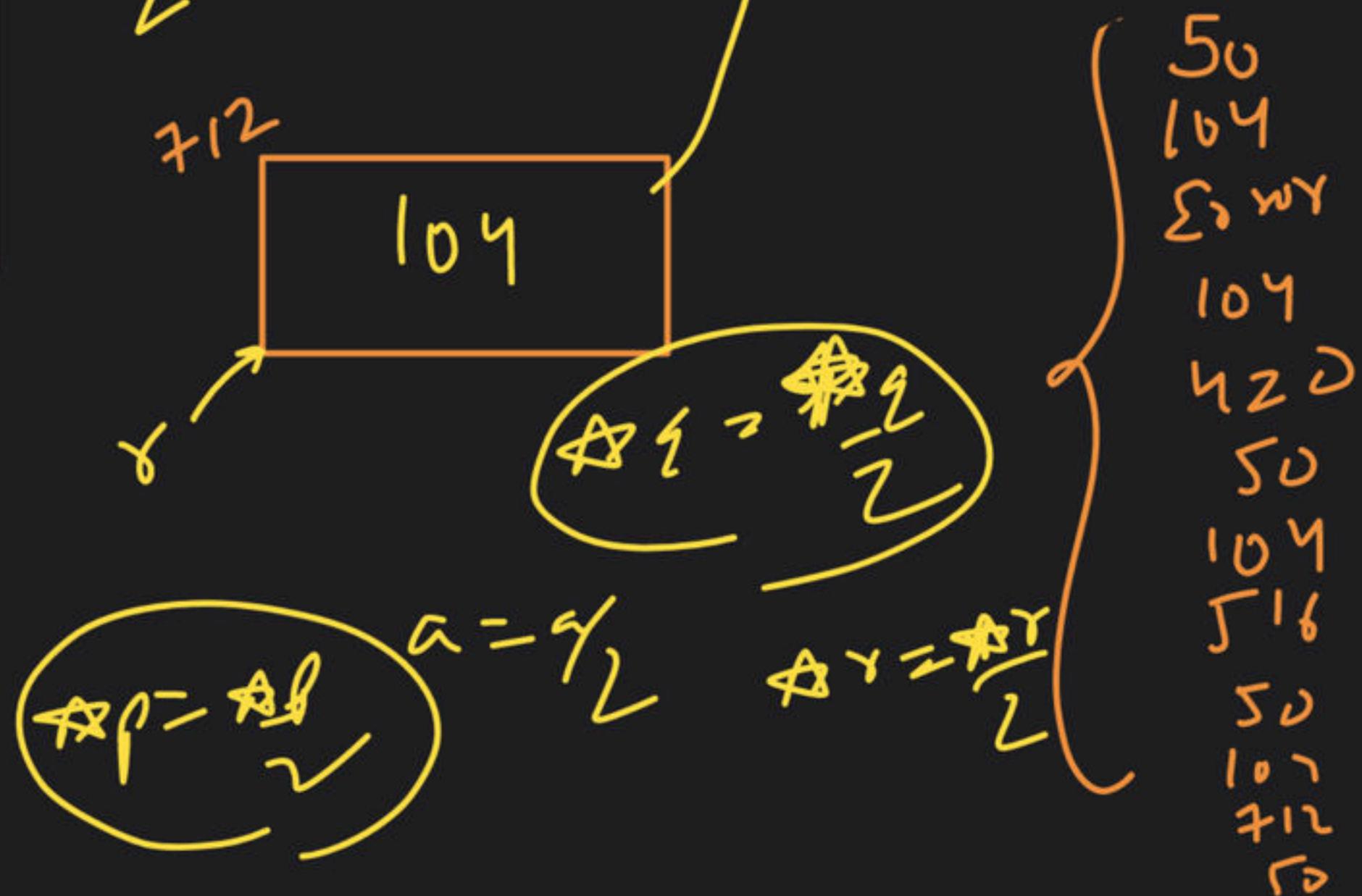
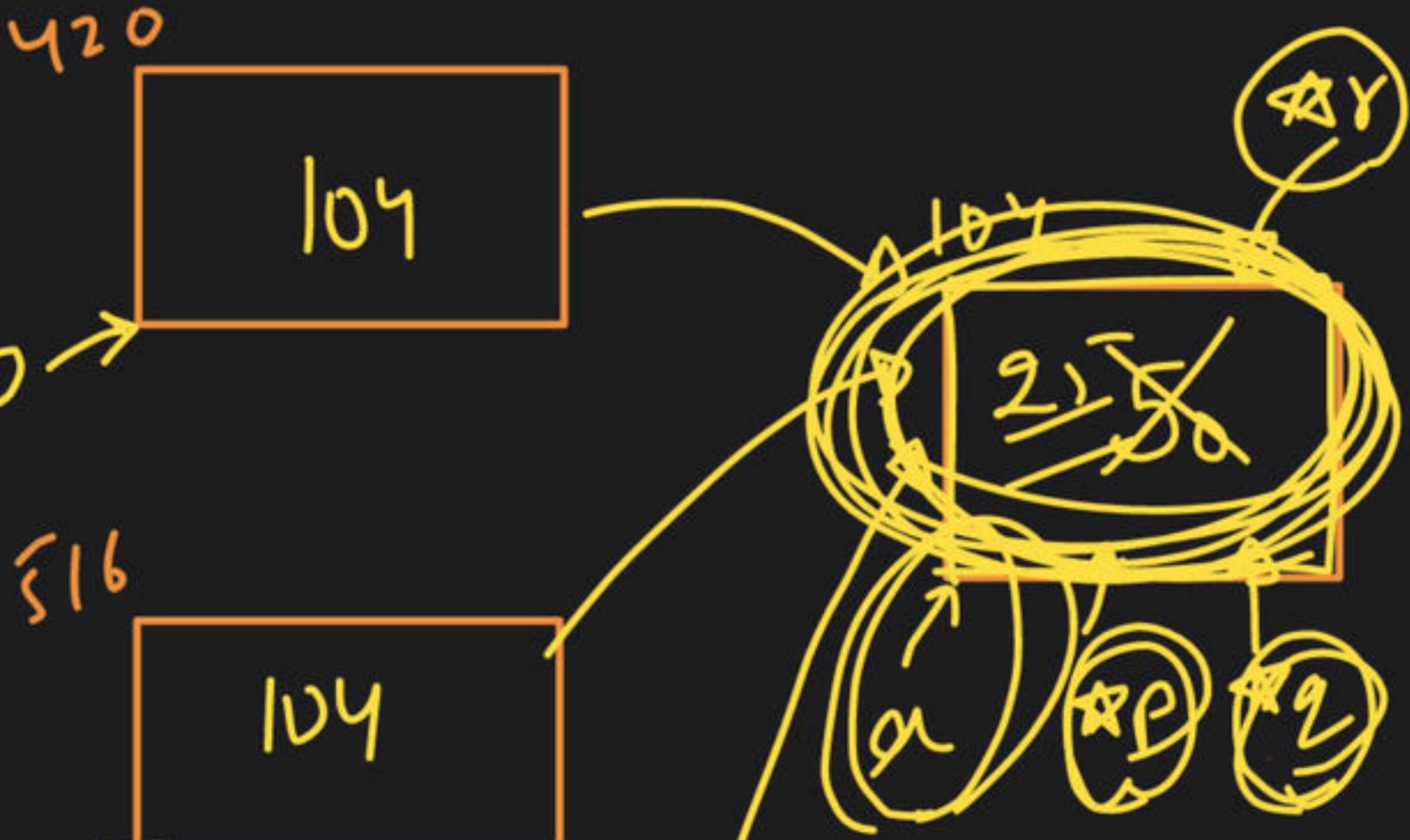
1 minute → ?





