

CS & IT ENGINEERING



Programming in C
Chapter -1
Data types and Operators
Lec- 03



By- Pankaj Sharma sir

TOPICS TO BE COVERED

Data type

why C is
platform
dependent

1 hr
constant/variable
chat

$$f(x) = 2$$

↳ constant
function

Library
Digital
IEEE
Java
.Net

Interface

ER already existing
Code को use कर
पर रहे हों।

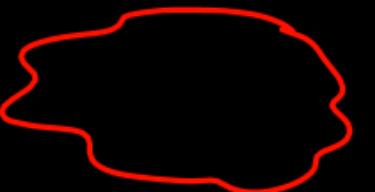
```
#include<stdio.h>
1. void main(){
    signed short int a;
}

```



2 byte

-32768 to
+32767

```
2.
#include<stdio.h>
void main(){
     short int i;
     signed
}

```

declaration

Signed short int i;

short int i ;

short i ; ✓

Signed short i ;

short

Same

Unsigned short int a ;

Unsigned short a ;

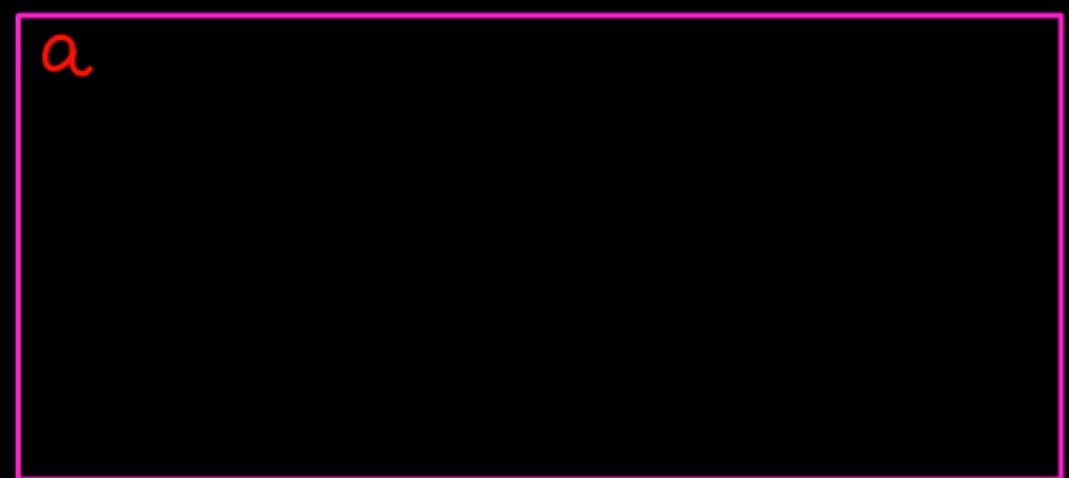
2 byte

```
#include<stdio.h>
void main(){
    short int a = 20;
    printf("a");
}
```



Integer
short int
int
long int
long long int

" " As it is
O/P



```
#include<stdio.h>
```

```
void main(){
```

```
printf("10+10");
```

```
}
```

