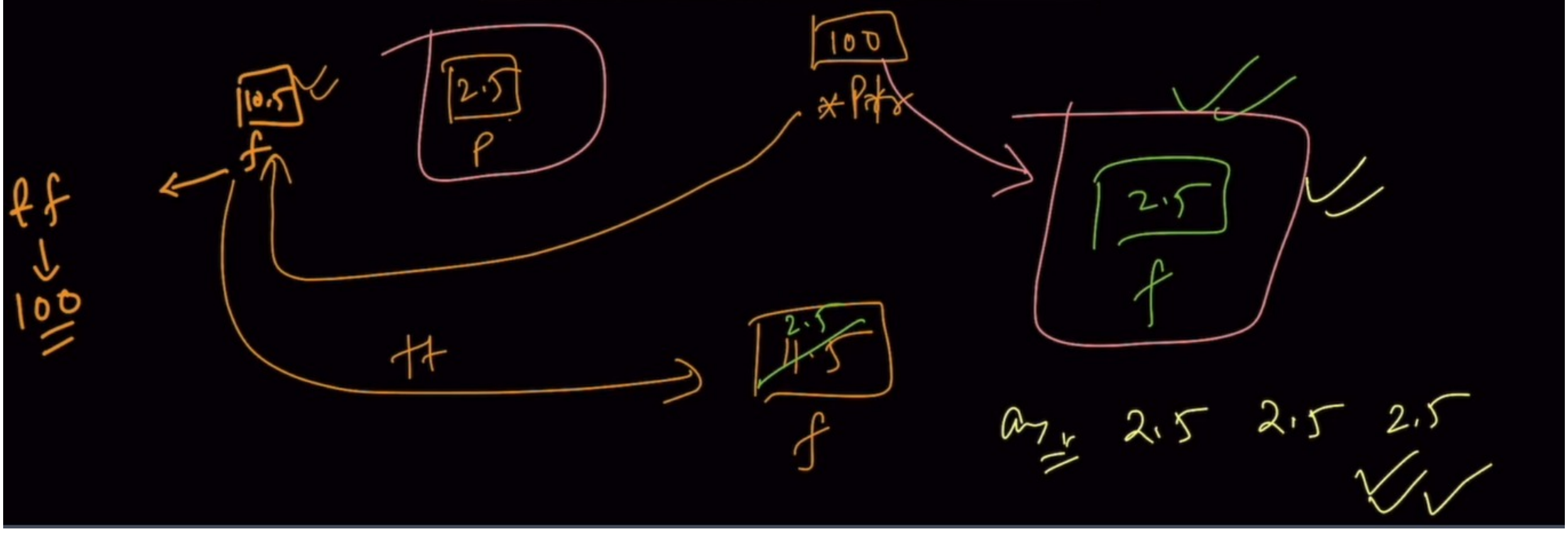
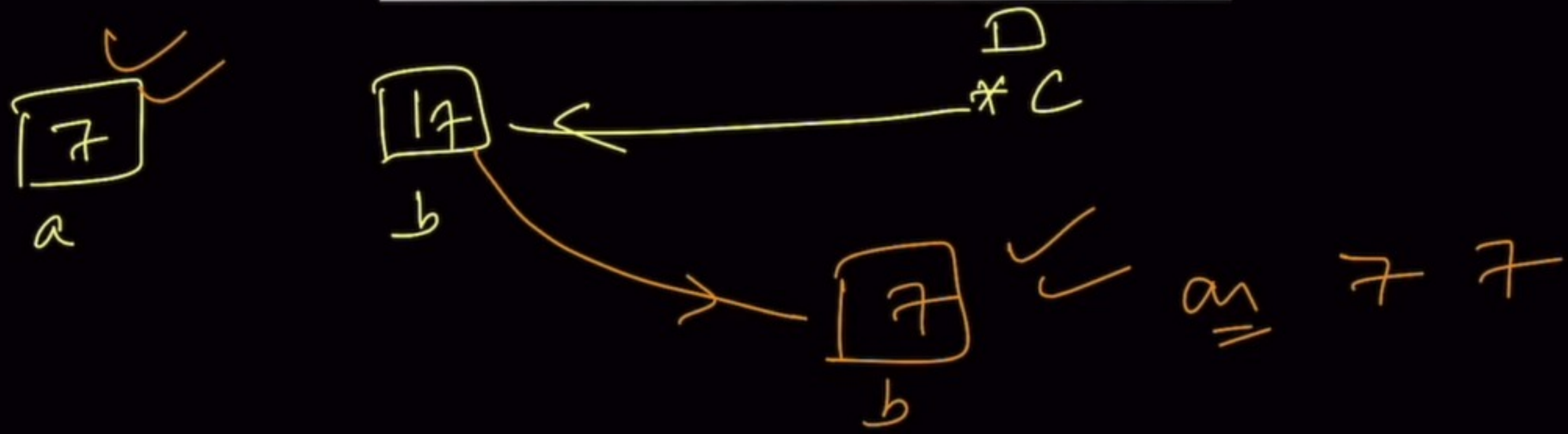
What will be the output ?

```
float f = 10.5;
float p = 2.5;
float* ptr = &f;
(*ptr)++;
*ptr = p;
cout << *ptr << " " << f << " " << p;
```

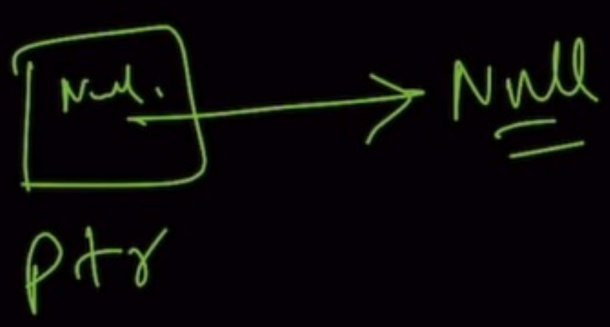



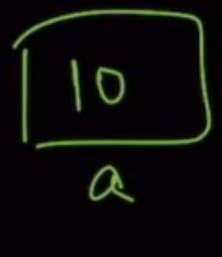
```
int a = 7;
int b = 17;
int *c = &b;
*c = 7;
cout  << a << "  " << b << endl;
```



```
int *ptr = 0;
int a = 10;
*ptr = a;
```

cout << *ptr << endl;





*ptr → Null pointer

*(0x0000)

learn.thecodehelp.in
anshawasthi3894@gmail.com
+918392823894

Which of the following gives the memory address of variable 'b' pointed by pointer 'a' i.e.

```
int b = 10;
int *a = &b;
```

a ✓✓

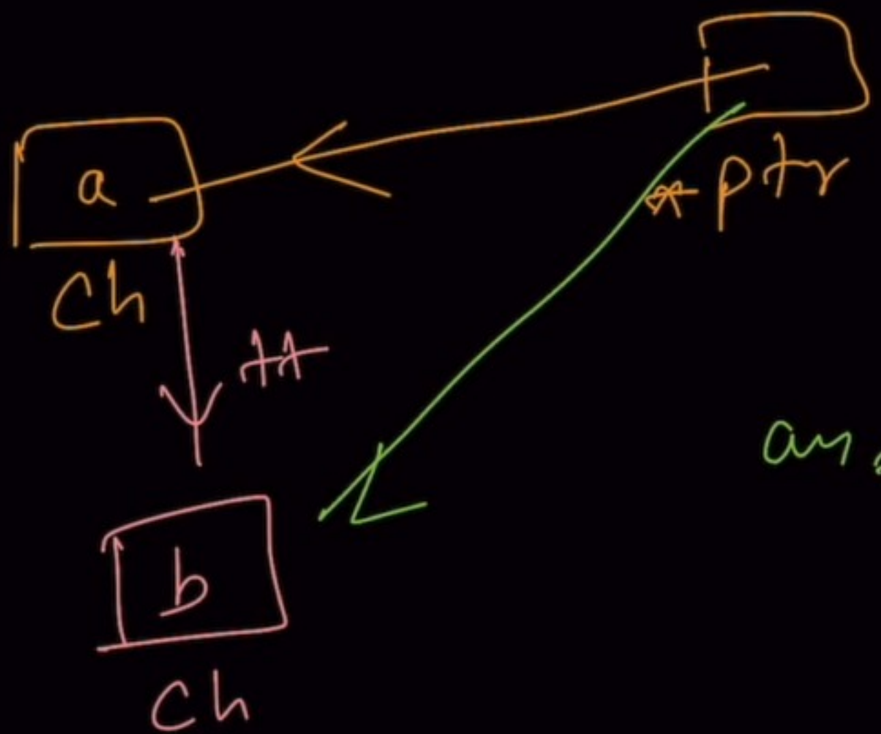
cout << a << endl

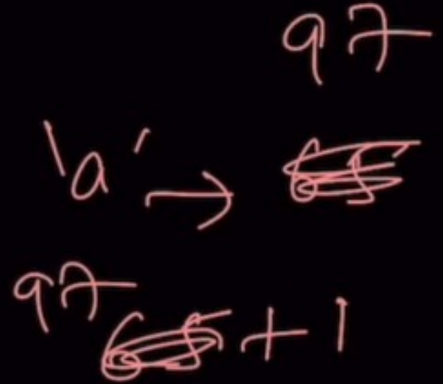
↓

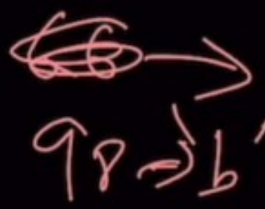
addrss of b

What will be the output ?

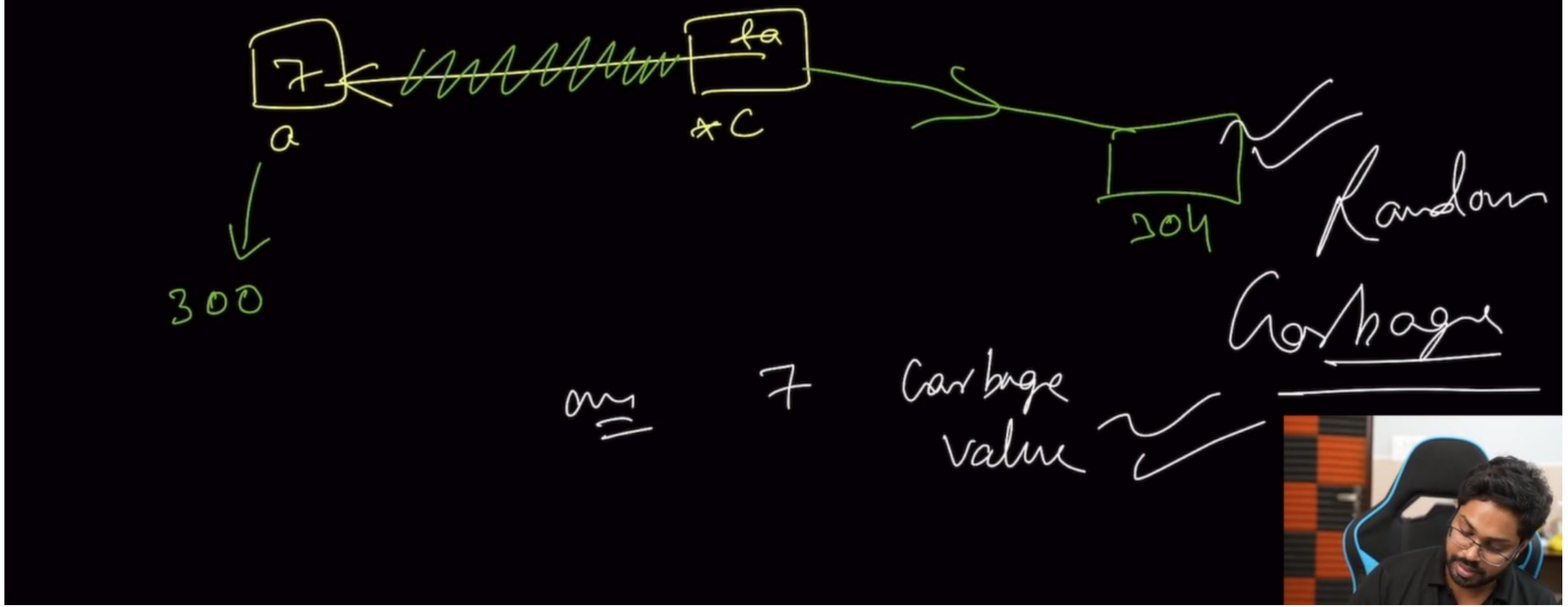
```
char ch = 'a';
char* ptr = &ch;
ch++;
cout << *ptr << endl;
```