POINTING COPY

int  $a = 50;$   
 int  $\star p = \&a;$   
 int  $\star q = p;$   
 int  $\star r = q;$

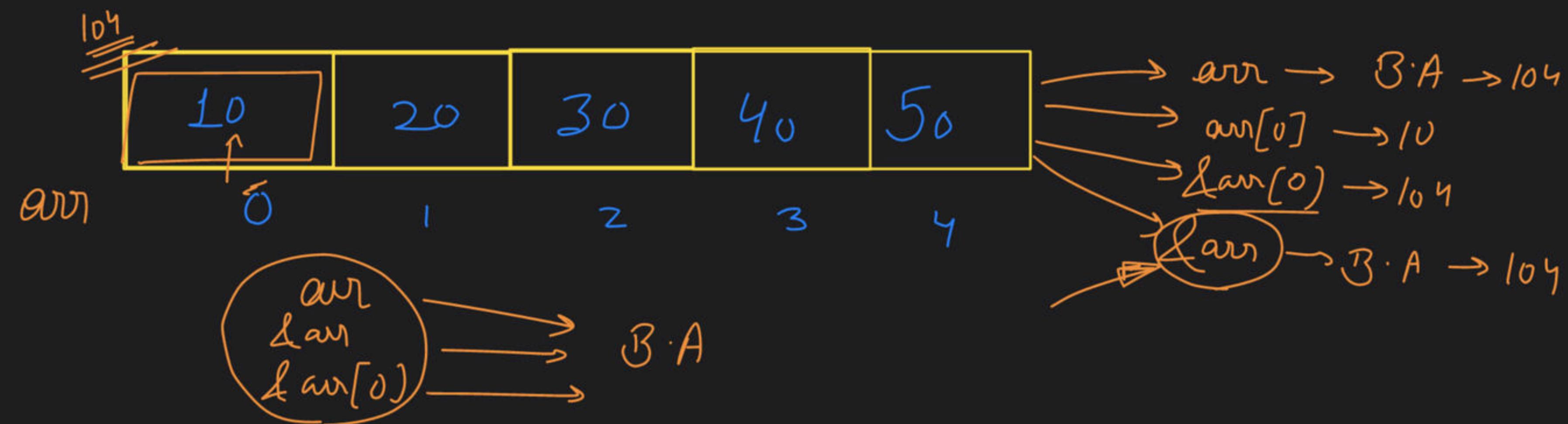


# Pointer with Array

```
int arr[5] = { 10, 20, 30, 40, 50 }
```

Paani  
Break

2 min

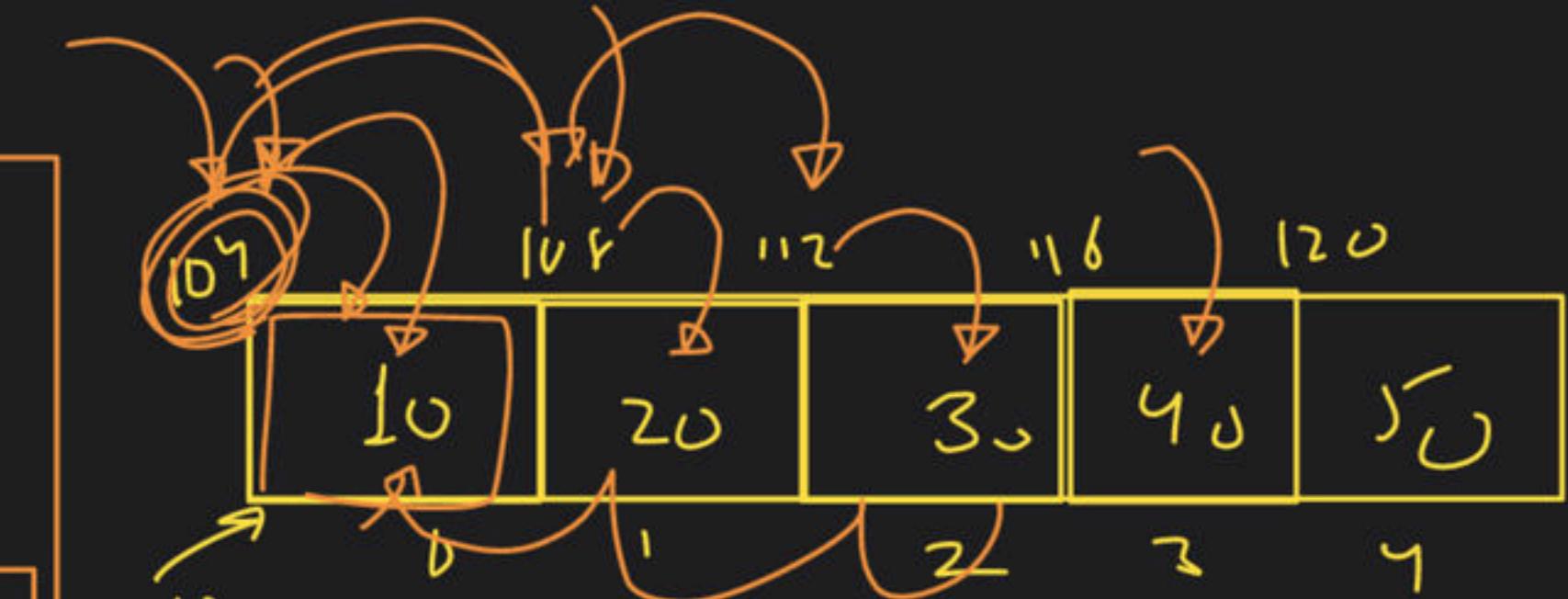
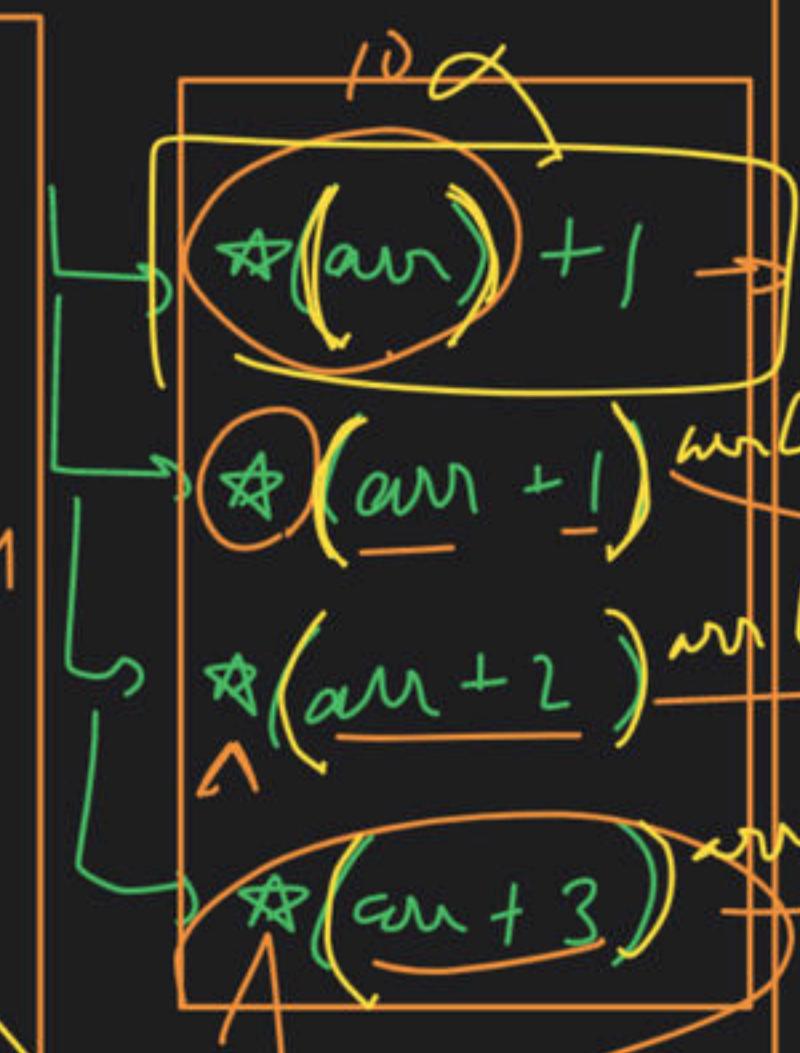


