



Pointers - II

Foundation Course on Data Structures & Algorithm - Part I

Homework → D is Not

→ Double pointers :-

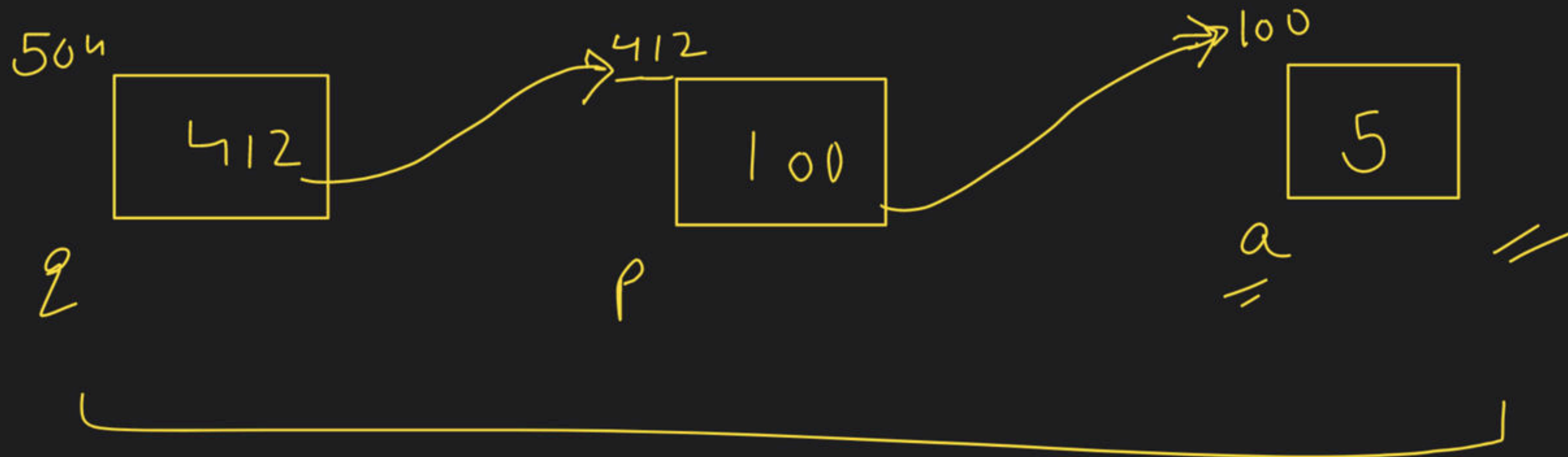
— (1) $\text{int } a = 5 \rightarrow \underline{\underline{\text{BTS}}}$

Doubt → $\text{int}^* p = \underline{\underline{\text{BTS}}} \rightarrow$
↓
address of a

— (2) p is a pointer
to integer data

— (3) $\text{int}^{**} q = \underline{\underline{\text{BTS}}} p; \rightarrow ?$
→ q is a pointer to int^* data

Let's class
repeat



→ why → ? need ?
↓
dekh na h

① $\text{int } a = 5$

② $\text{int } *p = \&a$

③ $\text{int } **q = \&p$

④ $\text{int } ***r = \&q$

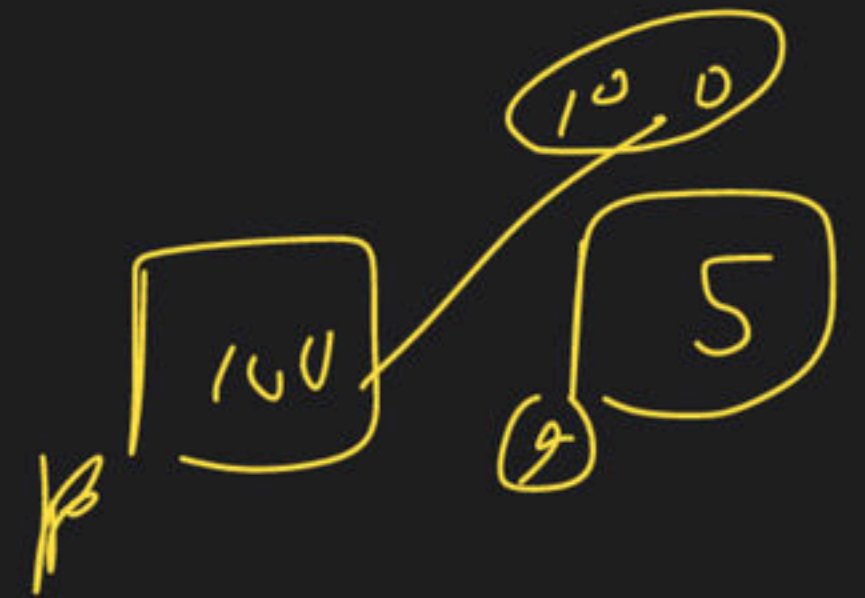
⑤ $\text{int } ****s = \&r;$



→ &a → address → hexa

→ a → (5)₂

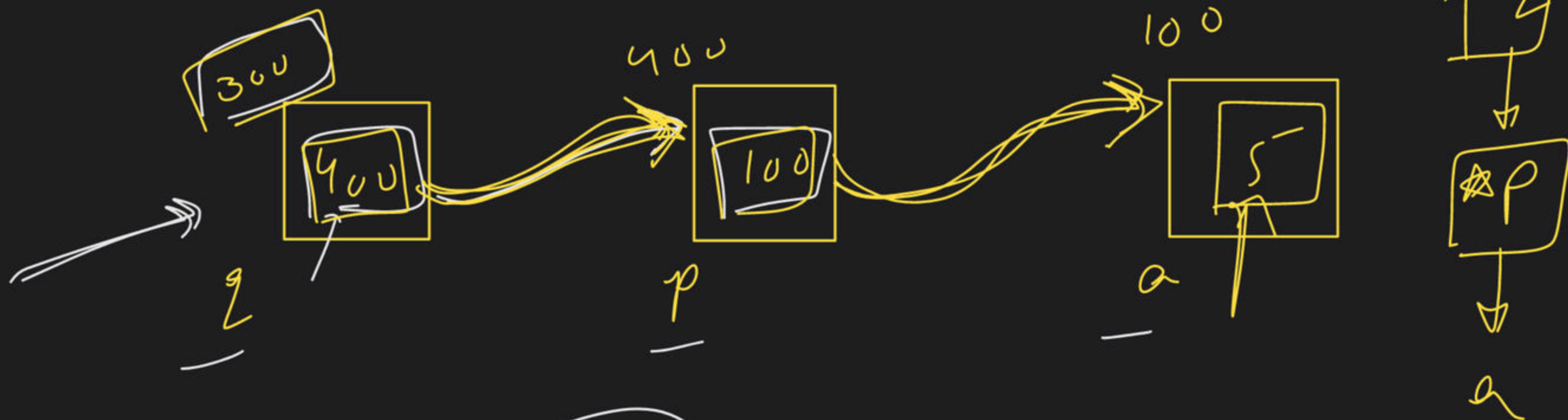
→ *a → error → why - ?



→ &p → [address of p]

→ p → [address of a] →
 var → (entry) string

→ *p → (5)



q → (300)

q → (400)

q

q → (5)

value at address pointed by q

→ 100

→ How many levels? → ?

✱ ✱ ✱ ✱ ✱ ✱

→ int main()

{
Bjpr
}

fun()

mpur
}

fun()