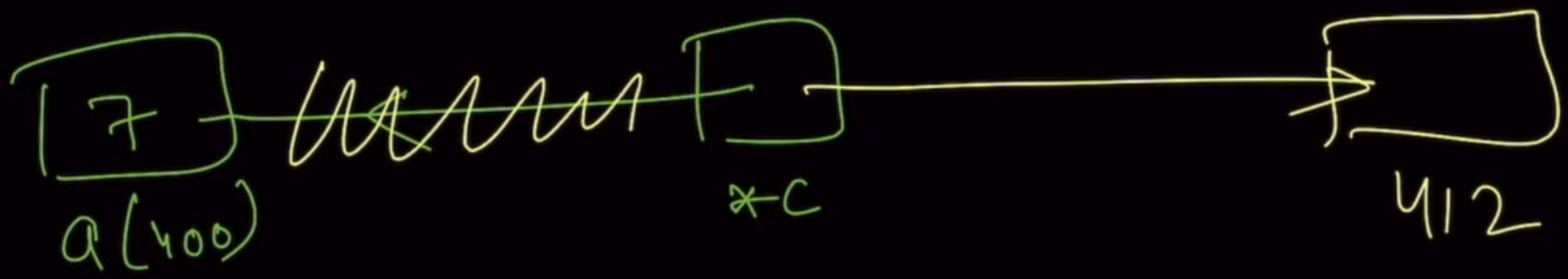


```cpp
int a = 7;
int *c = &a;
c = c + 1;
cout  << a << "  " << *c << endl;
```



Assume the memory address of variable 'a' is 400 (and an integer takes 4 bytes), what will be the output -

```
int a = 7;
int *c = &a;
c = c + 3;
cout<< c << endl;
```



Output
⇒ 412.

$c = 400$

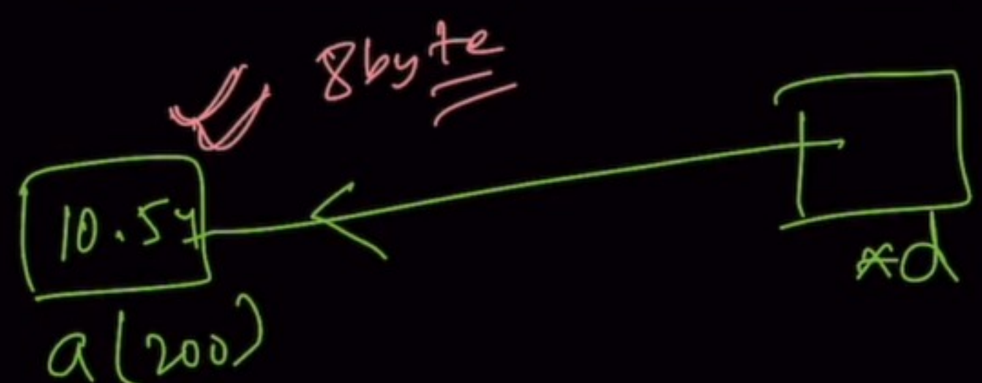
$c = 400 + 3 = 403$ ✗

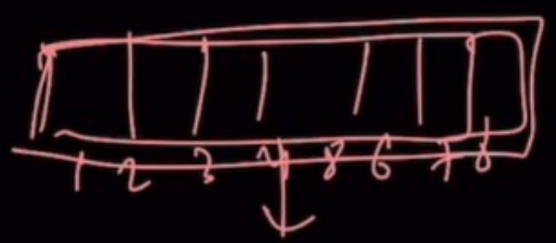
$c = 400 + (3 \times 4) = 400 + 12 = 412$ ✓

Assume memory address of variable 'a' is : 200 and a double variable is of size 8 bytes, what will be the output -

```
double a = 10.54;
double *d = &a;
d = d + 1;
cout  << d << endl;
```

208





d = 200.

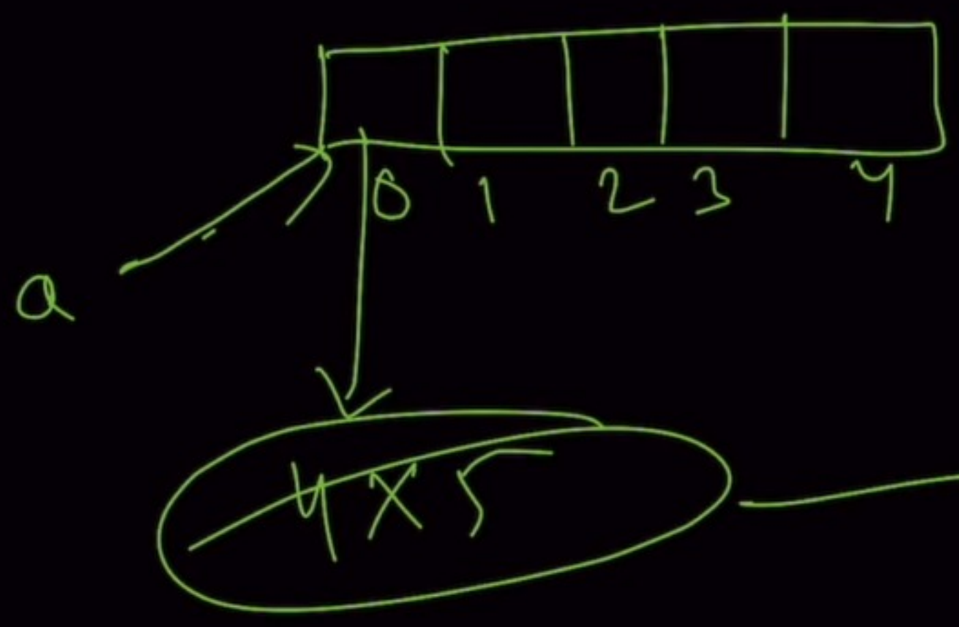
d = d + 1

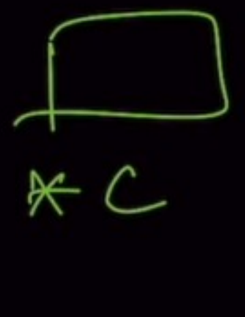
d = 200 + 1 (X) 120

d = 200 + (1 X 8) = 208.

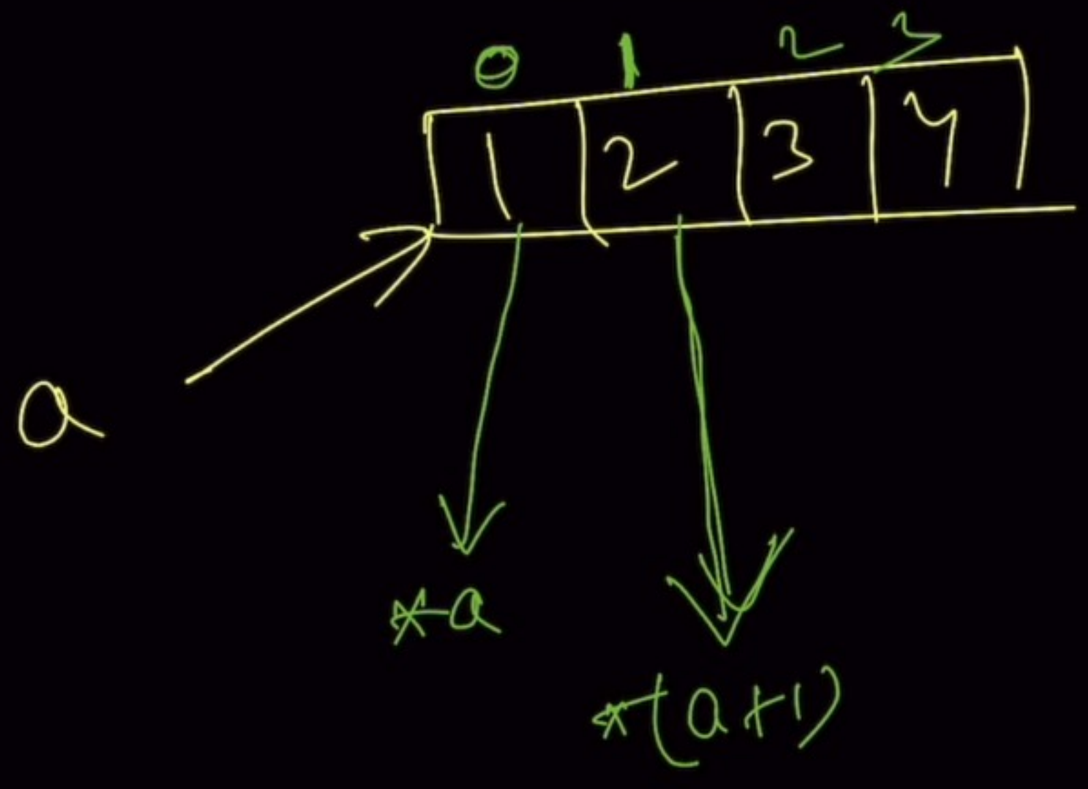
Assume integer takes 4 bytes and integer pointer 8 bytes.

```
int a[5];
int *c;
cout << sizeof(a) << " " << sizeof(c);
```



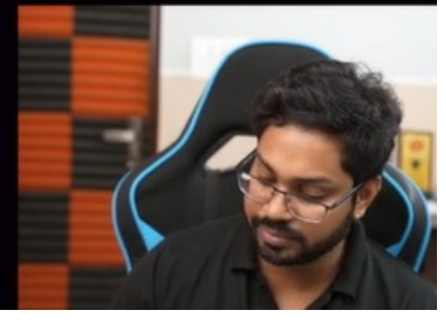


```
int a[] = {1, 2, 3, 4};
cout << *(a) << " " << *(a+1);
```