$\text{arr}$   
 $\text{base}$   
 $\text{arr}[0]$ )  
 $\rightarrow \text{BA}$

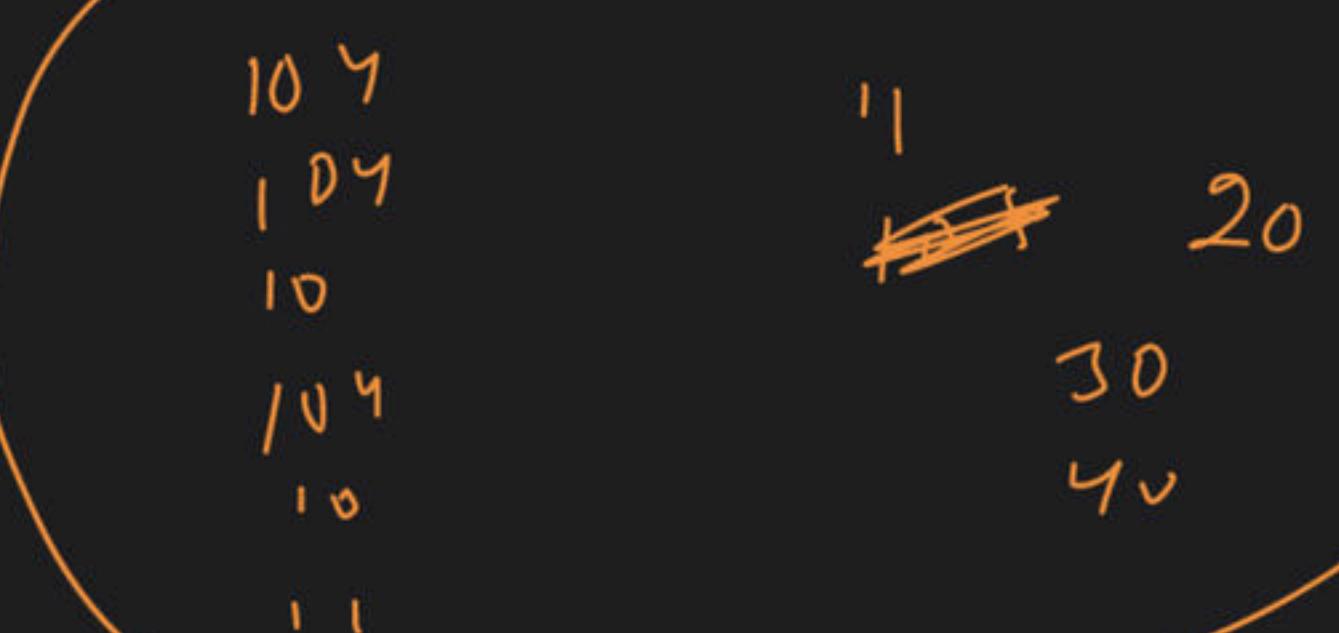
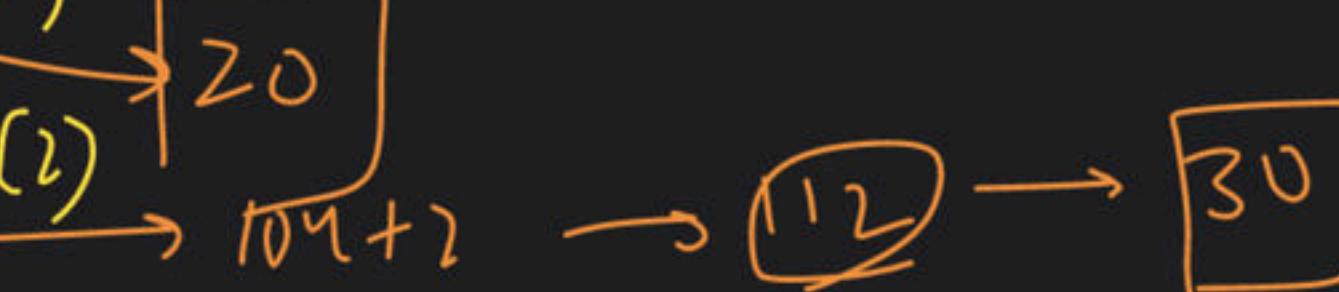
int  $\text{arr}(5) = \{10, 20, 30, 40, 50\}$