{
p = p + 1;
}

Assignment

Confidence

int $\star p = 0$
 $p = \&f$

float $f = 10.5$
 float $p = 2.5$

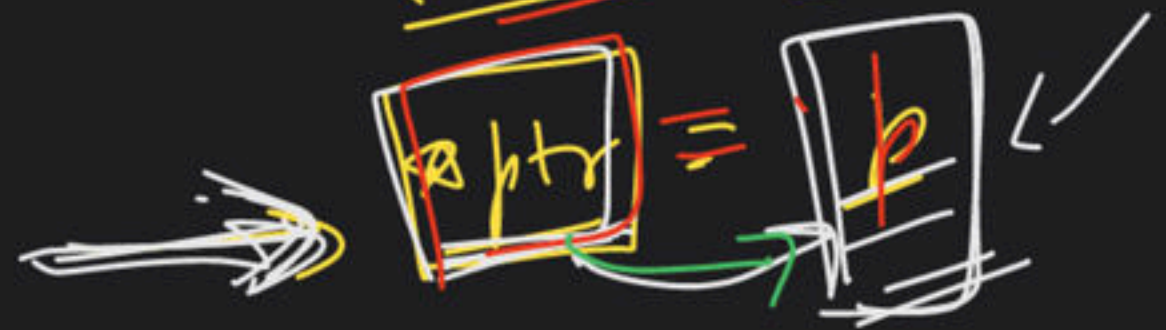
Code Studio
 → 6th
 → 1B
 100 q → p

2.5

float $\star ptr = \&f$

$\star ptr$

$(\star ptr)++$

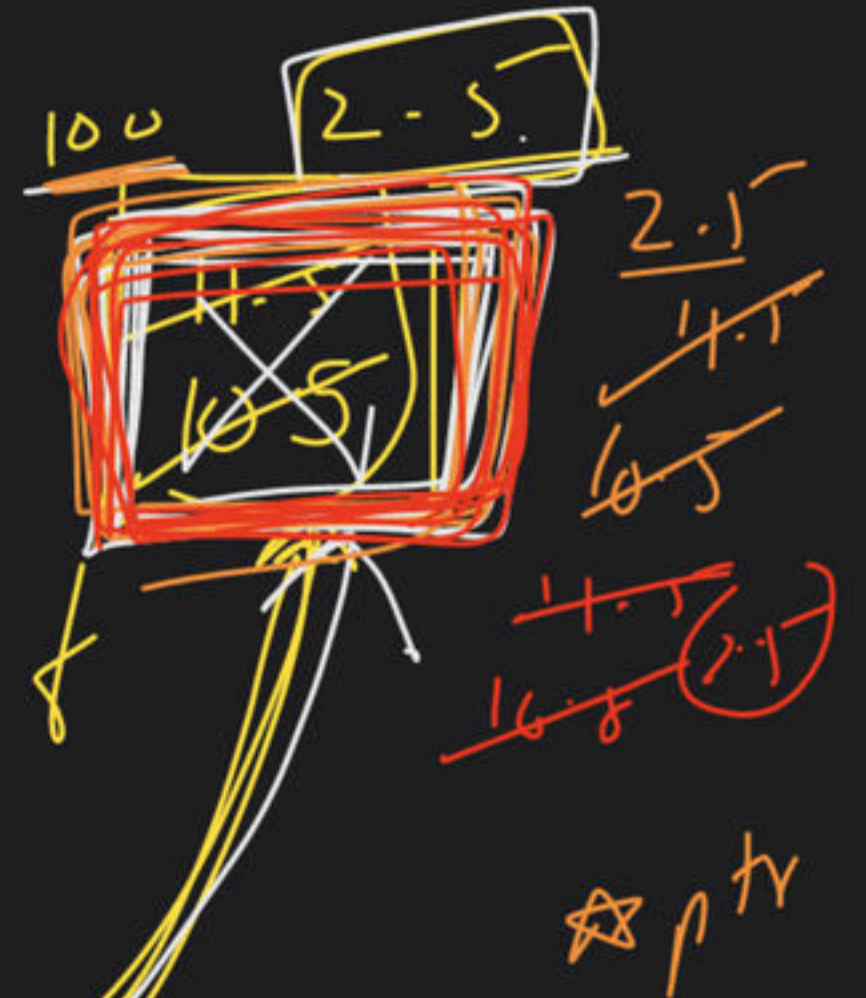


$\star ptr$

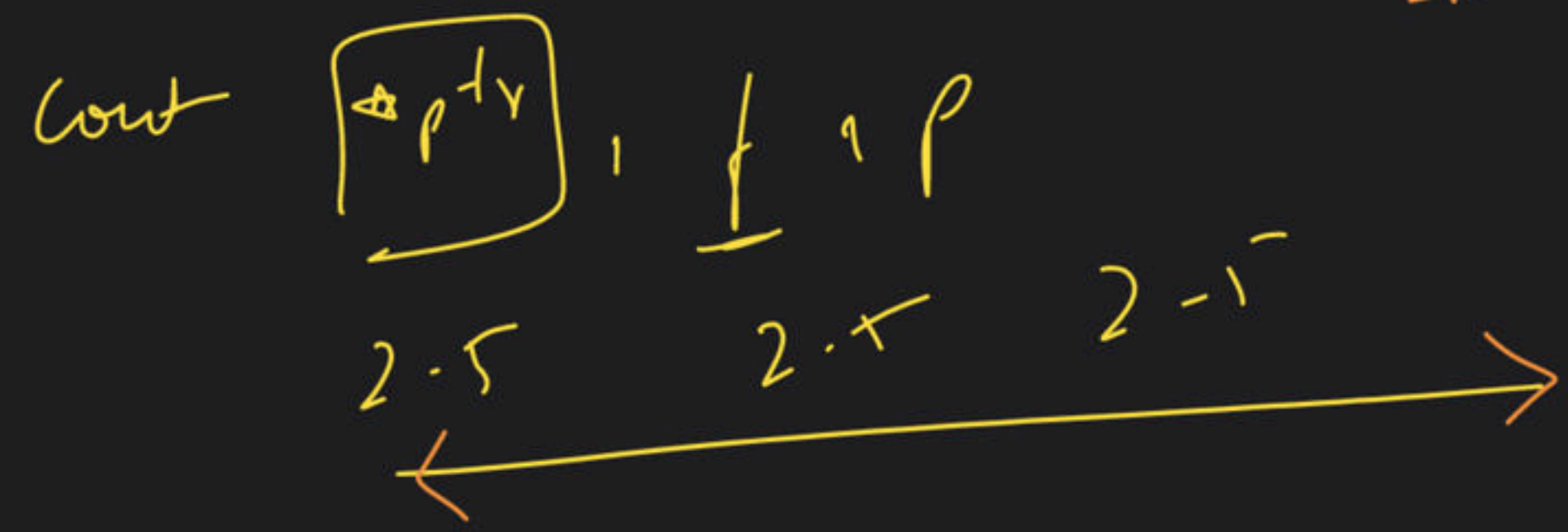
$\star ptr$

$\star ptr$
f

100



$\star ptr \equiv f$



30 MCQs
 1 MIN

→
int a = 7;

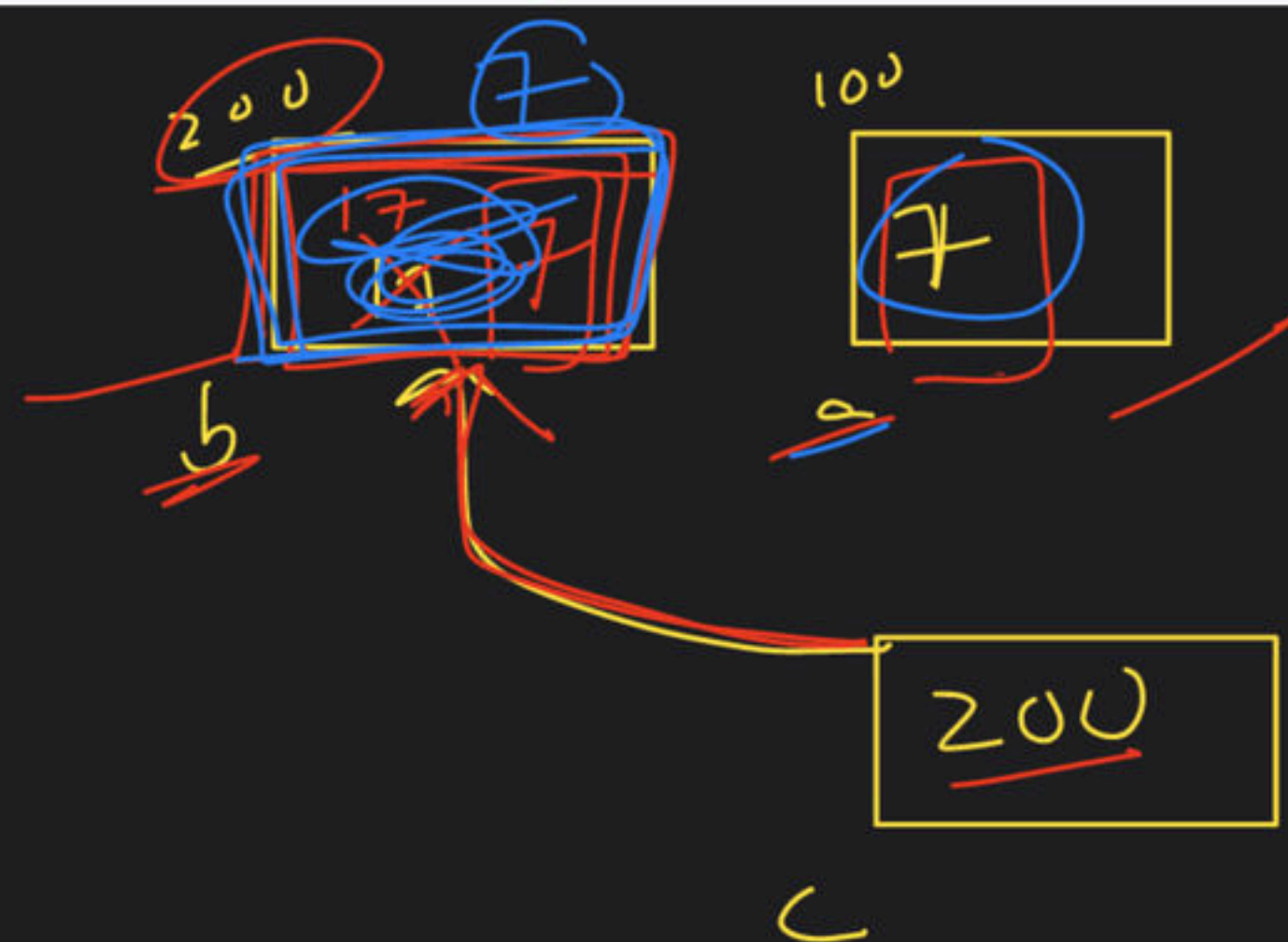
int b = 17;