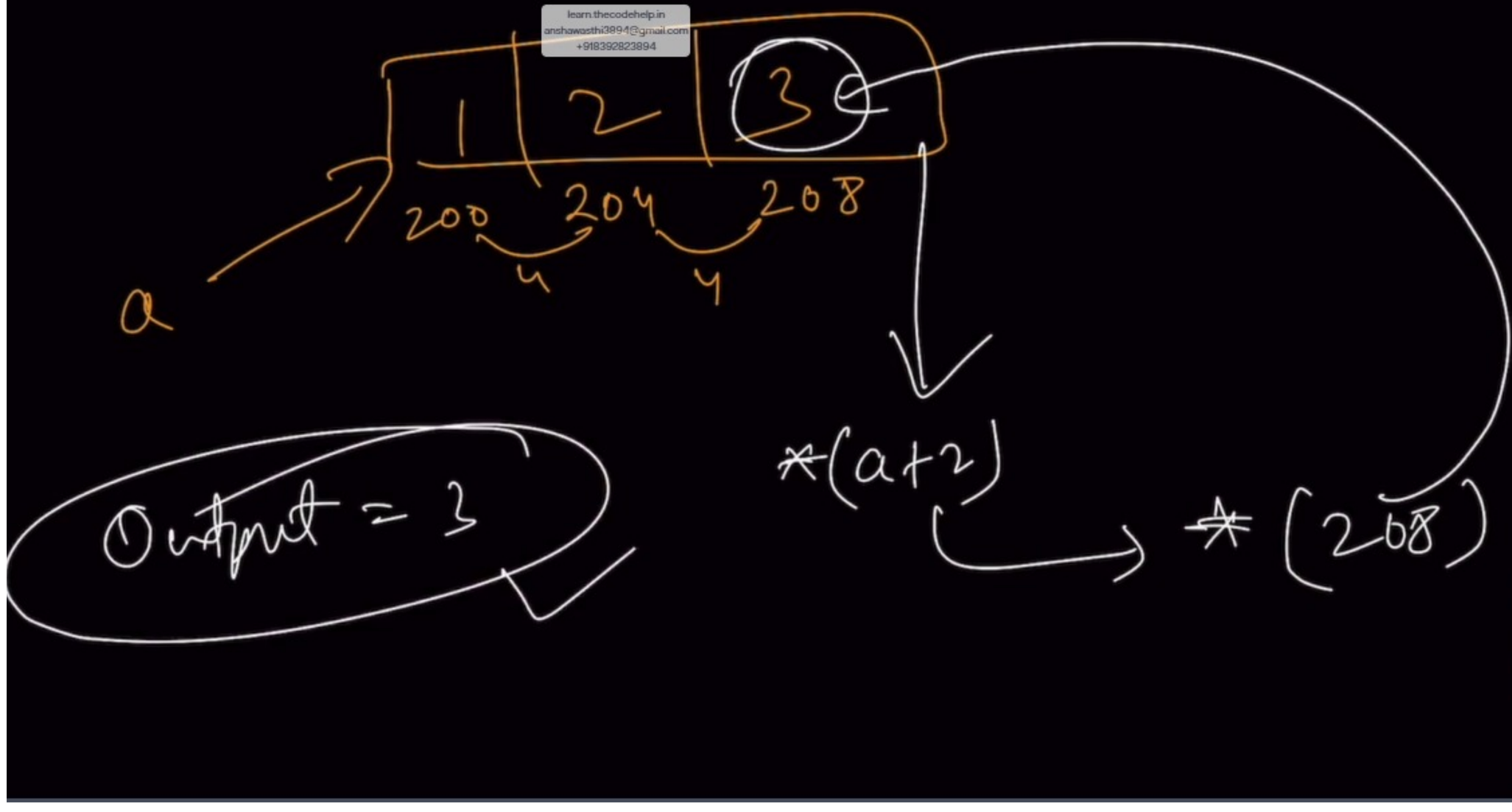
Output,

1 2



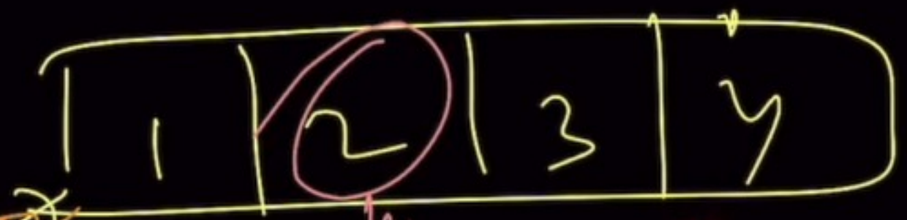
Assume that address of 0th index of array 'a' is : 200. What is the output -

```
int a[3] = {1, 2, 3};
cout << *(a + 2);
```

```
int a[] = {1, 2, 3, 4};
int *p = a++;
cout << *p << endl;
```

int a
int b
int a[3] —



a

Const pointer

int *p = (a++)

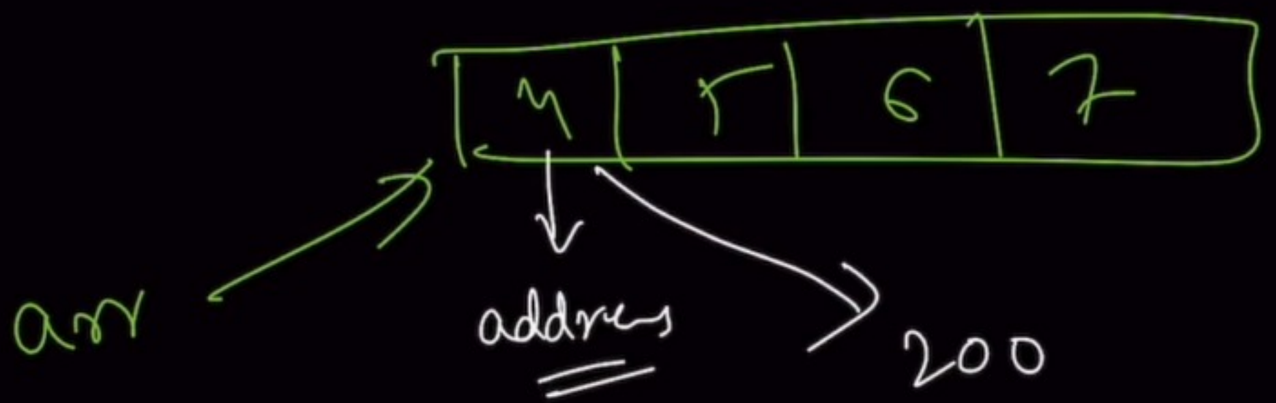
ERROR

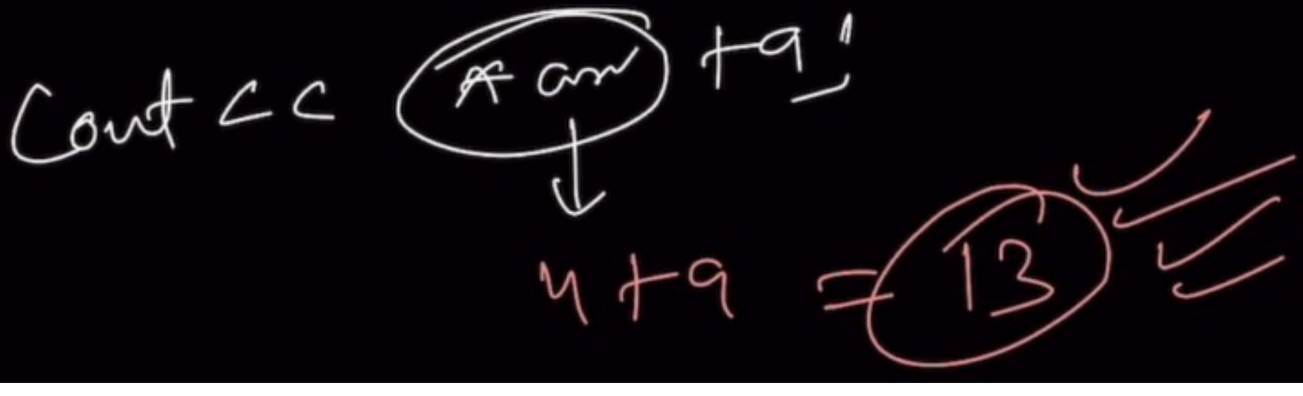
int *p = (a = a+1)

a = a+1;

a = 2

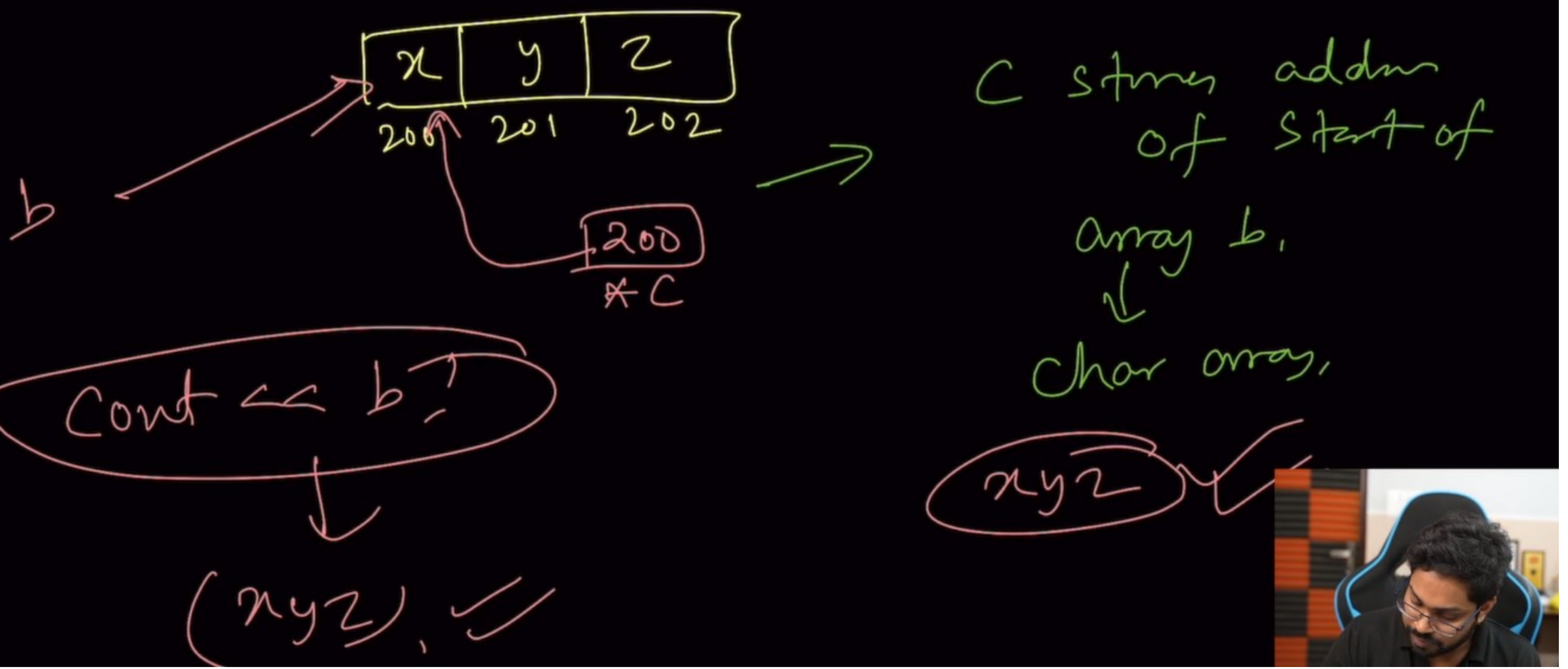
```cpp
#include <iostream>
using namespace std;
int main()
{
  int arr[] = {4, 5, 6, 7};
  int *p = (arr + 1);
  cout << *arr + 9;
  return 0;
}
```





Assume address of 0th index of array 'b' is 200. What is the output -

```
char b[] = "xyz";
char *c = &b[0];
cout << c << endl;
```



Assume address of 0th index of array 'b' is 200. What is the output -

```
char b[] = "xyz";
                ;
```

cout << b ;

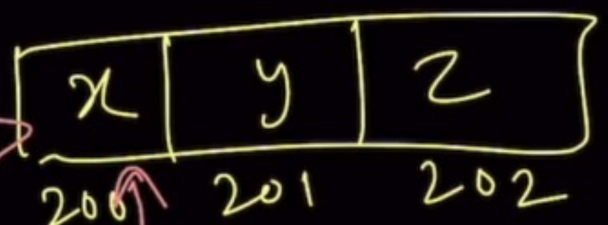
xyz

C stores addr of start of array b.