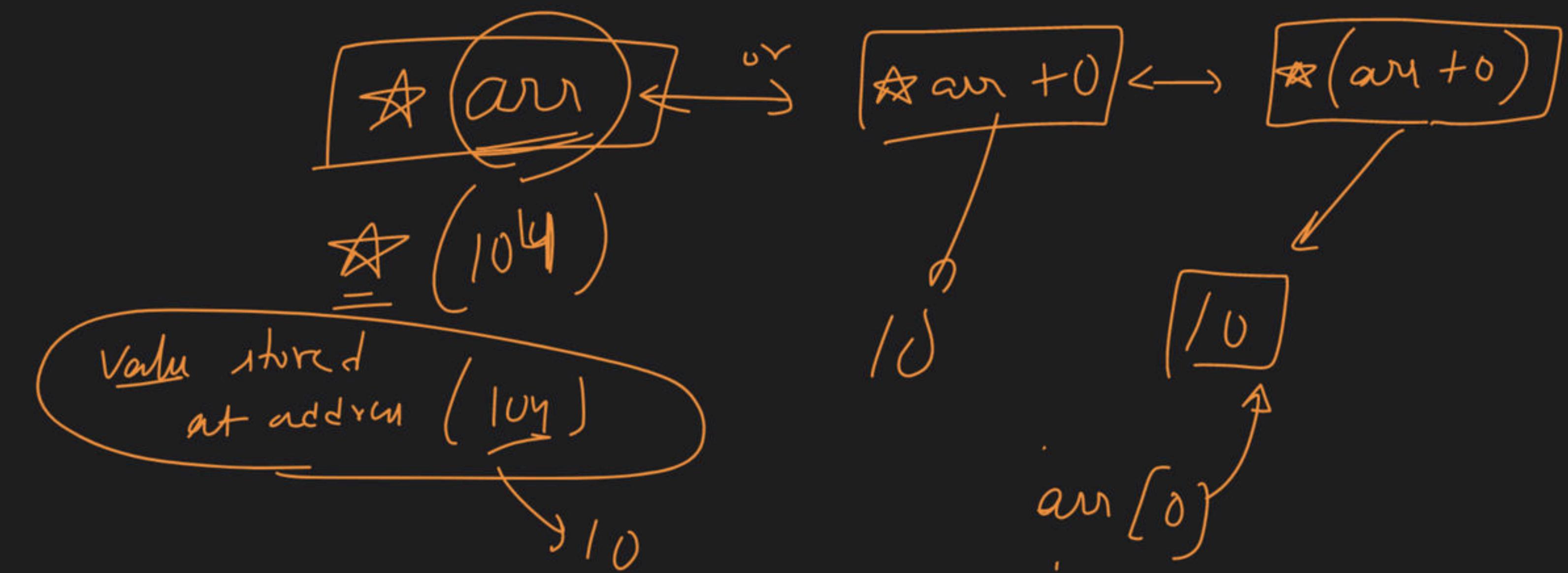
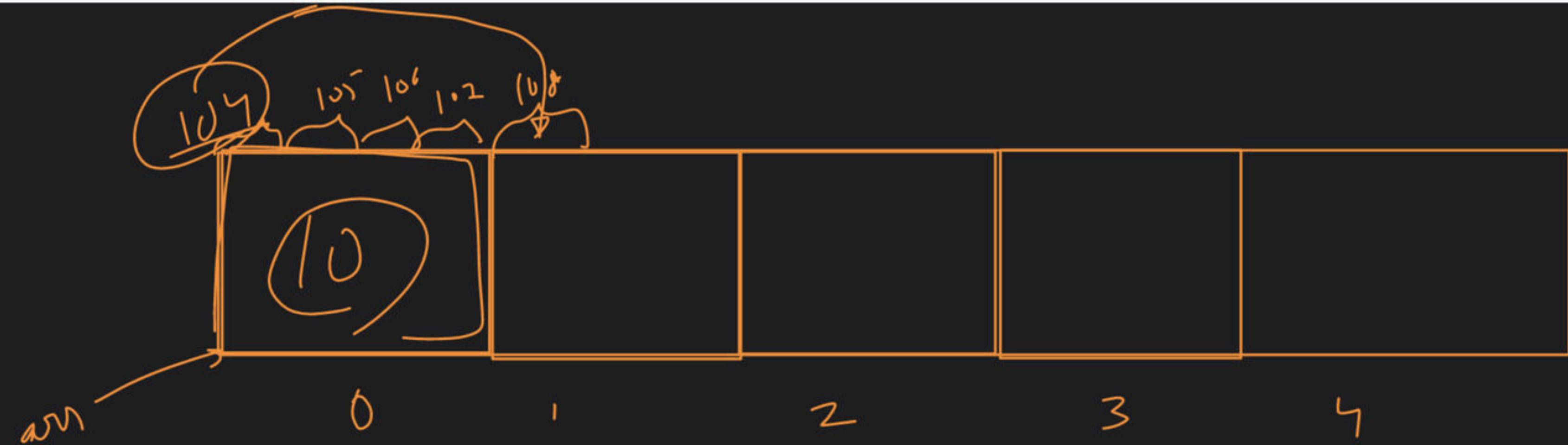
$\rightarrow \text{print}$

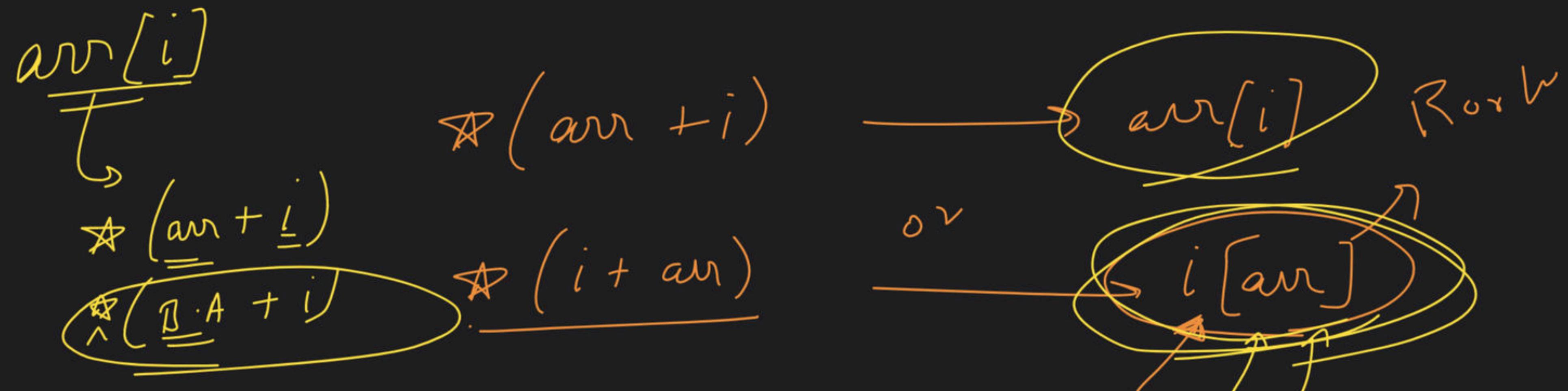
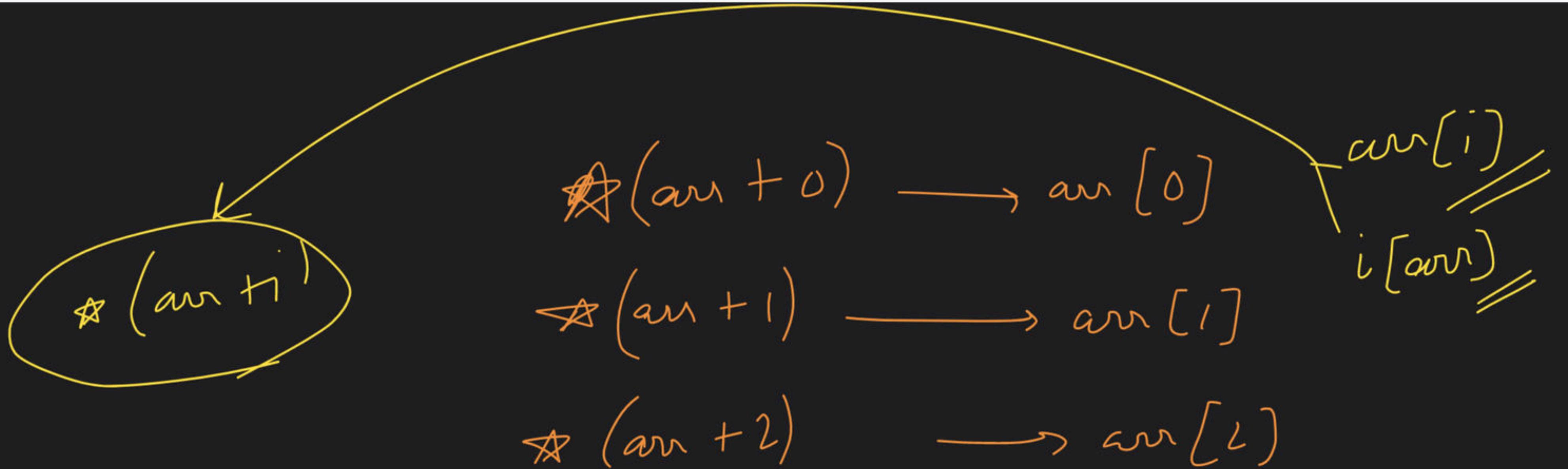
2 min