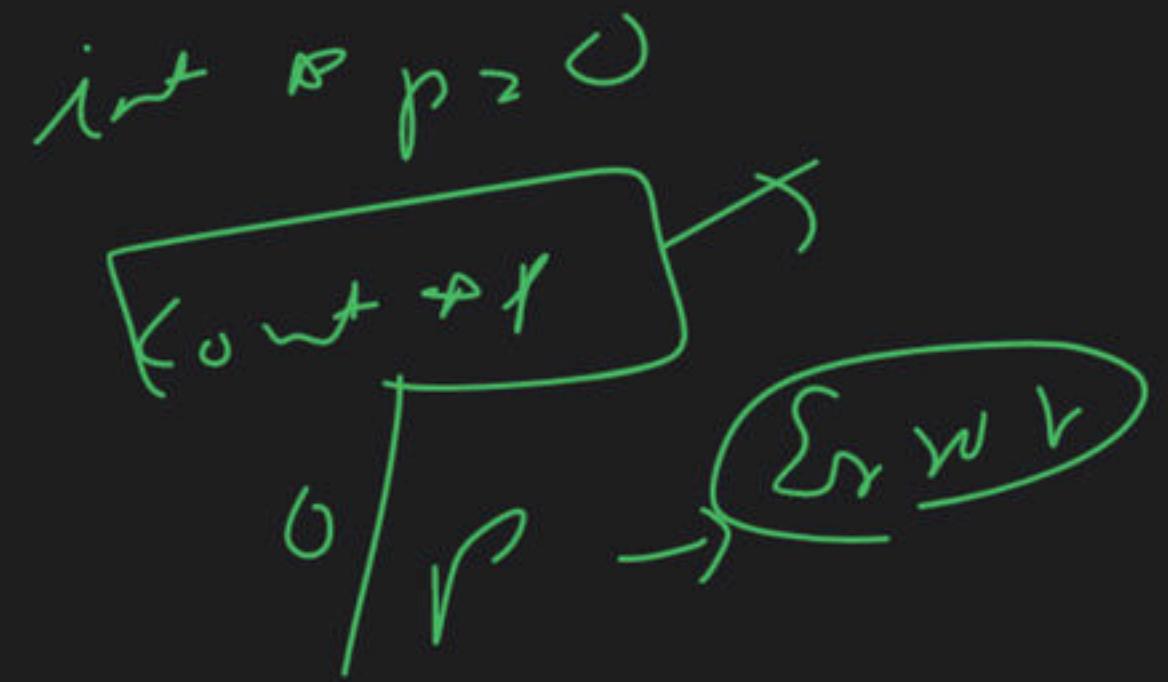
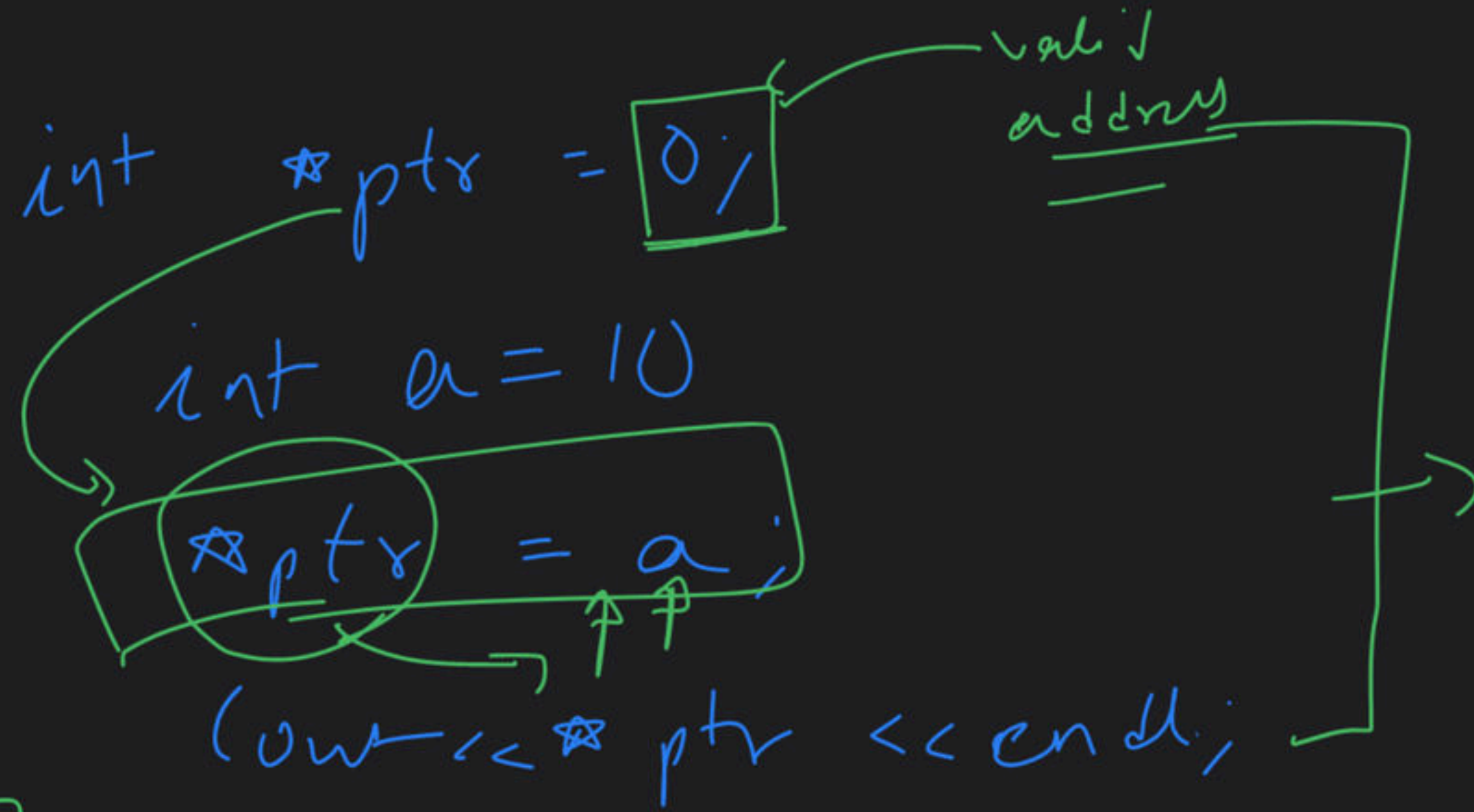
int *c = &b