↓

char array.

xyz



b

200

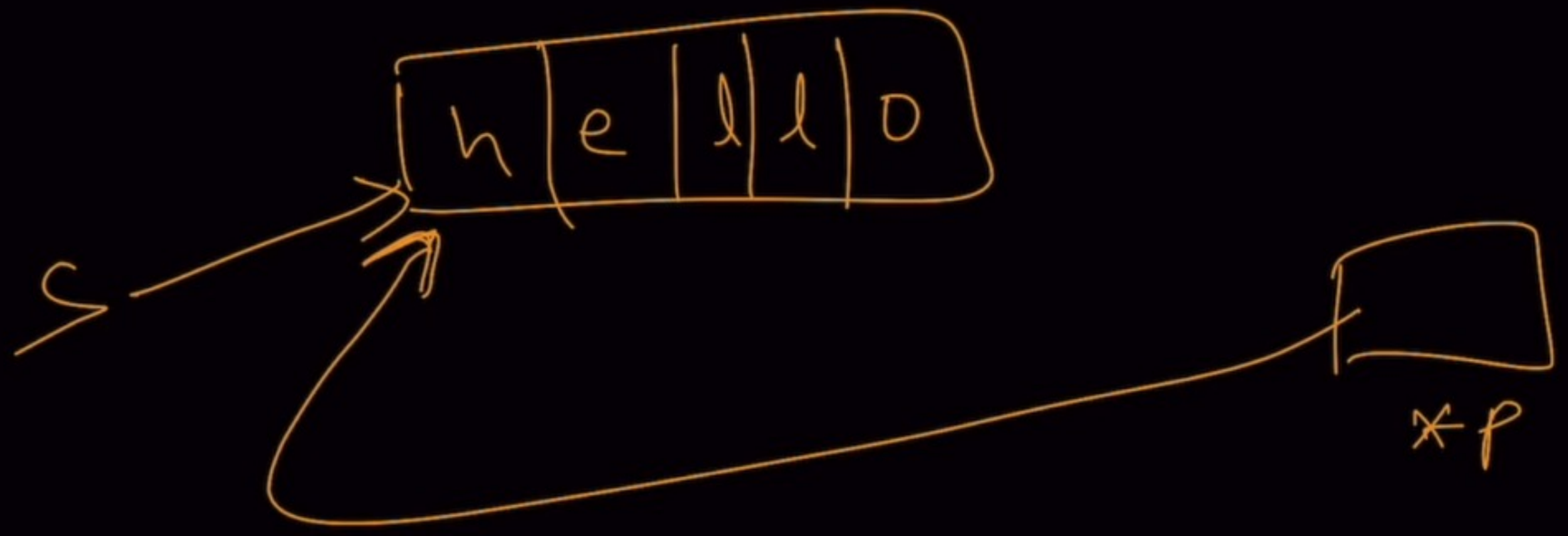
*C

cout << b ;

↓

(xyz)

```
char s[]= "hello";
char *p = s;
cout << s[0] << " " << p[0];
```



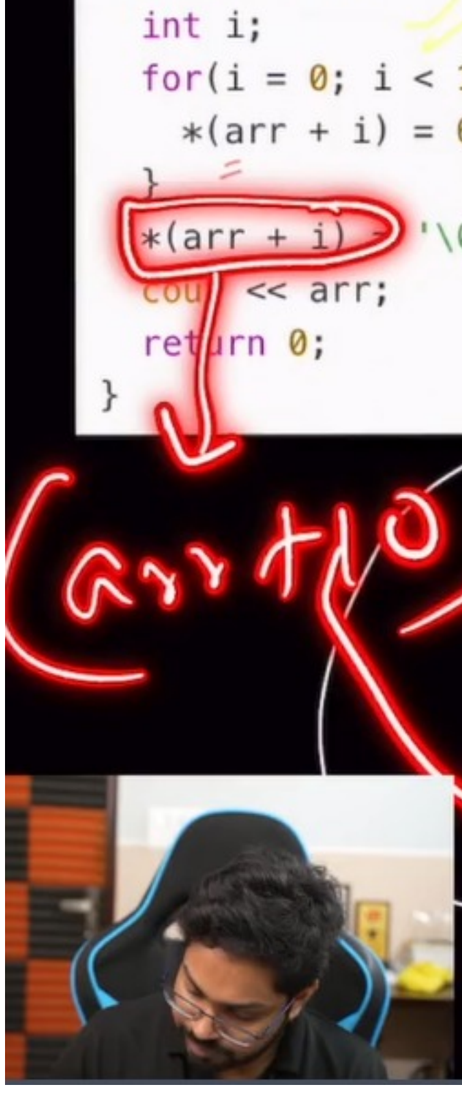
s[0] = h

p[0] = h

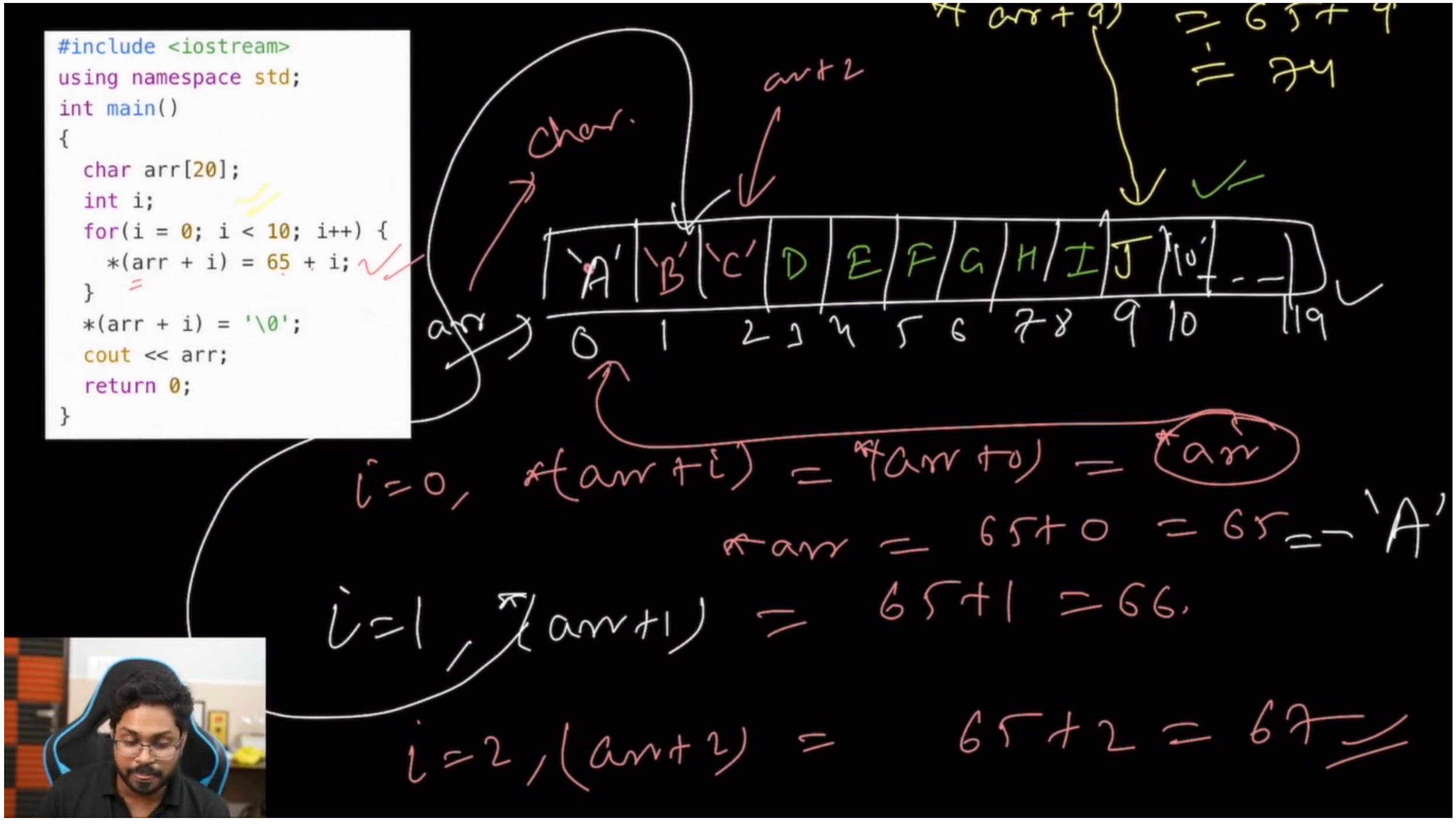
output :-

h h

```cpp
#include <iostream>
using namespace std;
int main()
{
  char arr[20];
  int i;
  for(i = 0; i < 10; i++) {
    *(arr + i) = 65 + i;
  }
  *(arr + i) = '\0';
  cout << arr;
  return 0;
}
```

```cpp
#include <iostream>
using namespace std;
int main()
{
  char arr[20];
  int i;
  for(i = 0; i < 10; i++) {
    *(arr + i) = 65 + i;
  }
  *(arr + i) = '\0';
  cout << arr;
  return 0;
}
```



```cpp
#include <iostream>
using namespace std;
int main()
{
  char arr[20];
  int i;
  for(i = 0; i < 10; i++) {
    *(arr + i) = 65 + i;
  }
  *(arr + i) = '\0';
  cout << arr;
  return 0;
}
```

arr+9

arr+2

char.

arr

A' B' C' D E F G H I J '\0'

0 1 2 3 4 5 6 7 8 9 10 19

i=0, *(arr+i) = *(arr+0) = arr

*arr = 65+0 = 65 = 'A'

i=1, *(arr+1) = 65+1 = 66.

i=2, (arr+2) = 65+2 = 67

```cpp
#include <iostream>
using namespace std;
int main()
{
  char arr[20];
  int i;
  for(i = 0; i < 10; i++) {
    *(arr + i) = 65 + i;
  }
  *(arr + i) = '\0';
  cout << arr;
  return 0;
}
```

arr+9

arr+2

char.