$$10 + 1 = 108$$







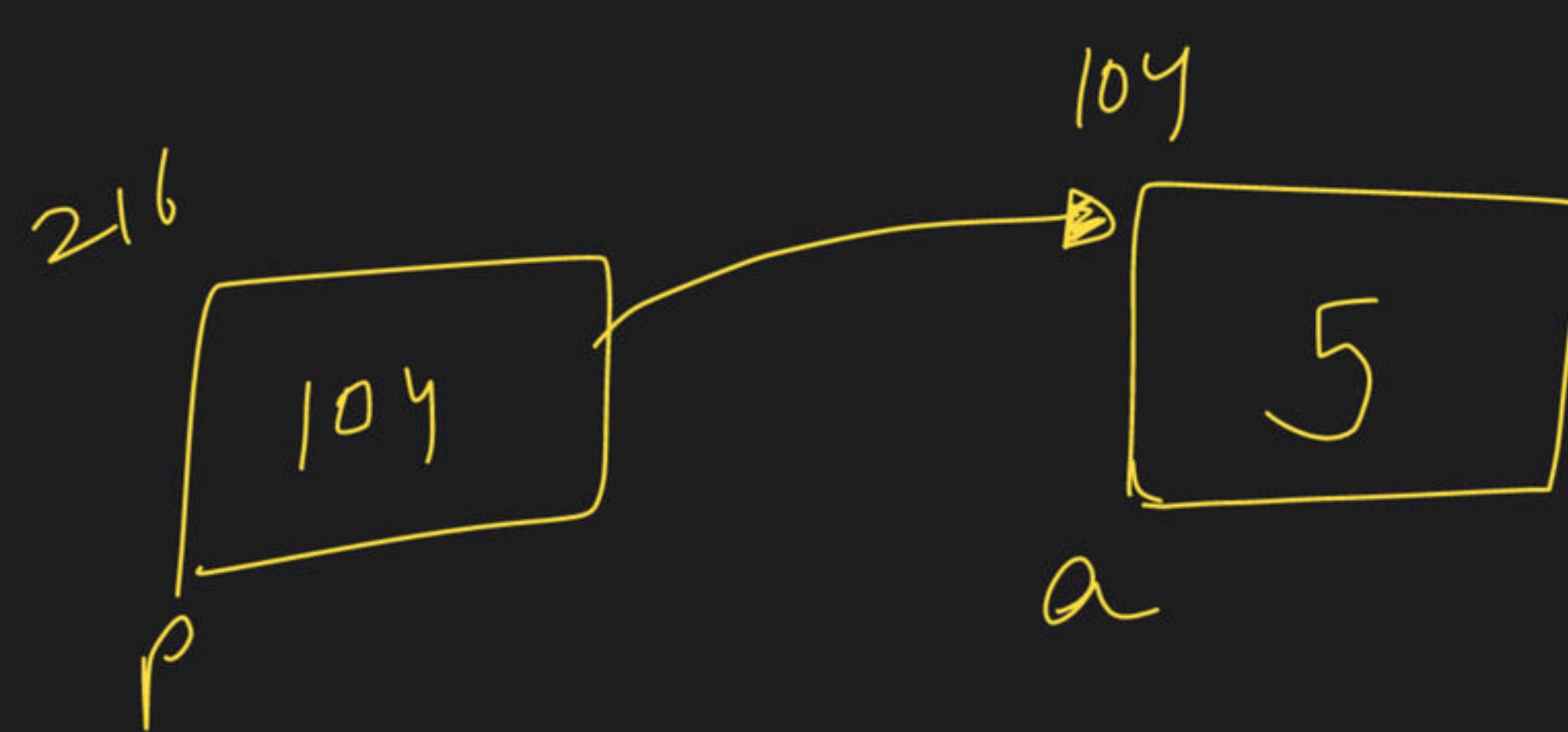
$$\star(a\mathbf{v} + \mathbf{i}) \rightarrow a\mathbf{v}[i]$$

Nutk'.

int a = 5;