*c = 17

cout << c << b << endl;
7 17 =

*c





`int *p = &a`

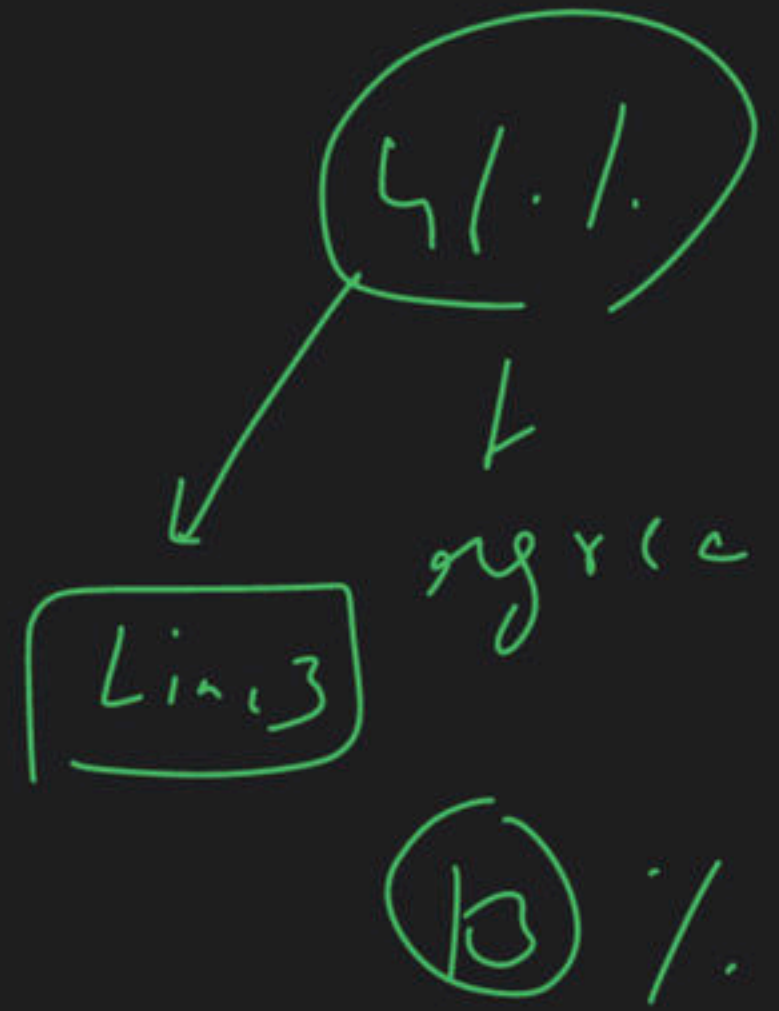
or

10

error

`int *p = 0`

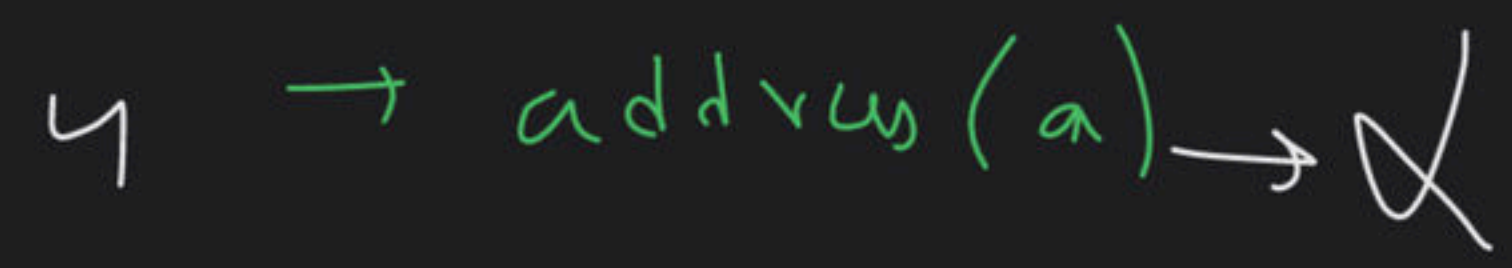
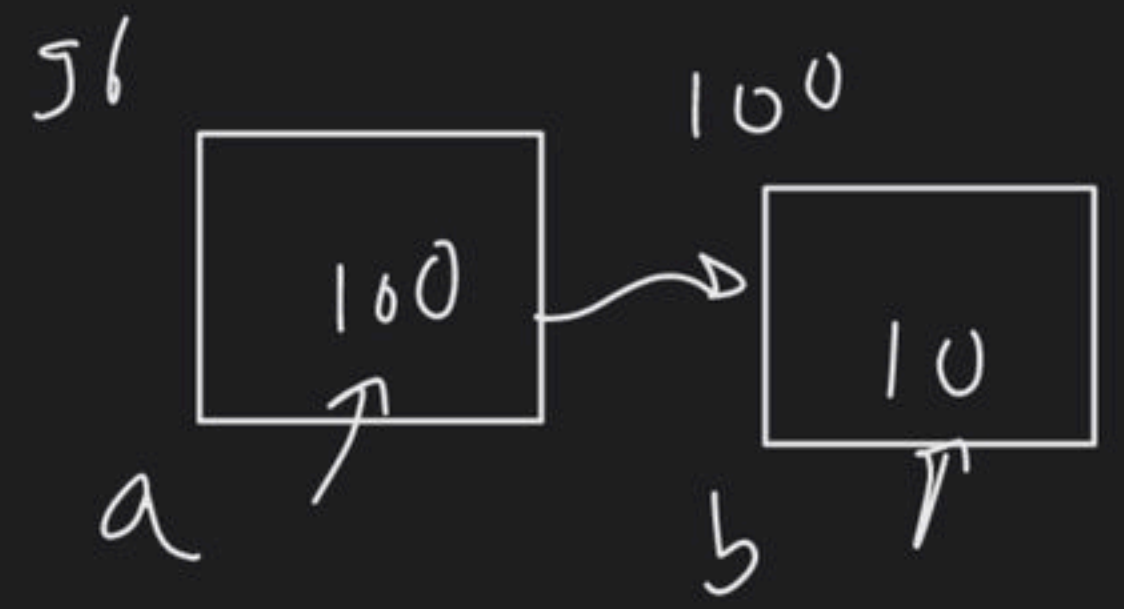
`p = &a` (The `&` is boxed)



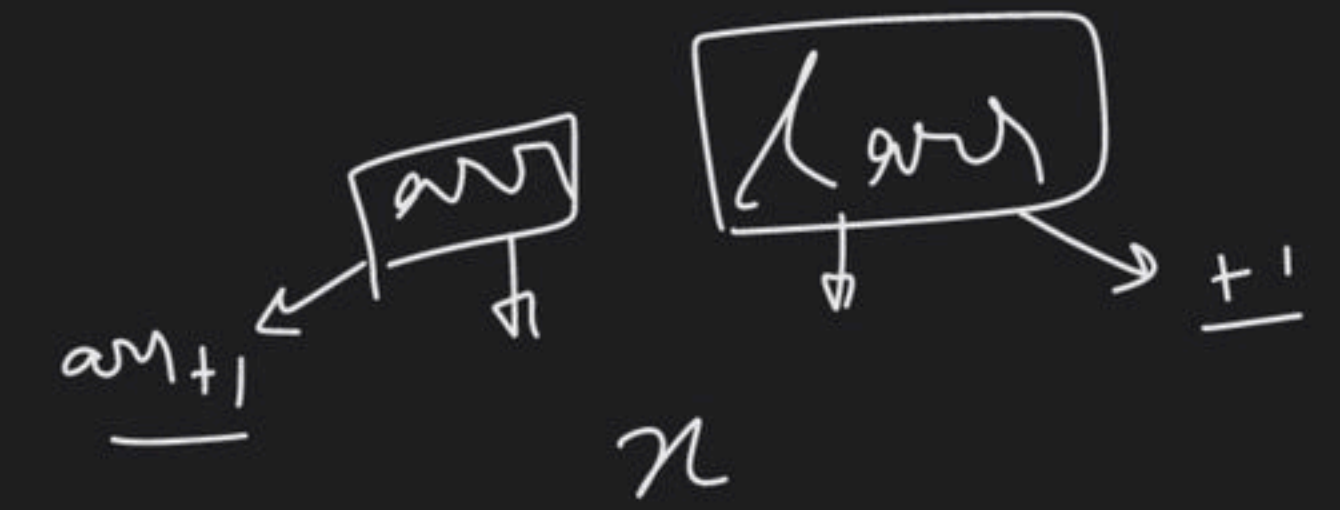


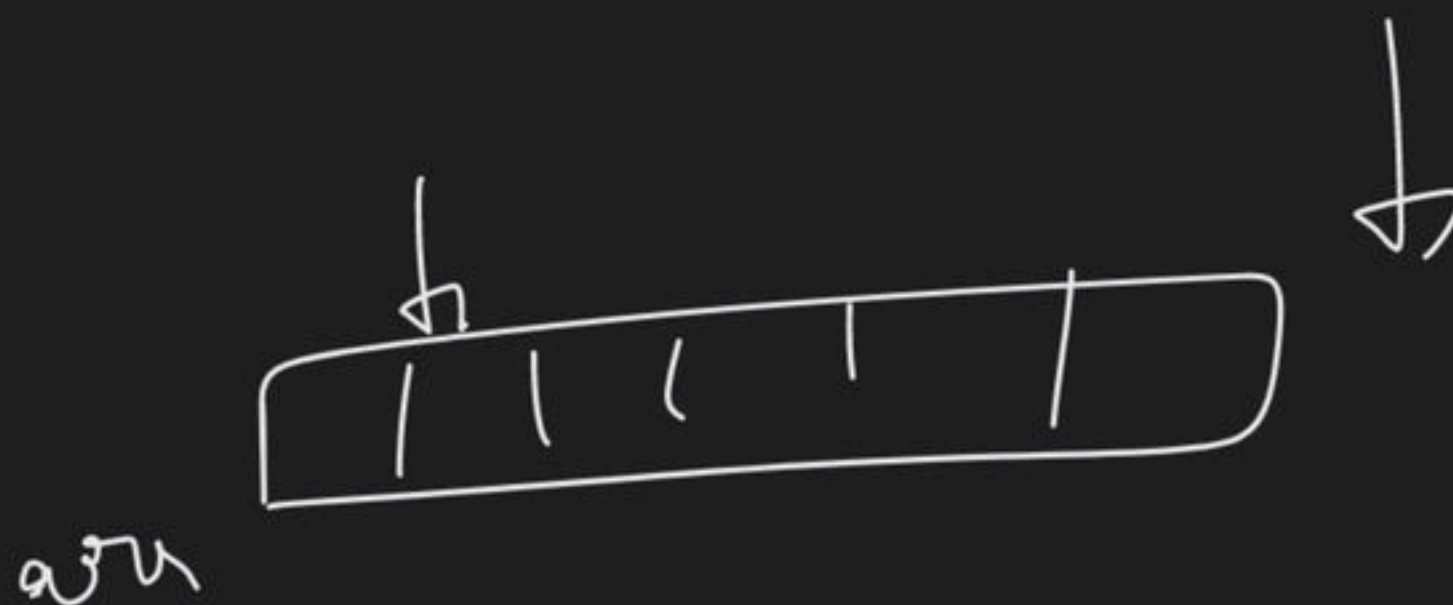
int b = 10

int *a = &b;