0	1	2	3	4	5	6	7	8	9	10	...	19
'A'	'B'	'C'	D	E	F	G	H	I	J	'\0'		

arr

i=0, *(arr+i) = *(arr+0) = arr

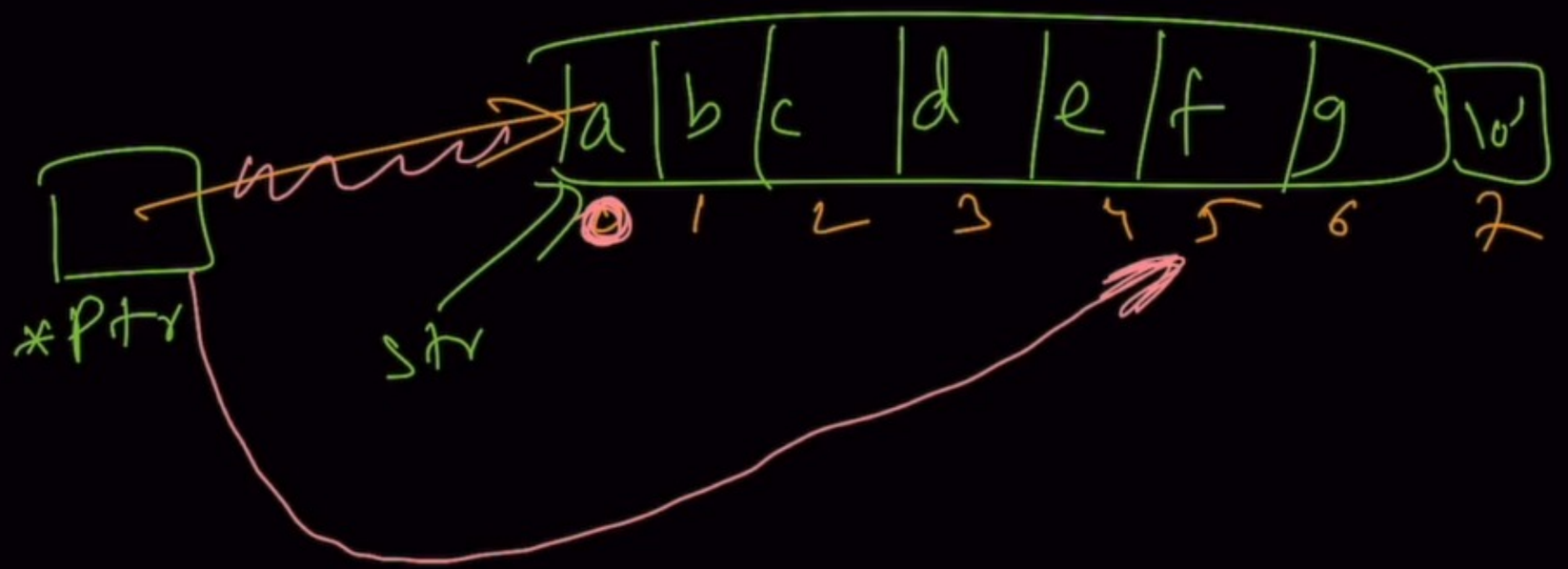
*arr = 65+0 = 65 = 'A'

i=1, *(arr+1) = 65+1 = 66.

i=2, (arr+2) = 65+2 = 67

Output
ABCDEFGHIJ

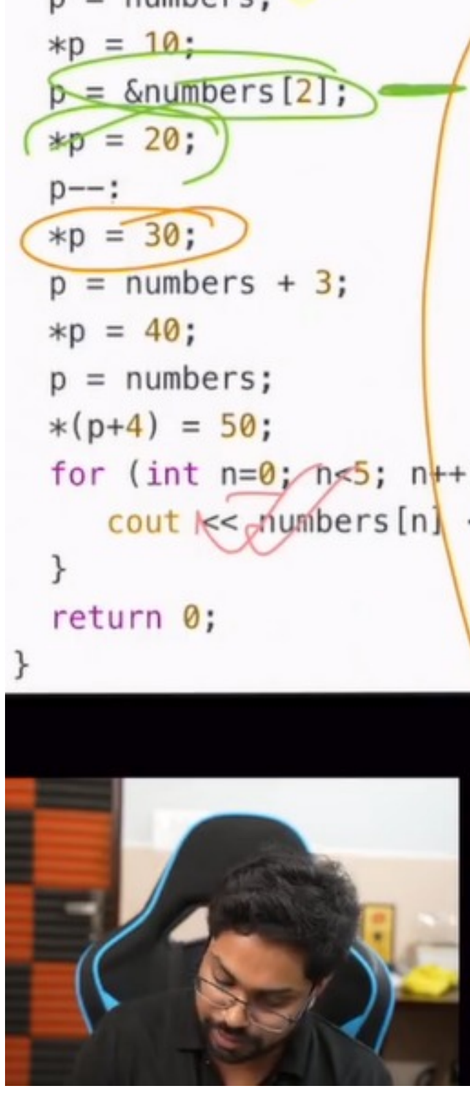
```cpp
#include <iostream>
using namespace std;
int main()
{
  char *ptr;
  char Str[] = "abcdefg";
  ptr = Str;
  ptr += 5;
  cout << ptr;
  return 0;
}
```

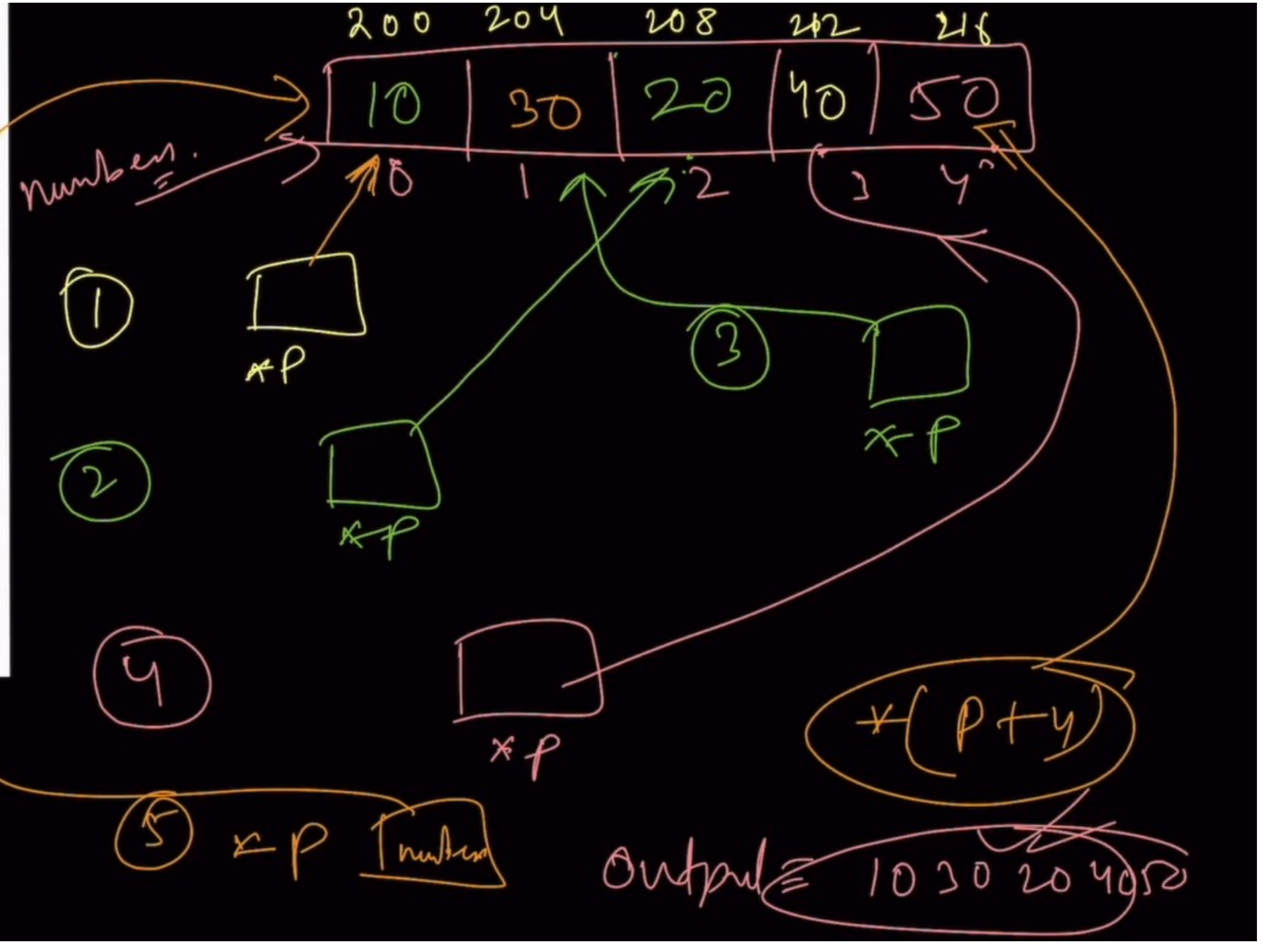


Cout << ptr;

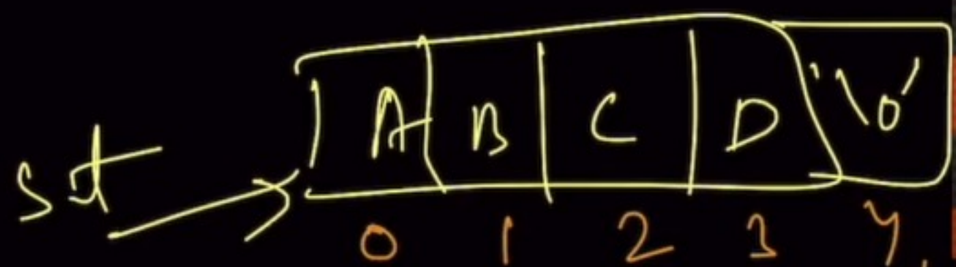
⇒ Output fg

```cpp
#include <iostream>
using namespace std;
int main ()
{
  int numbers[5];
  int * p;
  p = numbers;
  *p = 10;
  p = &numbers[2];
  *p = 20;
  p--;
  *p = 30;
  p = numbers + 3;
  *p = 40;
  p = numbers;
  *(p+4) = 50;
  for (int n=0; n<5; n++) {
    cout << numbers[n] << ",";
  }
  return 0;
}
```





```cpp
#include<iostream>
using namespace std;
int main() {
  char st[] = "ABCD";
  for(int i = 0; st[i] != '\0'; i++) {
     cout << st[i] << *(st)+i << *(i+st) << i[st];
  }
  return 0;
}
```



① i=0, st[0] ⇒ 'A', *(st)+i

'A'+0

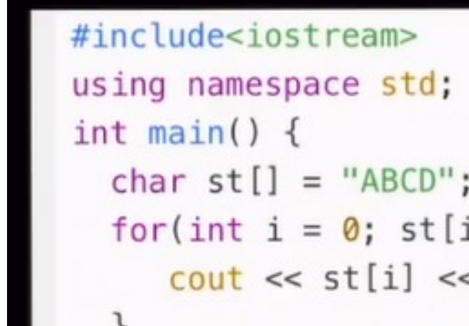
65

*(0+st) ⇒ *(st) ⇒ A

i[st] ⇒ 0[st] ⇒ st[0] ⇒ 'A'