int \*p = &a

p = p + 1



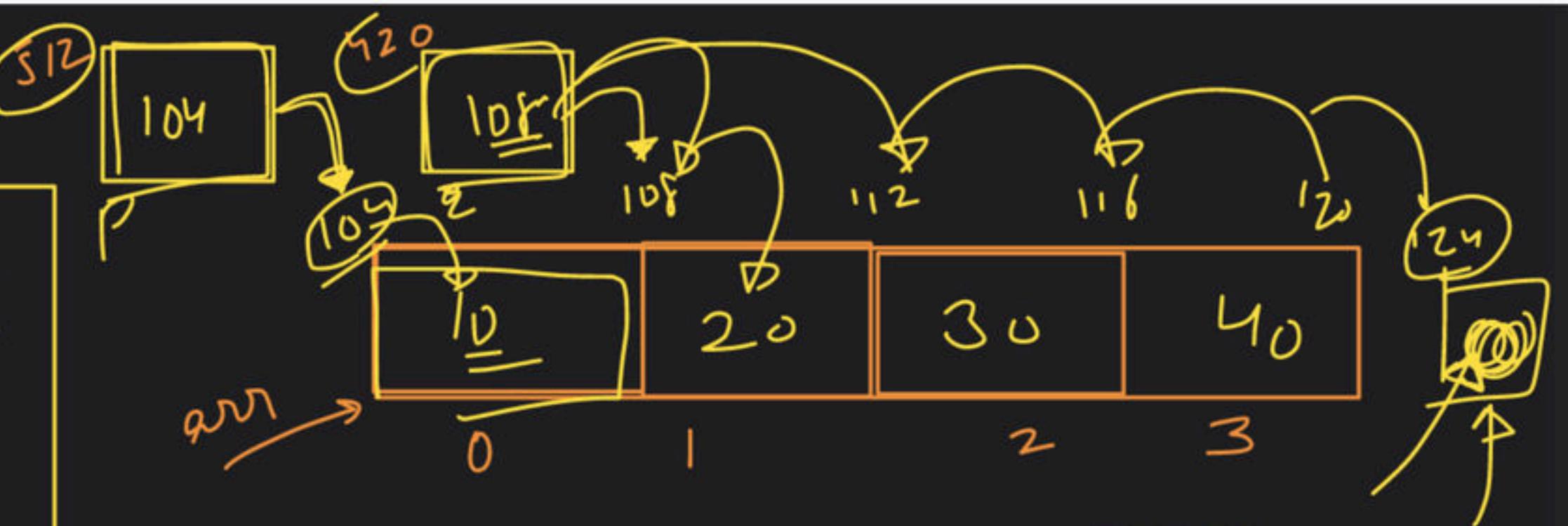
int arr[5] = { 10, 20, 30, 40, 50 }

HW

arr = arr +  
why not  
in this case →  
comparable

$\rightarrow \text{int arr[4] = \{10, 20, 30, 40\}}$   
 $\text{int *p} = \underline{\underline{\text{arr}}};$   
 $\text{int } g = \underline{\underline{\text{arr} + 1}};$

*2 min*

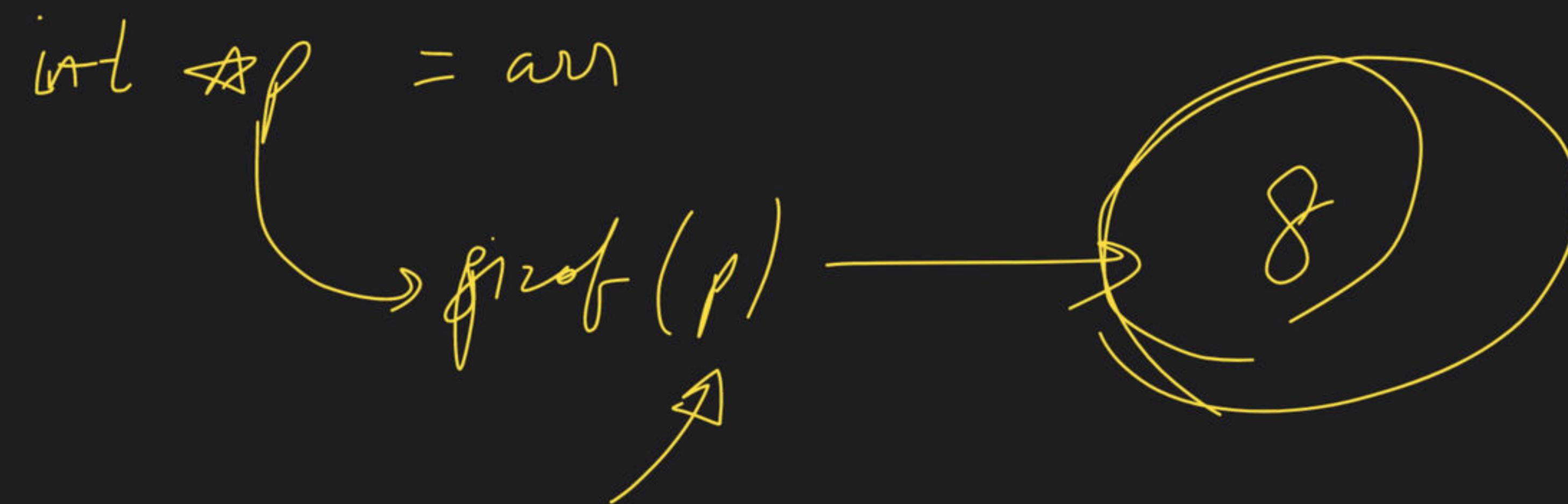


$\text{arr}$   
 $\&\text{arr}$   
 $\&\text{arr}[0]$   
 $\&\text{arr}[0]$

$\&p$   
 $\&P$   
 $\&1$   
 $\&g$

$$\begin{aligned}
 &\star p \rightarrow 10 \\
 &\star (p) + 2 \\
 &\star (1) + 2 \\
 &\star (g + 1)
 \end{aligned}$$

$10 + 2 = 12$   
 $20 + 2 = 22$   
 $10$   
 $10$   
 $10$   
 $10$   
 $512$   
 $\rightarrow 10$   
 $10$   
 $420$   
 $\rightarrow 20$





char ch [50] = "love"

char \* cptr = ch

cout << cptr

int arr[10] = {10, 20, 30}

~~\*it~~ \*p = arr

cout << p

10  
104

10  
104  
arr

\*cptr  
\*(cptr)  
\*(cptr + 0)  
cptr[0] →

$\text{cout} \ll \text{ch}$

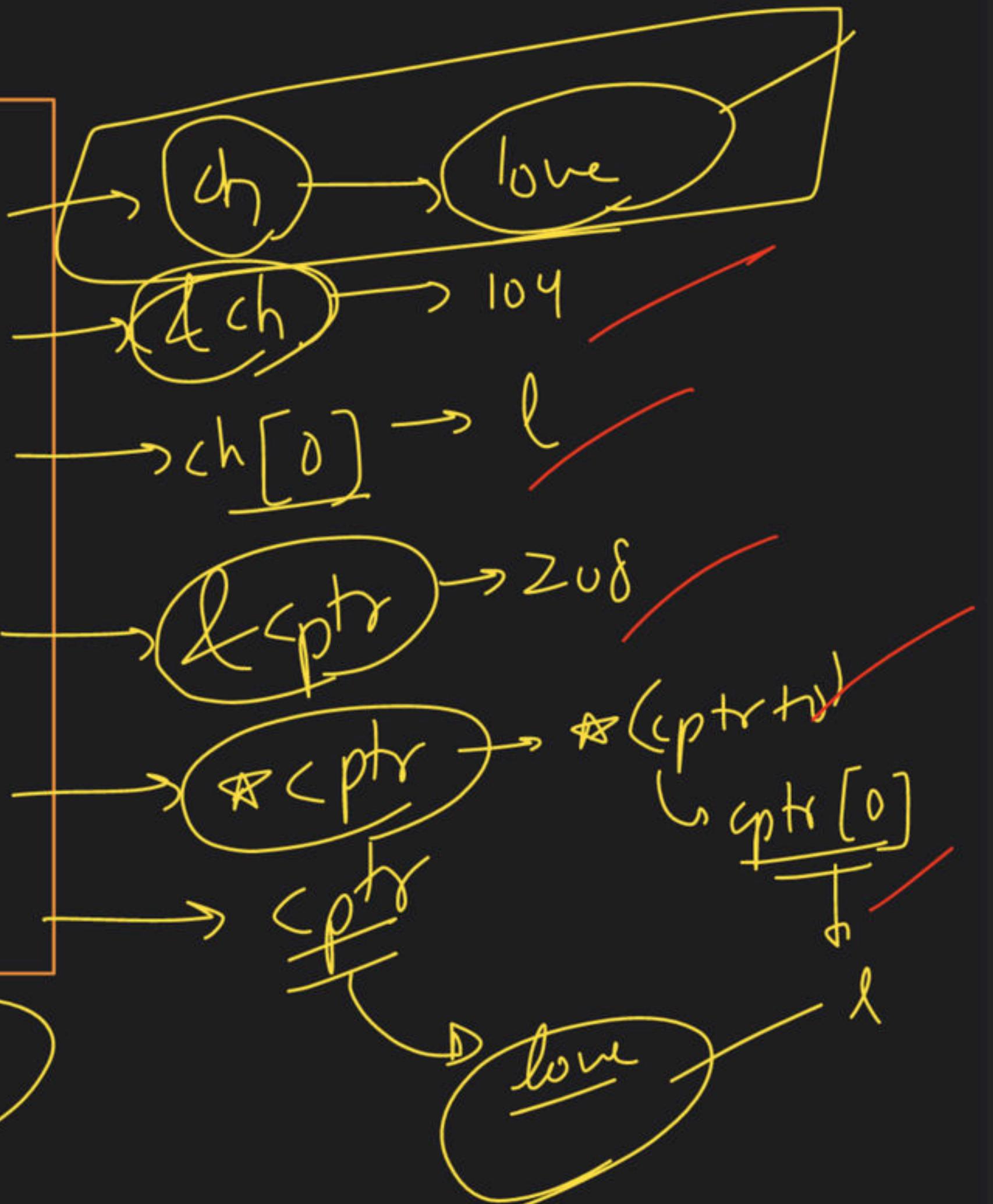
char  $\text{ch}[5]$  = "love";