b ka address → 100



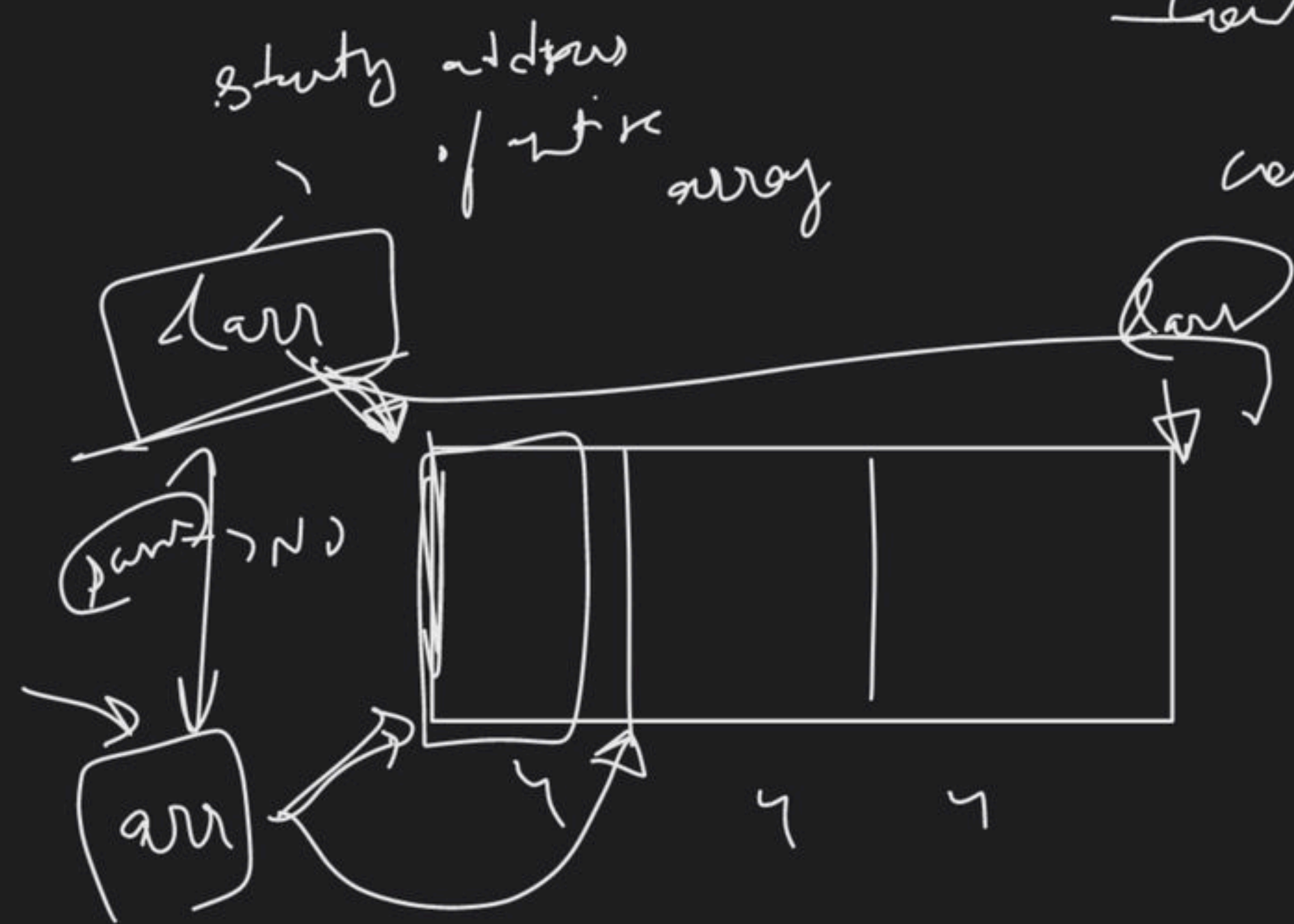


$count < arr + 1$

$count < arr + 1$

$\boxed{arr + 1}$

$\underline{arr \neq 1}$ — $\boxed{146}$





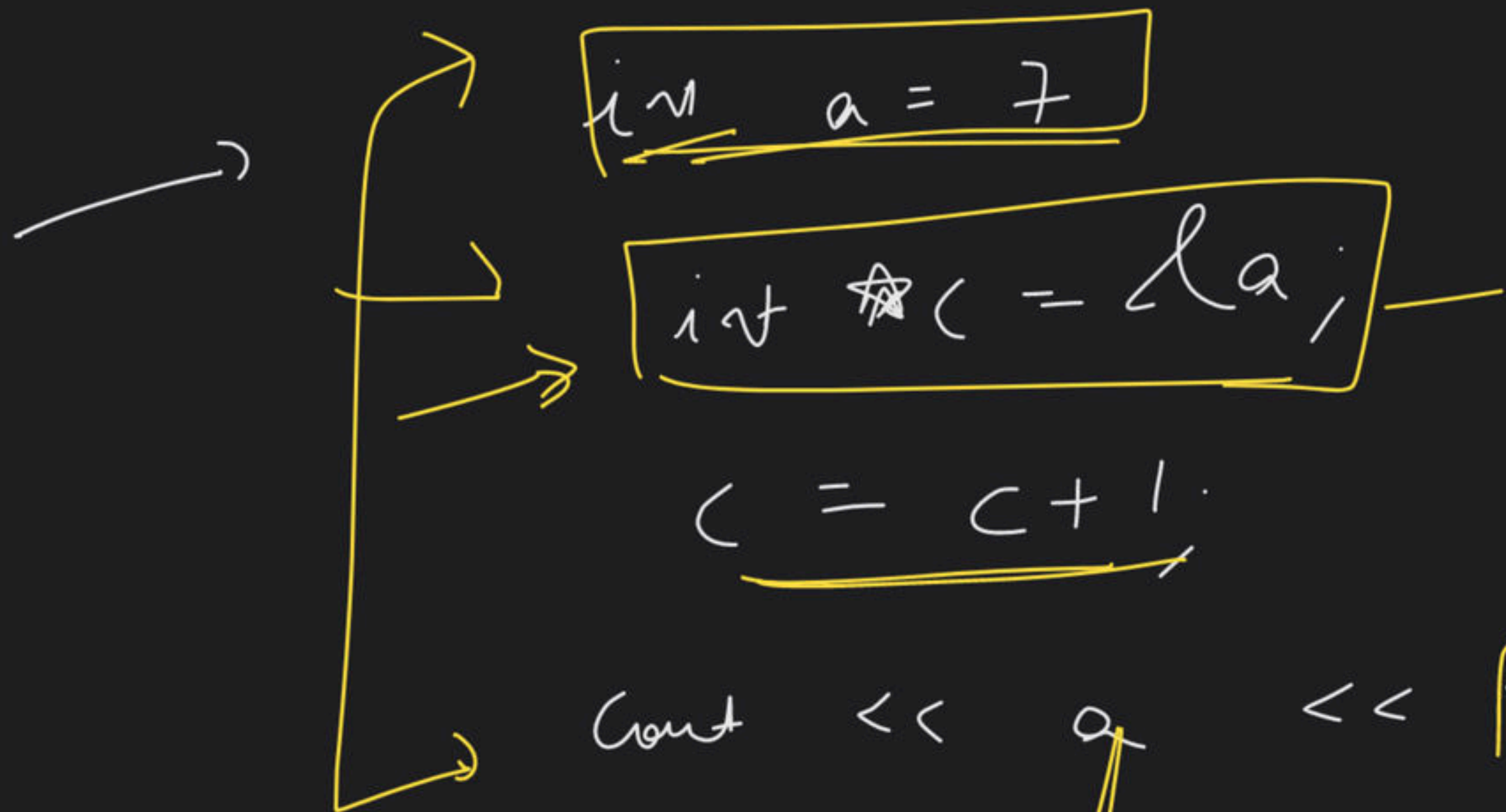
char ch = 'a'

char *ptr = &ch

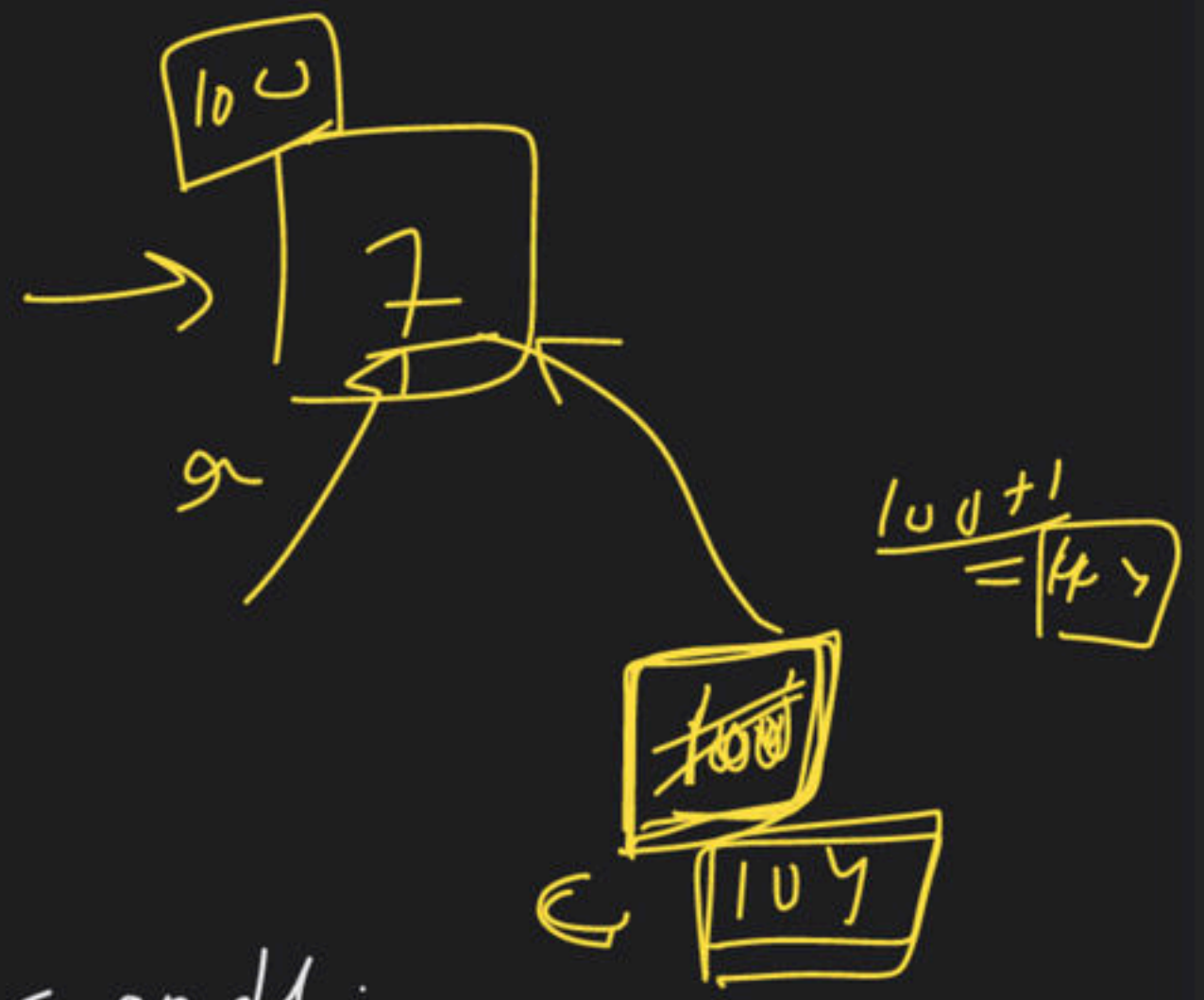
ch++;

cout << *ptr;





`cout << a << *c << endl;`

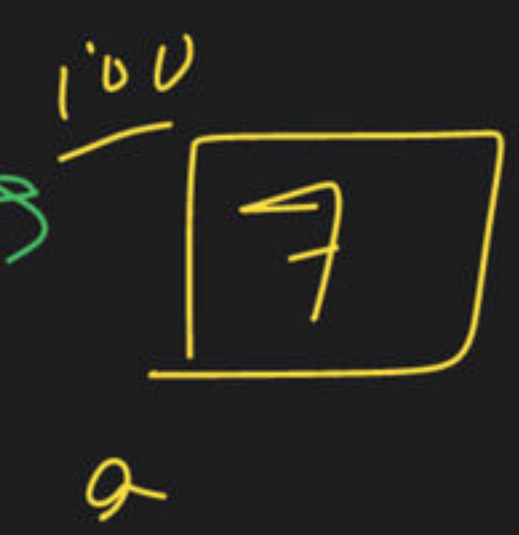


64 bit → 7

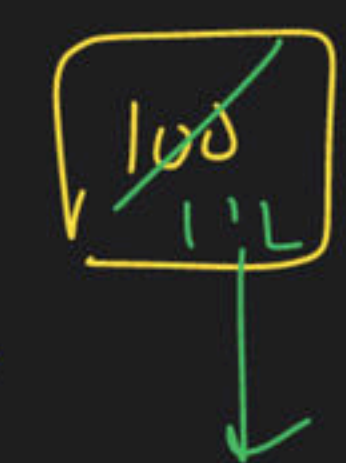
64 bit



int a = 7



int *c = &a



c = c + 3;

cout << ~~c~~
ans

doubt

garbage
112

3×4
 $100 + 3$
 $=$
 $100 + 3 \times 4$
 $= 2(112)$



double $e = 10 \cdot 54$

downl. * $d = 25$

$d = d + 1$;

(unit <> d) ?