A 65 A A

```cpp
#include<iostream>
using namespace std;
int main() {
  char st[] = "ABCD";
  for(int i = 0; st[i] != '\0'; i++) {
     cout << st[i] << *(st)+i << *(i+st) << i[st];
  }
  return 0;
}
```



i = 1, st[1] ⇓ B

*(st)+i ⇓ 'A'+1 ⇓ 65+1 ⇓ 66

*(1+st) ⇓ B

1[st] ⇓ st[1] ⇓ B

A66BB

i = 3 ⇒ st[3] ⇒ D

(*st)+3 'A'+3 = 68

*(3+st) ⇓ D

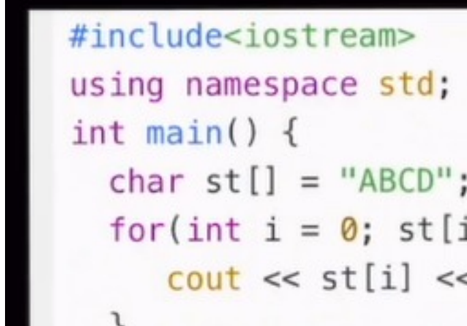
3[st] st[3] ⇓ D

st → A B C D '\0' (0 1 2 3 4)

A65AA
B66BB
C67CC
D68DD

A65AA

```cpp
#include<iostream>
using namespace std;
int main() {
  char st[] = "ABCD";
  for(int i = 0; st[i] != '\0'; i++) {
     cout << st[i] << *(st)+i << *(i+st) << i[st];
  }
  return 0;
}
```



i=1, st[1] ⇓ B

*(st)+i ⇓ 'A'+1 ⇓ 65+1 ⇓ 66

*(1+st) ⇓ B

1[st] ⇓ st[1] ⇓ B

A66BB

i = 3 ⇒ st[3] ⇒ D

*(st)+3 'A'+3 = 68

*(3+st) ⇓ D

3[st] st[3] ⇓ D

st → A B C D '\0' (0 1 2 3 4)

A65AA

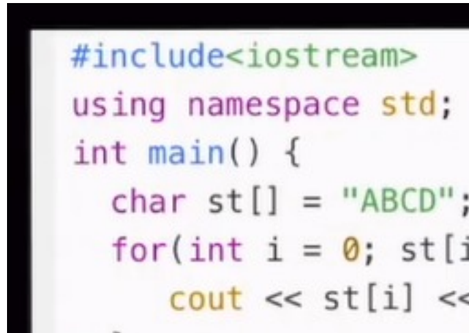
B66BB

C67CC

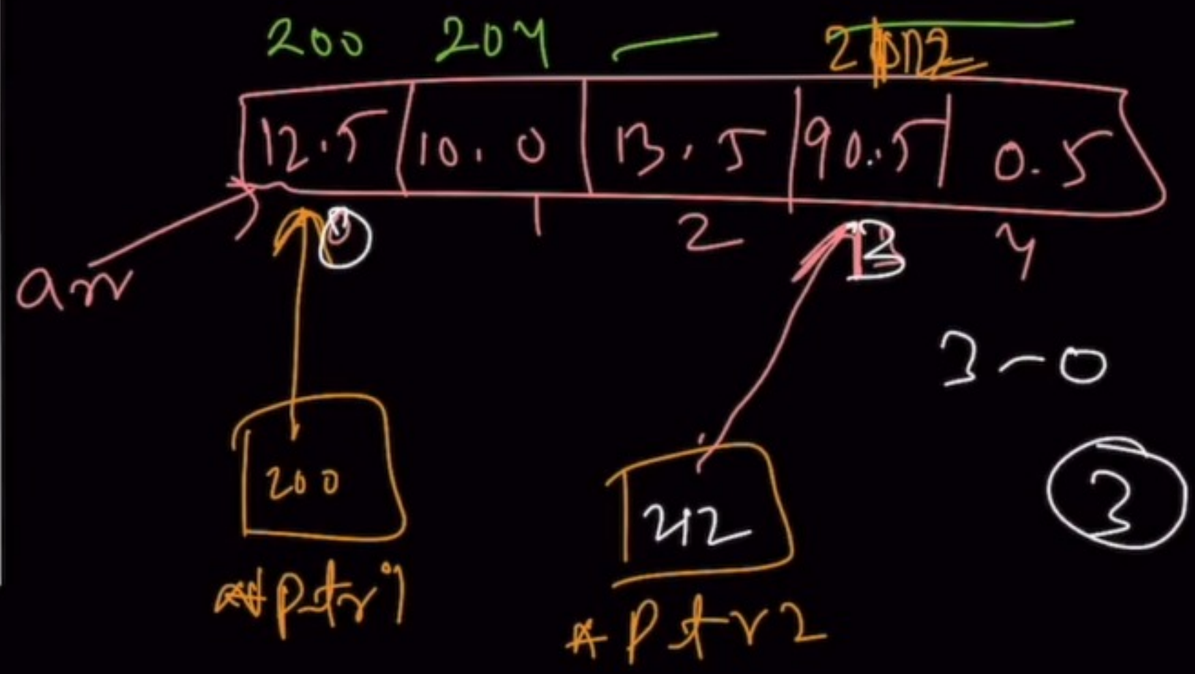
D68DD

A65AA

```cpp
#include<iostream>
using namespace std;
int main() {
  char st[] = "ABCD";
  for(int i = 0; st[i] != '\0'; i++) {
     cout << st[i] << *(st)+i << *(i+st) << i[st];
  }
  return 0;
}
```



```cpp
#include <iostream>
using namespace std;
int main()
{
  float arr[5] = {12.5, 10.0, 13.5, 90.5, 0.5};
  float *ptr1 = &arr[0];
  float *ptr2 = ptr1 + 3;
  cout<<*ptr2<<" ";
  cout<< ptr2 - ptr1;
  return 0;
}
```

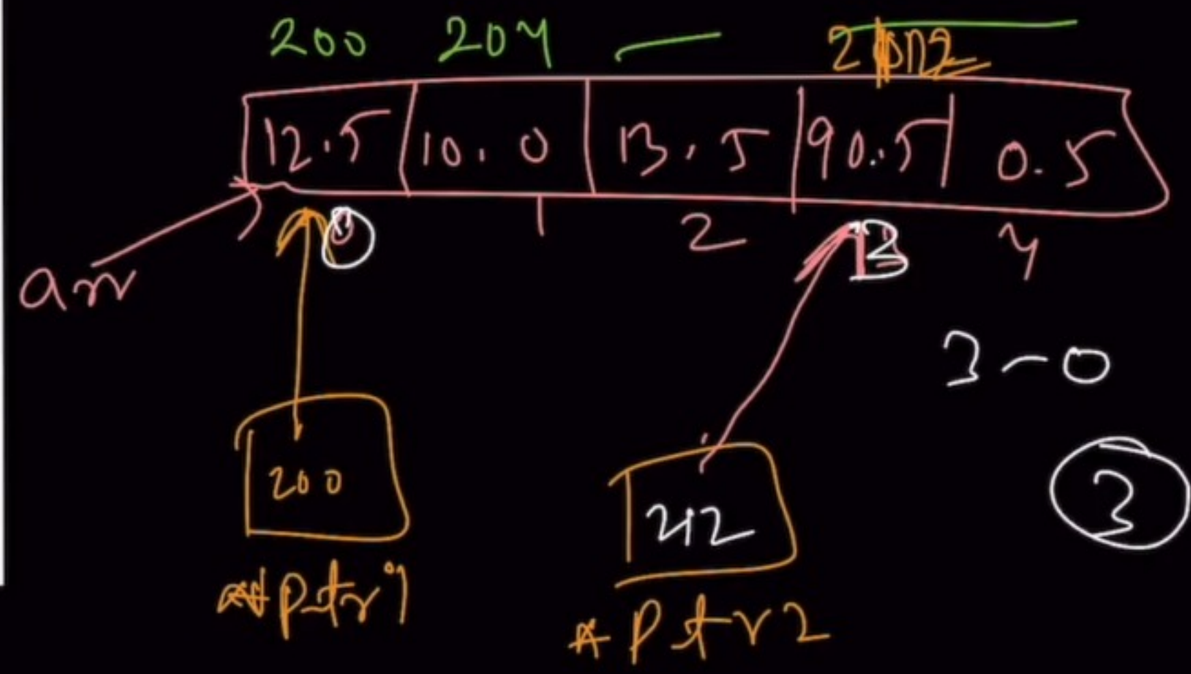


Output :- 90.5 3

$(212) = (200)$

$\frac{12}{4} = 3$

```cpp
#include <iostream>
using namespace std;
int main()
{
  float arr[5] = {12.5, 10.0, 13.5, 90.5, 0.5};
  float *ptr1 = &arr[0];
  float *ptr2 = ptr1 + 3;
  cout<<*ptr2<<" ";
  cout << ptr2 - ptr1;
  return 0;
}
```



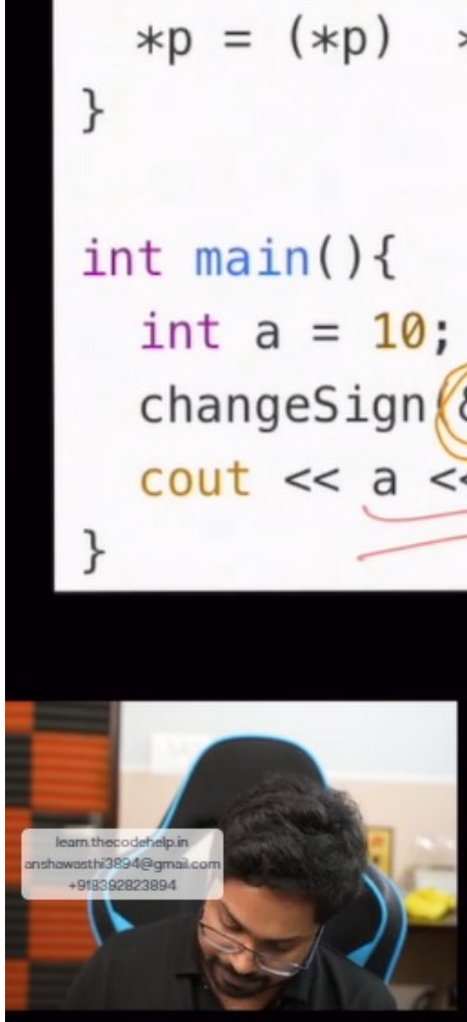
Output :- 90.5 3

(212) = (200)

$\frac{12}{4} = 3$

```cpp
void changeSign(int *p){
  *p = (*p)  *  -1;
}

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



changeSign (int *p)

{

10

a

*p

*p = *p * -1

= a x(-1)

}

10 x (-1)

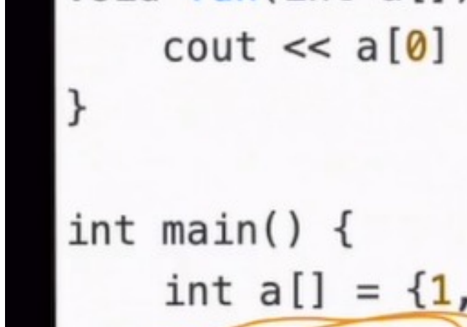
-10

-10

a