char \*  $\text{cptr}$  =  $\text{ch}$ ;

$\text{cptr}$  is a pointer  
to  $\text{char}$   $\text{ch}$

$\text{ch}[0]$

2 min



Codchup



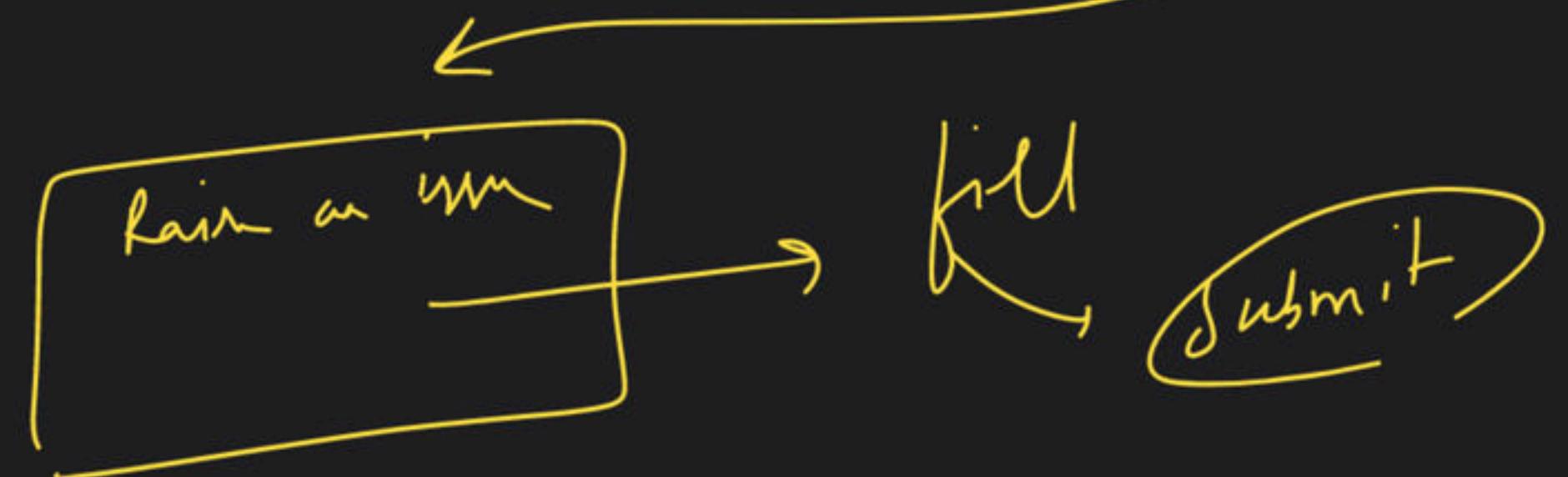
Bottom Right



fairly an  
Ism



click

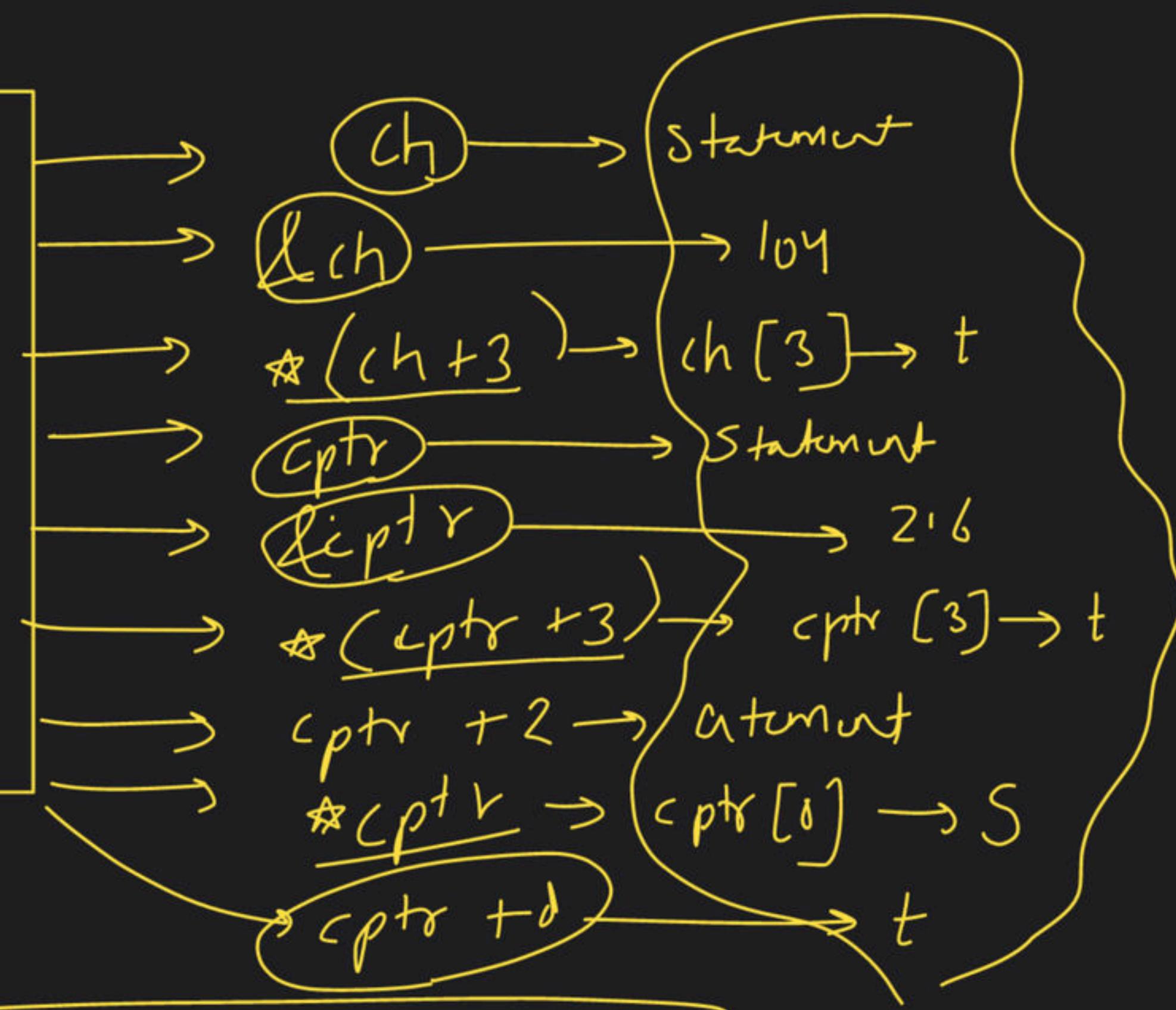


$\text{char} \text{ ch}[30] = "Statement";$

$\text{char} * \text{ cptr} = &\text{ch}[0];$

$\text{char} \text{ ch}[30]$

$\text{char} * \text{ cptr}$



$\text{char} \text{ ch}[30] = "Statement";$