108

100

10.54

a

100+1

100 $\approx \frac{104d}{108}$

d

104 → ?

(108) → why → ?

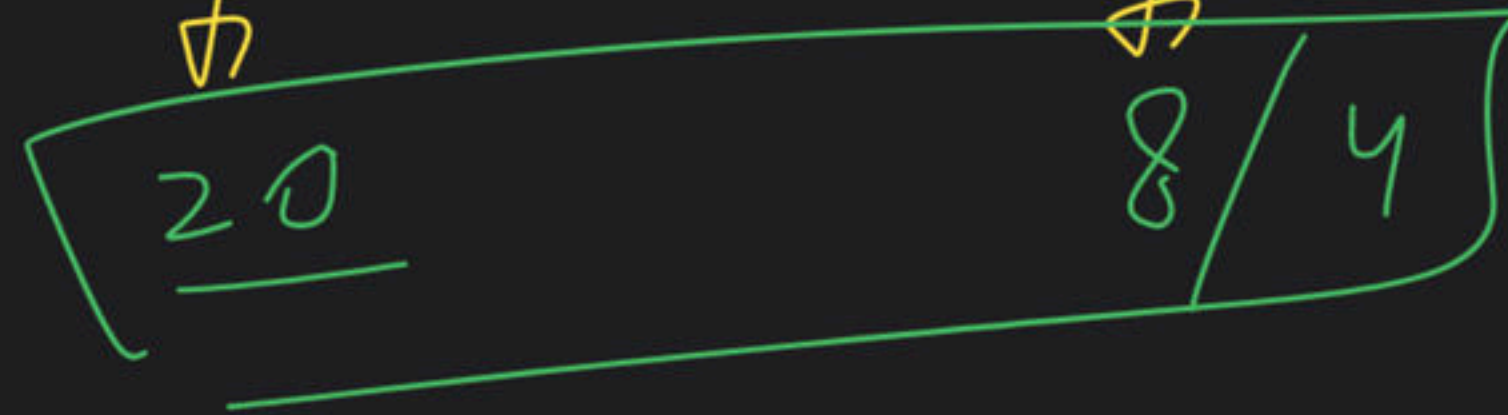
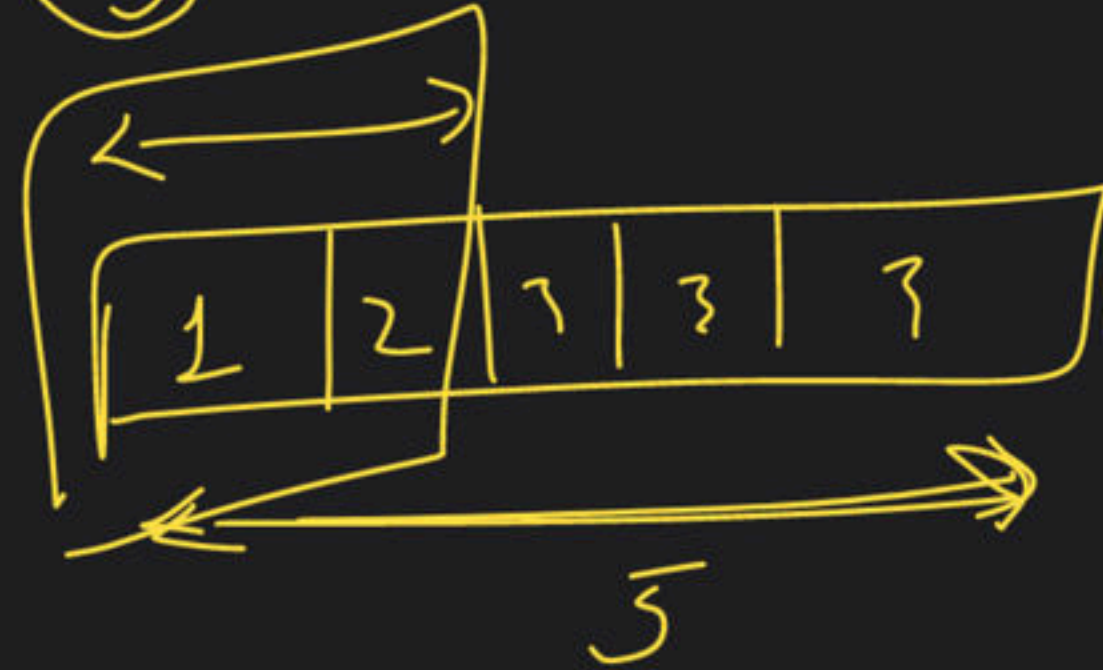
1 → 4b,
5x = 20

→ int a[5];

int *c;

(out << sizeof(a)) << sizeof(c);

(3)





int a[] =

{ 1, 2, 3, 4 }

0 1 2

cout <<

<<

*a

<<

*(a+1)



*a

*(a+0)

a[0]

1

*(a+1)

a[1]

*(a+2)



a[2] = 3

No
chaki

int a[] = {1, 2, 3, 4}

int *p = a++;

cout << *p << endl;

Run

11

Error msg

PC

Error - 1

int arr[] = {4, 5, 6, 7};

int *p = (arr + 1);

cout << *arr + 9;

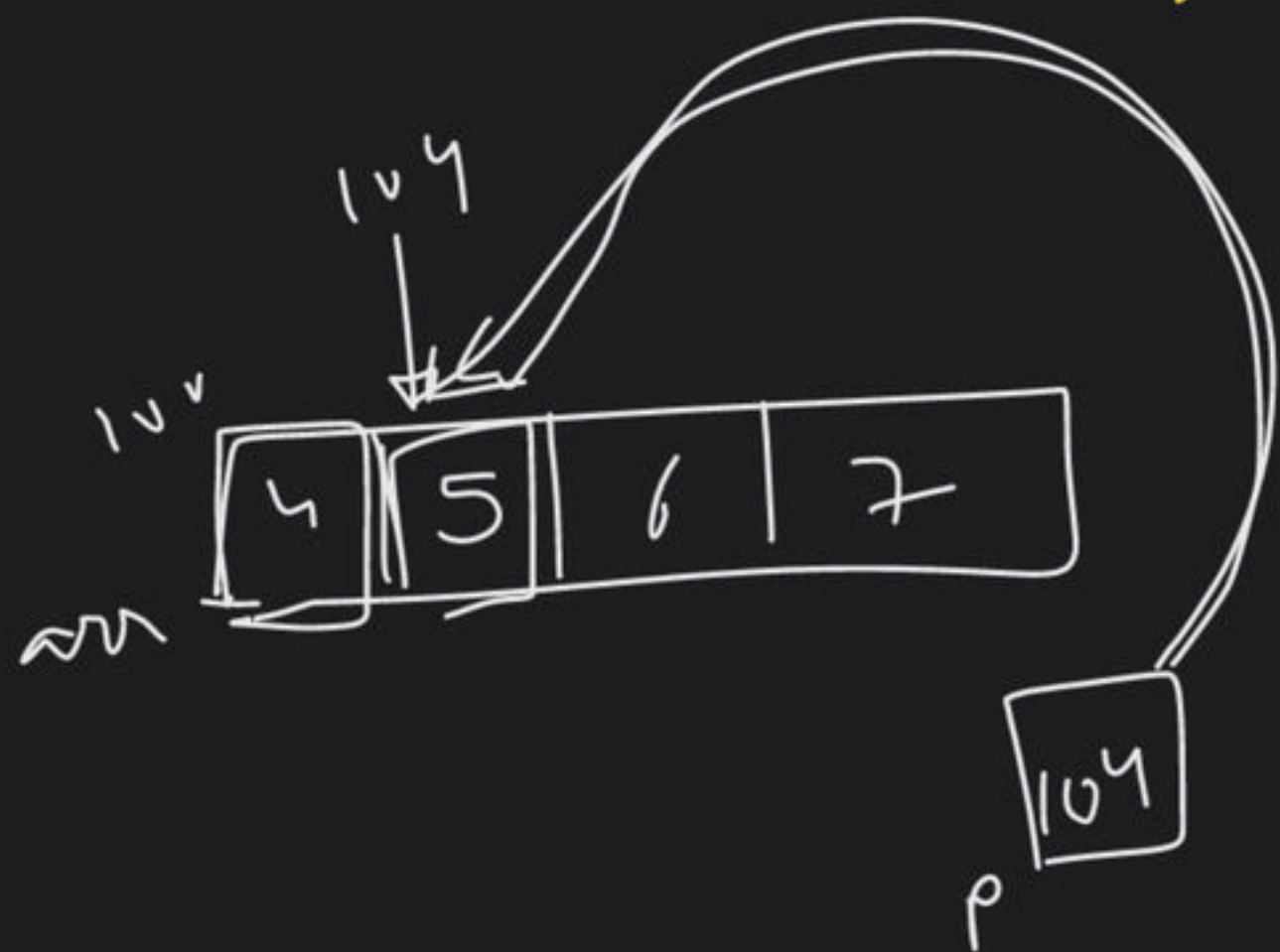
return 0;