```
void fun(int a[]) {
    cout << a[0] << " ";
}

int main() {
    int a[] = {1, 2, 3, 4};
    fun(a + 1);
    cout << a[0];
}
```

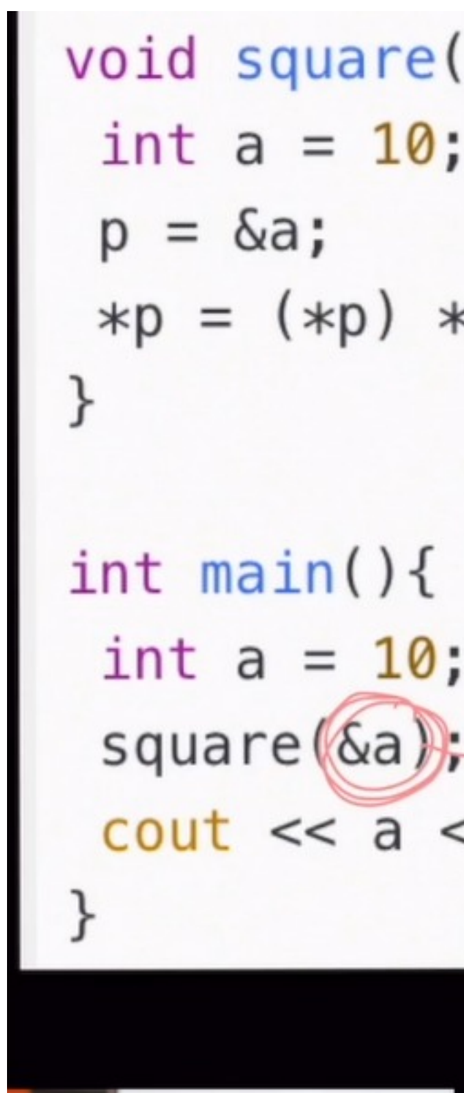


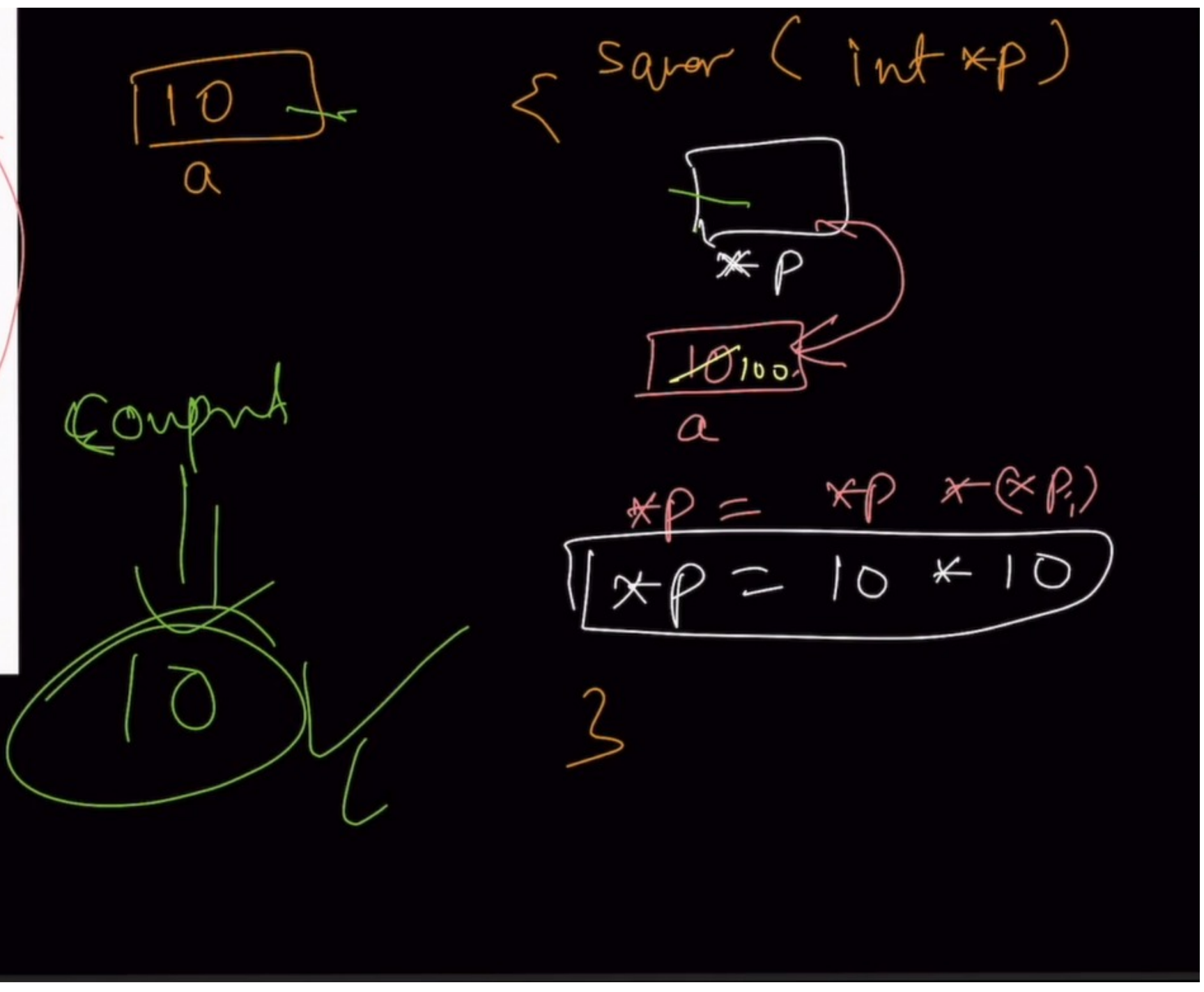
2 1

Output = 2 1

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

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

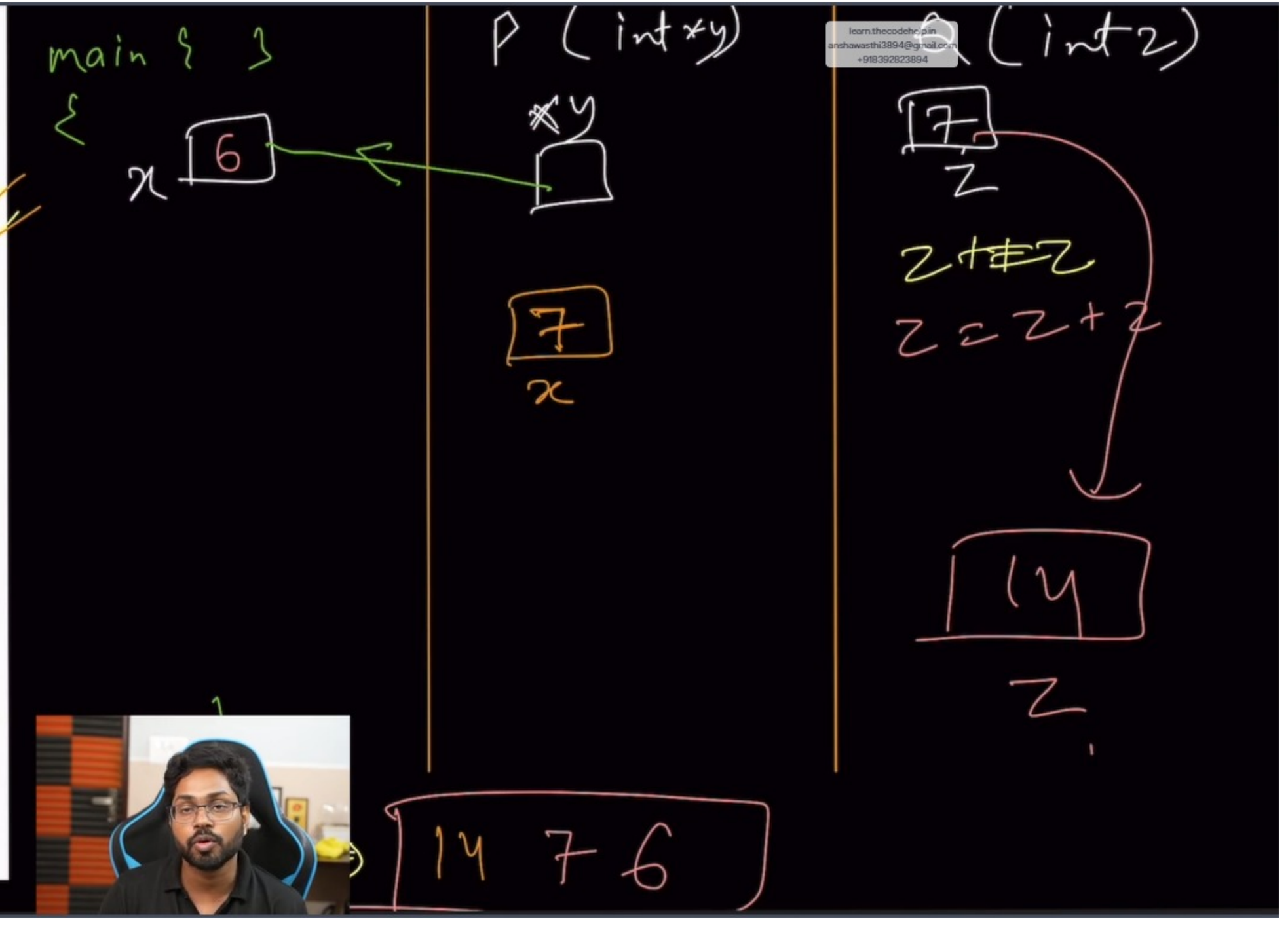




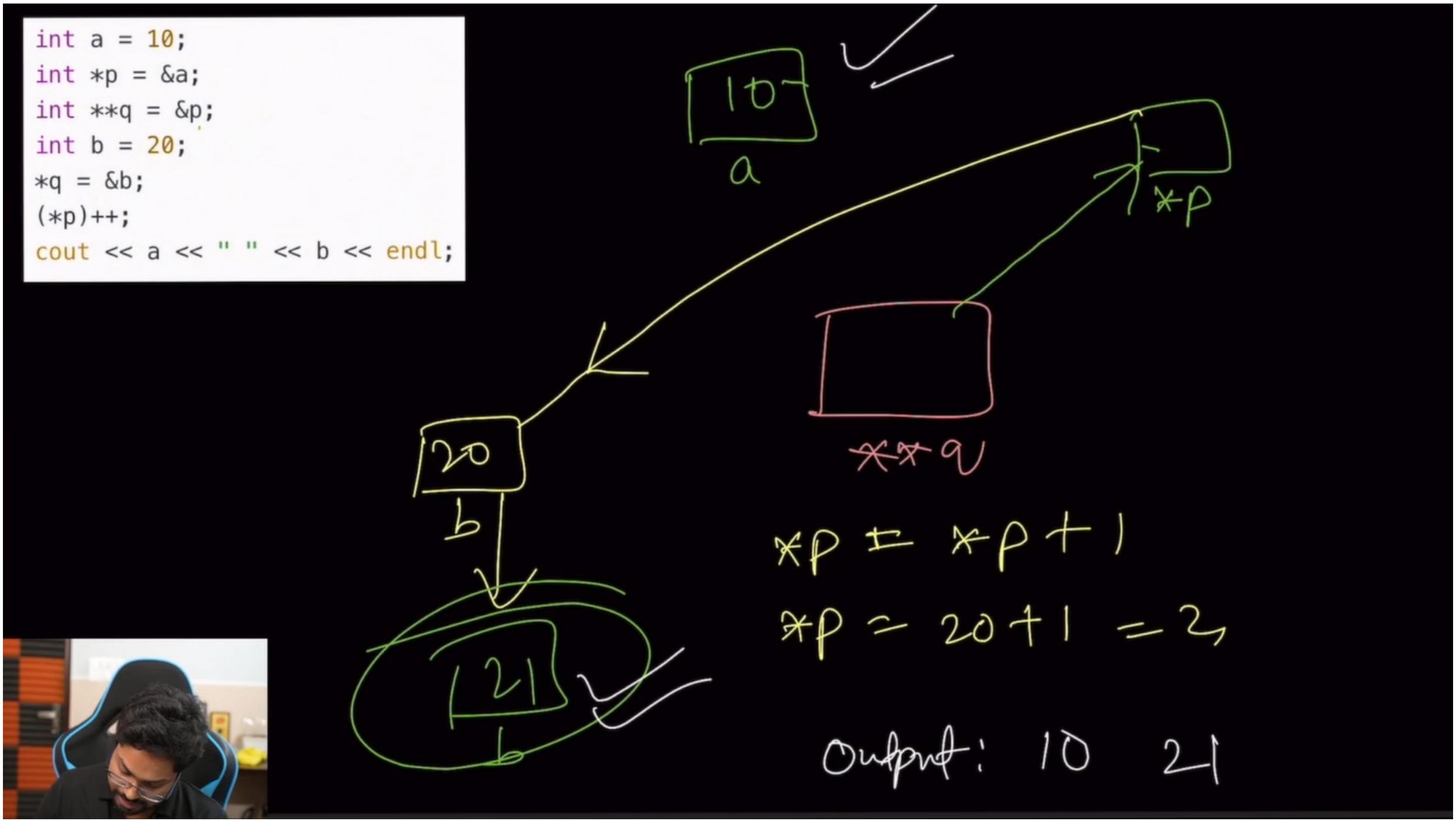
```cpp
#include <iostream>
using namespace std;
void Q(int z)
{
  z += z;
  cout<<z << " ";
}