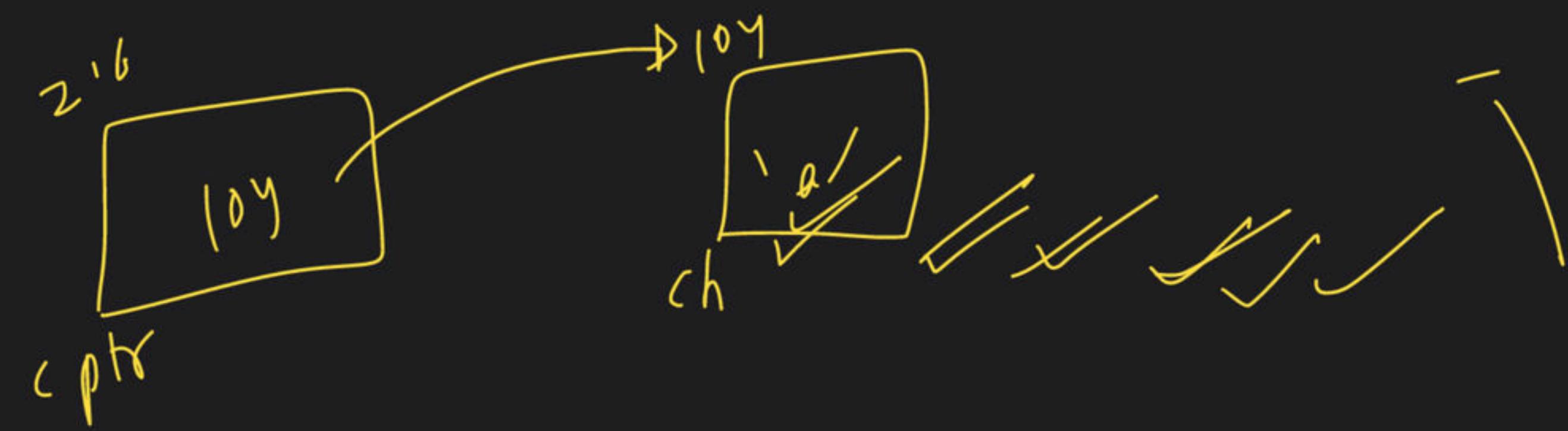
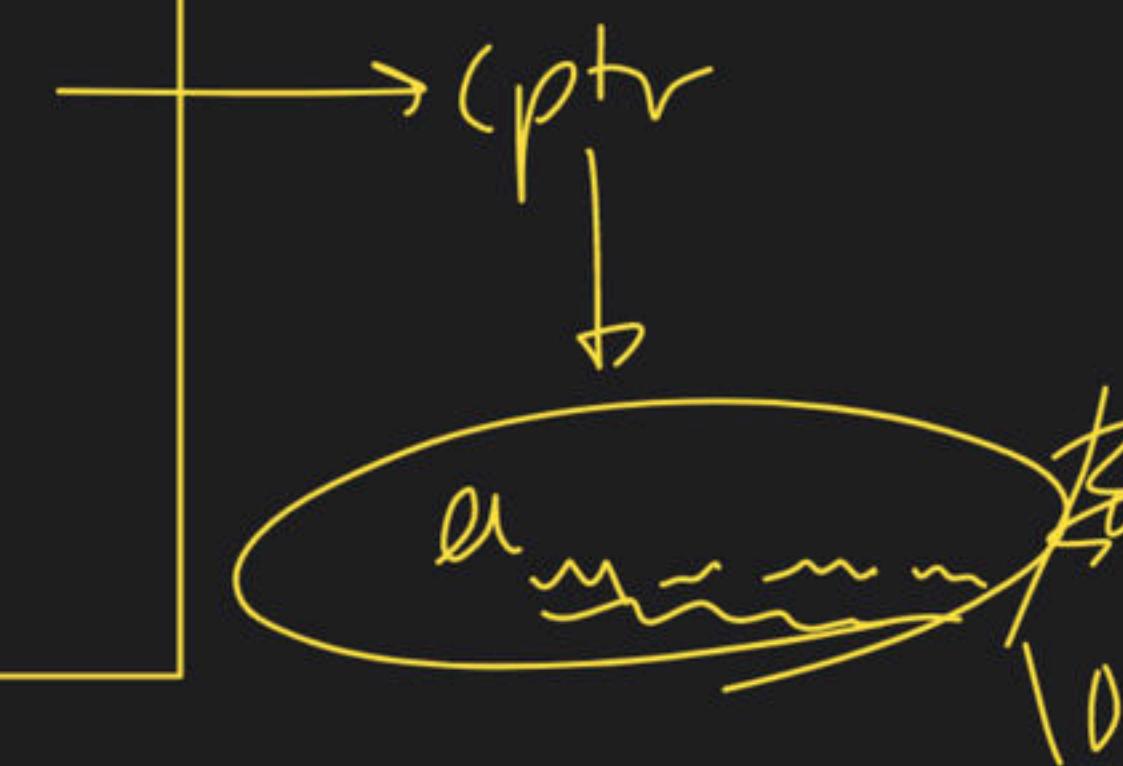
$\text{char} * \text{ cptr} = &\text{ch}[0];$

$\text{char} \text{ ch}[30]$

$\text{char} * \text{ cptr}$

char ch = 'a'  
char \* cptr = &ch

cout << cptr



char ch[10] = "Babbar";

char \* c = ch;

~~char~~  
~~char~~  
~~char~~

char \* cptr = "Babbar";

cout < cptr

possible  
temp storage

why?

SLEET

TD

Our last  
Design

after 26<sup>th</sup> Oct