13

(13) - IT

(14) - IT

4 + 9 = (13)



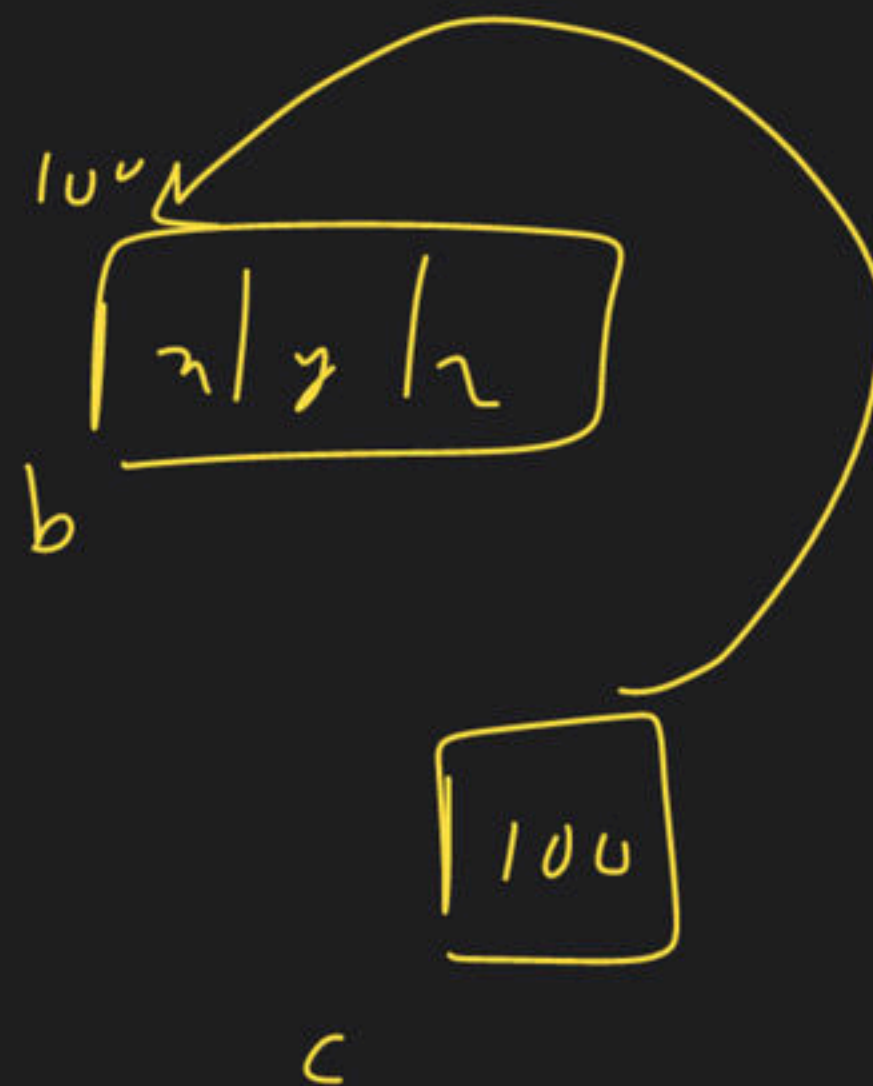
Repeat

char b[] = "xyz";

char *c = &b[0];

cout << c << endl;

↓
x.



4/5

len!
chars

← why

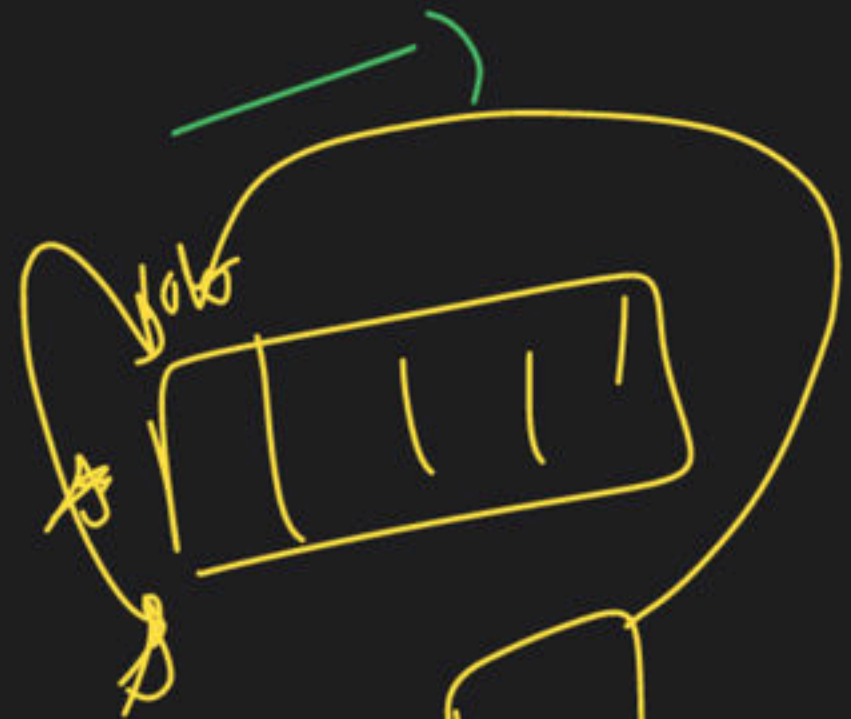
100 →

21.1.

xyz → 93.1.

(s) - base address

char s[] = "hello";



char *p = s;

cout << s[0] << p[0];



- > o/p - >

33%

4%

Downst

9



```
void square (int *p)
{
    int a = 10;
    p = &a;
    *p = (*p) * (*p);
}
```

(100) → Yes or No
No

```
int main ()
{
    int a = 10;
    square (&a);
    cout << a << endl;
}
```

(10) → 54%.

Table

2 June

20 Jun ^

3da y

(10 min)

```
main()
```

PL

~~May - June~~

$$\underline{10^7}$$


1

100
104

20 June

$$p \geq 1$$

ind $\star p = da$

syner (42)

110

f

$$\text{row} < \text{K}$$

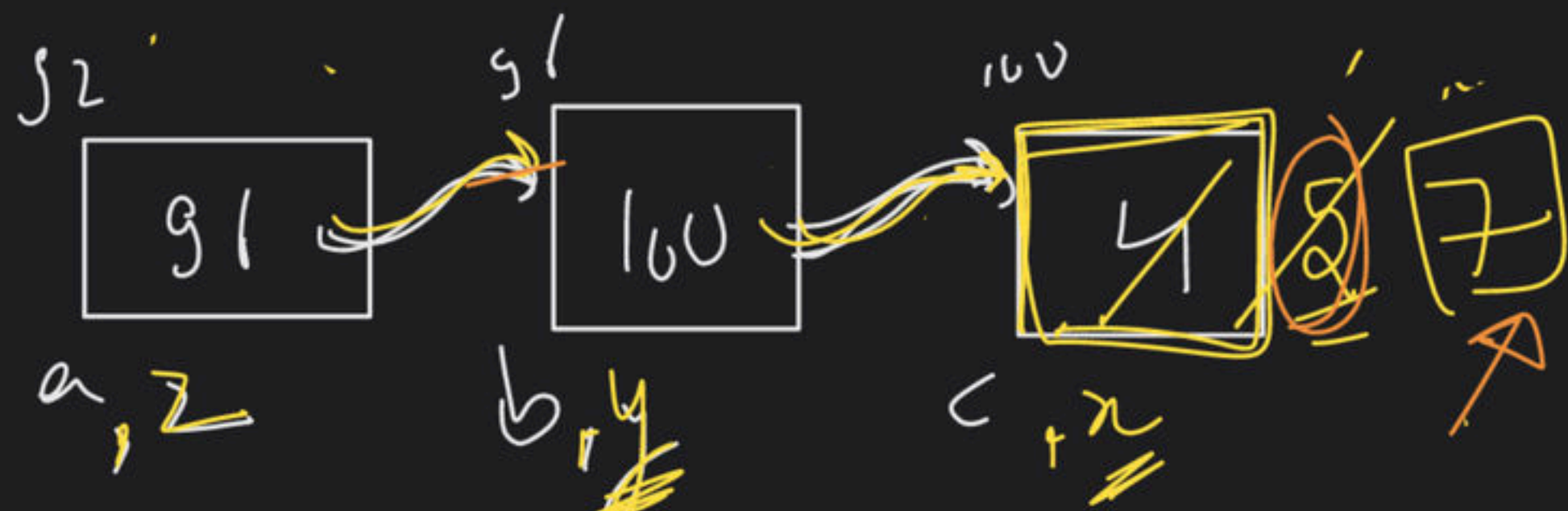
Count dp

2016

10

(10) →

$$\boxed{\star p} \times (\star p) \times (\star p) = 10 \times 10 = 100$$



DRK RUN

$$n = n + 3$$

$$= 4 + 3$$

$$= 7$$

$$**z = **z + 1$$

$$= 7 + 1 = 8$$

$$*y = *y + 1$$

$$= 1 + 2 = 3$$

```

int main()
{
    int c, *b, **a;
    c = 4;
    b = &c;
    a = &b;
    cout << f(c, b, a);
    return 0;
}