void P(int *y)
{
  int x = *y + 2;
  Q(x);
  *y = x - 1;
  cout<<x << " ";
}

int main()
{
  int x = 5;
  P(&x);
  cout<<x;
  return 0;
}
```

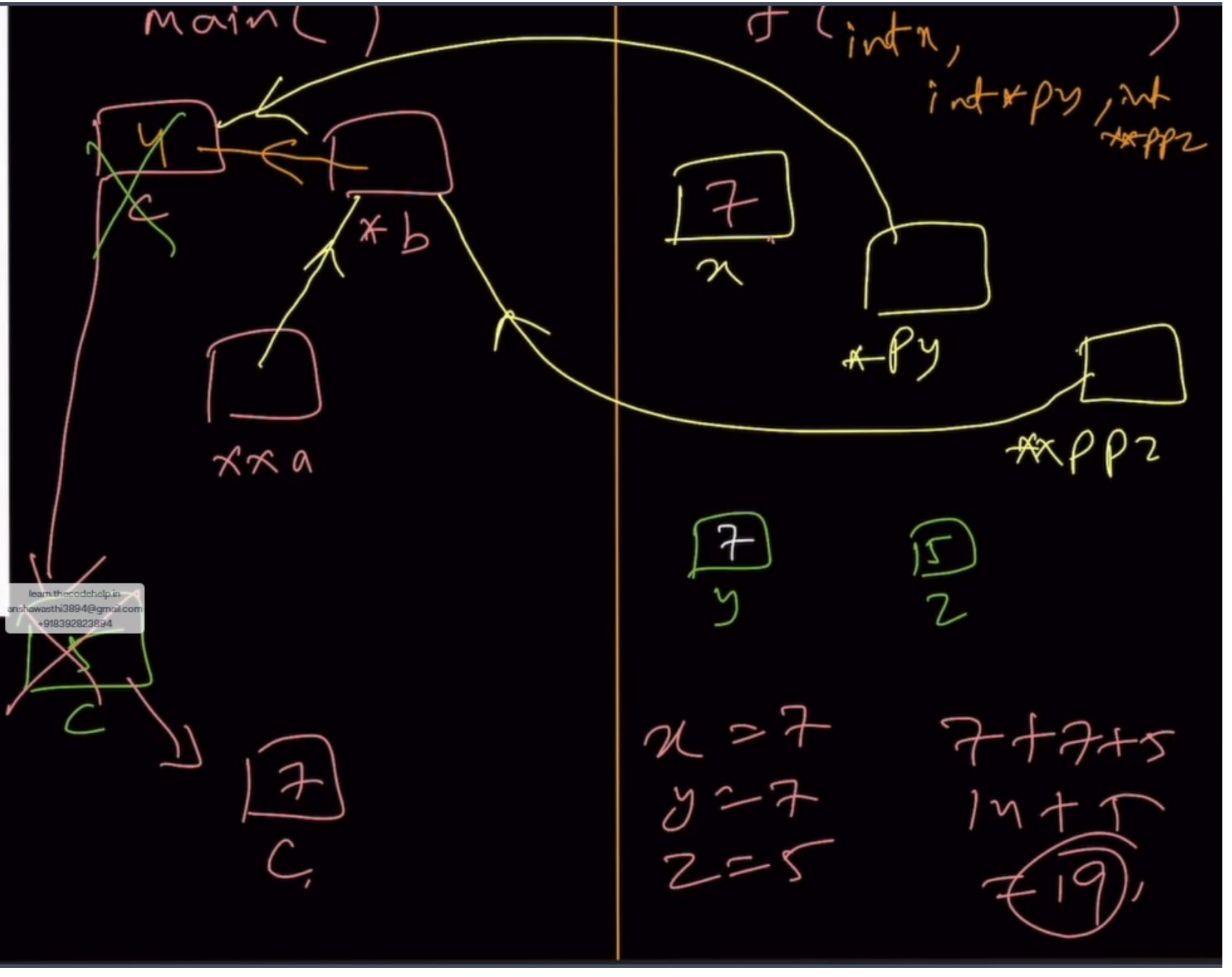


```
int a = 10;
int *p = &a;
int **q = &p;
int b = 20;
*q = &b;
(*p)++;
cout << a << " " << b << endl;
```

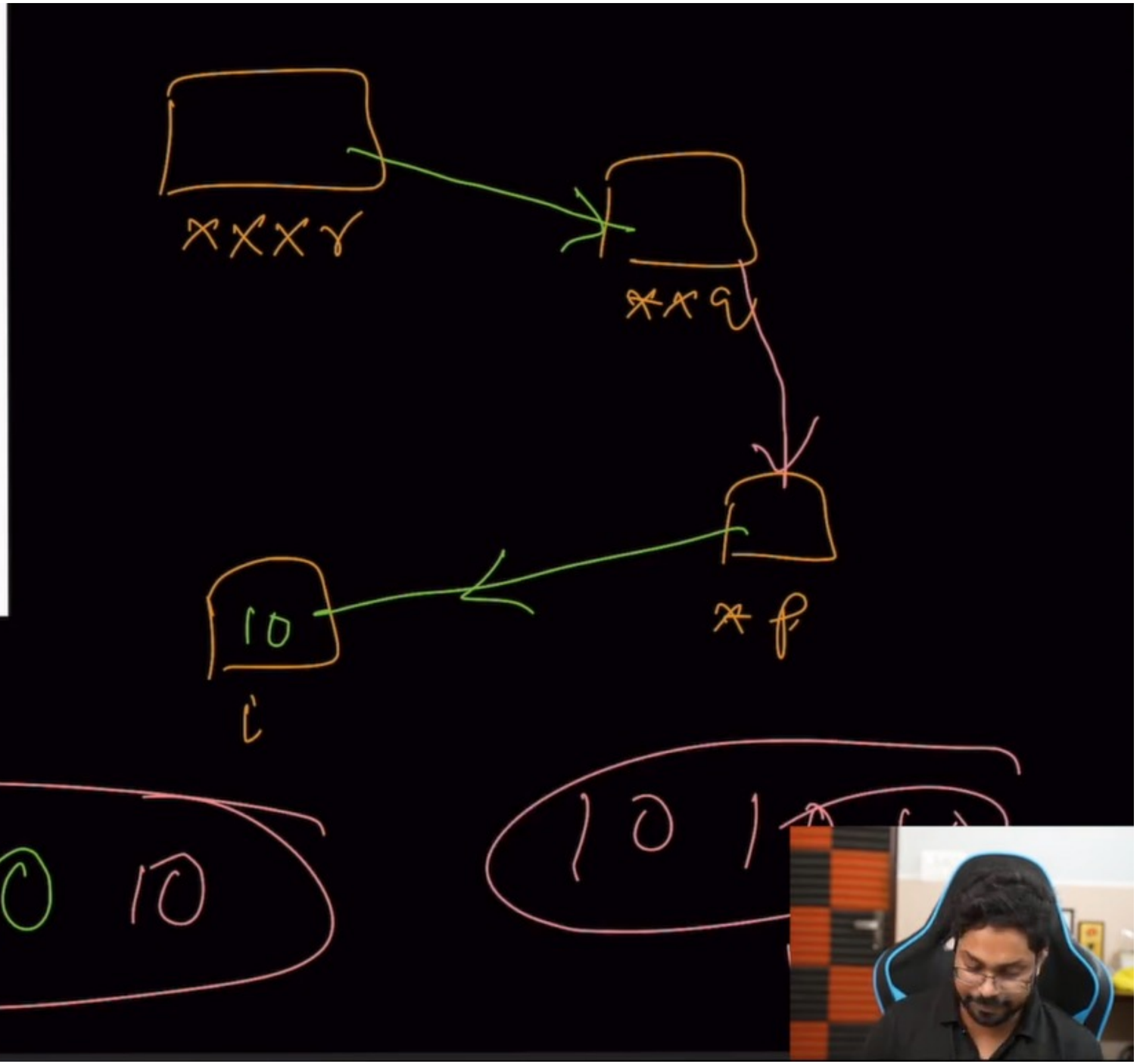


```
int f(int x, int *py, int **ppz) {
    int y, z;
    **ppz += 1;
    z = **ppz;
    *py += 2;
    y = *py;
    x += 3;
    return x + y + z;
}

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

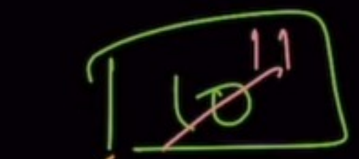


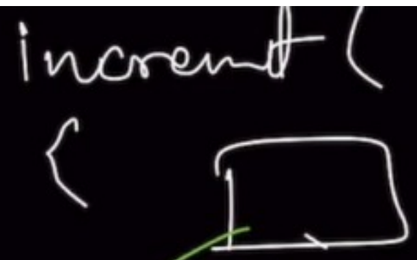
```cpp
#include<iostream>
using namespace std;
int main()
{
  int ***r, **q, *p, i=8;
  p = &i;
  (*p)++;
  q = &p;
  (**q)++;
  r = &q;
  cout<<*p << " " <<**q << " "<<***r;
  return 0;
}
```

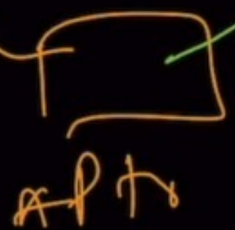


```cpp
void increment(int **p){
  (**p)++;
}

int main(){
 int num = 10;
 int *ptr = &num;
 increment(&ptr);
 cout << num << endl;
}
```







3

cout << num

11