```

```

int f(int n, int *y, int **z)
{
    int p, q;
    **z += 1;
    q = **z;
    *y += 2;
    p = *y;
    n += 3;
    return n + p + q;
}

```

majority

4

z = 5

p = 7

n = 10

19


```
int main()
```

```
{ int xxx, xxg, p, i=0;
```

```
    p = i;
```

```
    (*p) ++;
```

```
    g = p;
```

```
    (xxg) ++;
```

```
    x = g;
```

```
    cout << *p << xxg << xxx;
```

```
}
```

100/1

(*p) ++
*p = *p + 1

xxg = xxg + 1

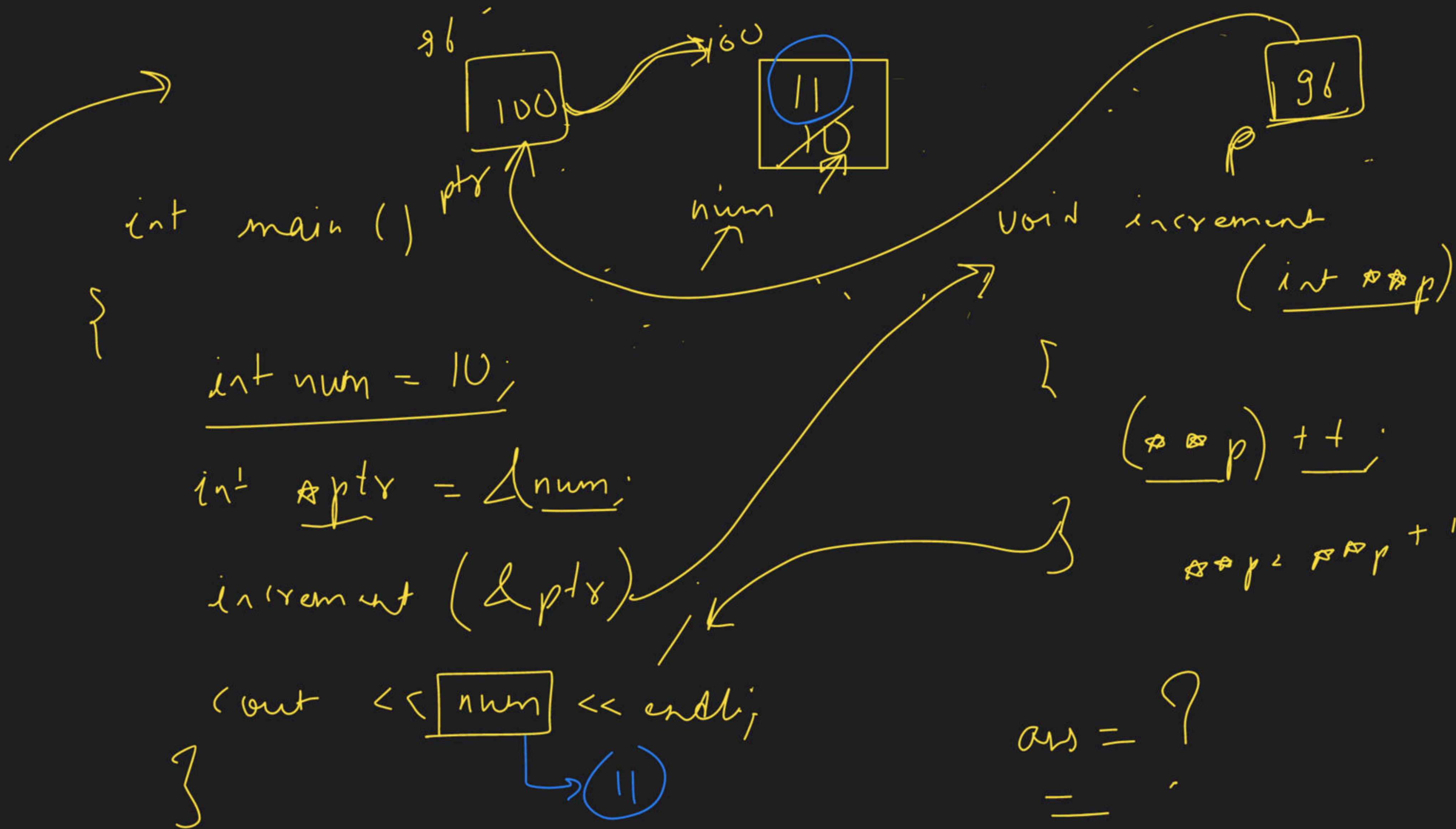


<< xxx;

(CBLINK)
Header++

0/p

?



ans = ?
=

→ Double pointer

(1-1) → 2.hvt

→ Assignment

Multiple

quest

Ramah Ultra pro max h1
++

→ * Array
→ * Use Arr

→ * funct

→ * *

→ * * *

Maza

ay

→ 98%

Let + Video

Repeat

→ Pointer

Don't

→ 1 level

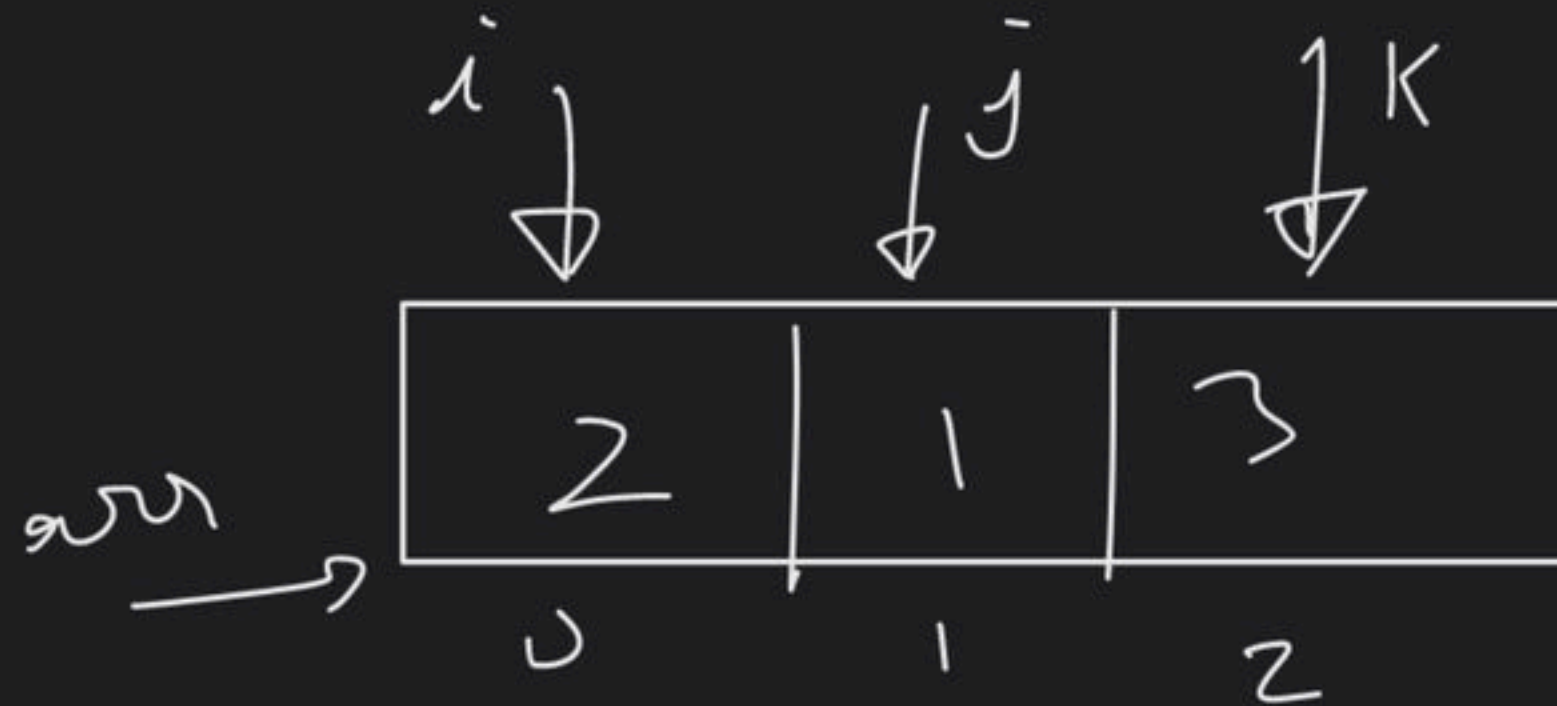
goyl

Discard

* * * * *

Sunday

→ Ultam:-



Optimize

approach

triple (x, y, z)

triple
↓
0

for ($i \rightarrow 0$

{ for ($j \rightarrow i+1$

{ for ($k \rightarrow j+1$

{ if $arr[i] + arr[j] + arr[k] == 0$
return

$(x, y, z) \rightarrow 0$

→ $O(n^3)$

6 0 0 → 0

1101

110
011
010

sum = sum + arr[i]
sum = sum + arr[i];

1/T: 17.1
mark

derived mark
pair
tree

If $a \Delta b \Rightarrow 0$

sqrt
ptr

yes or no

T.C (CP) 16m

h2
sum

0 4 →
0 < n →
cin >> h
scanf
Acc

$O(n^3) \rightarrow O(n^2)$

At

Sunday

h^2

Upan

4 5 6
5 7 8
6 5 4
8 4 5

course

?

Crash course

possible

or not

applied

18

4 | 5 | 6 | 7 | | |

DRY RUN

heap

count

Observation

single traversal

applied