O/P

10+10

printf → text format

format

Format Specifier

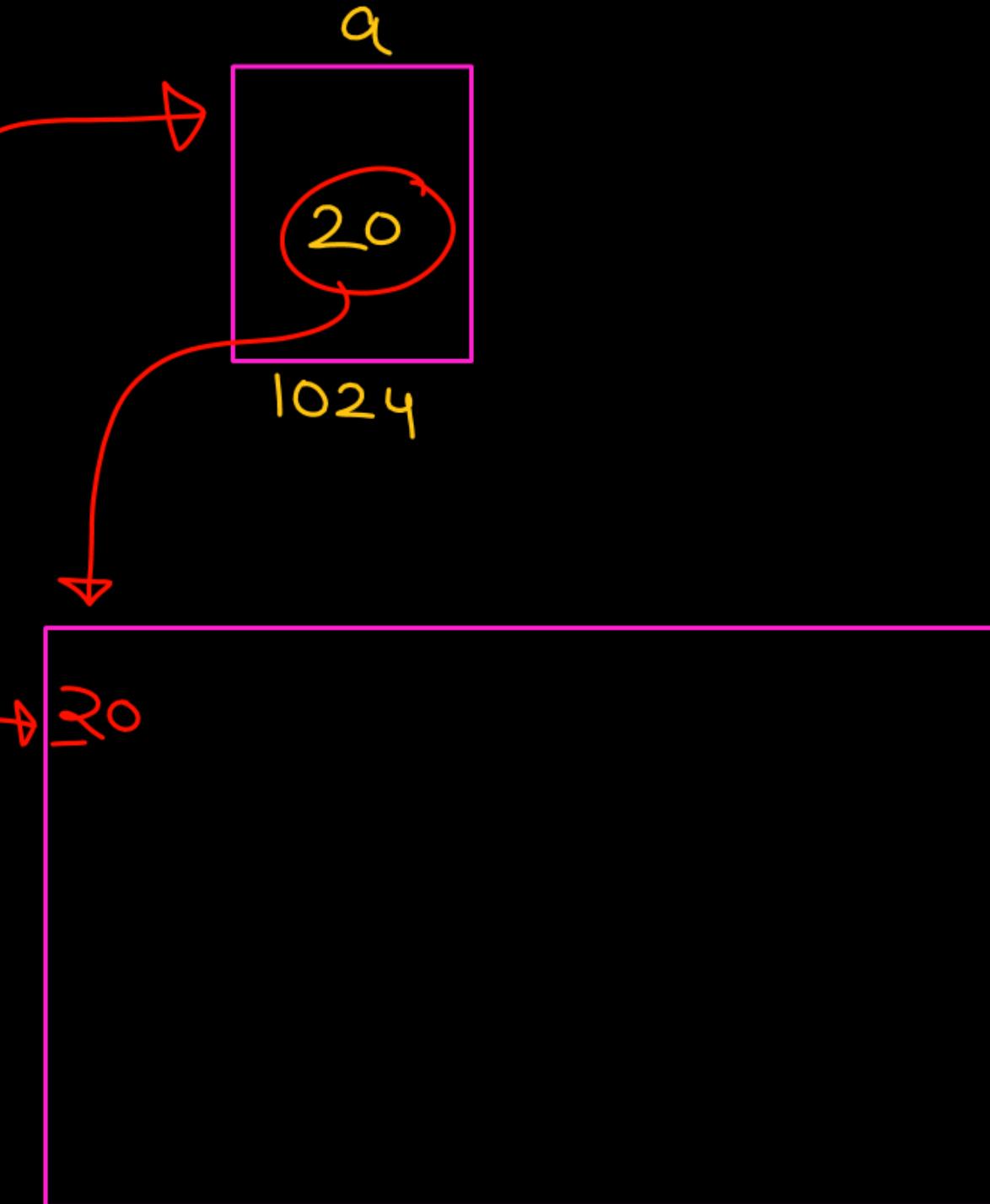
```
#include<stdio.h>
```

```
void main(){
```

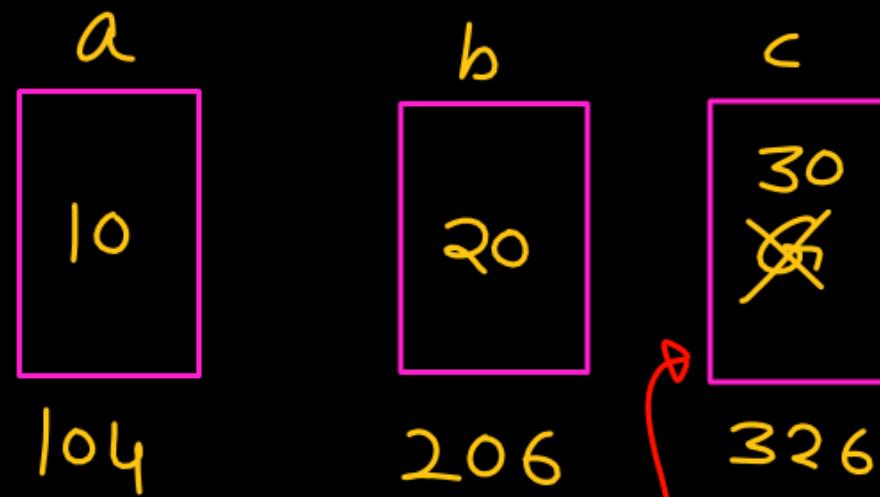
```
    int a = 20;
```

```
    printf("%d", a)
```

short int
int



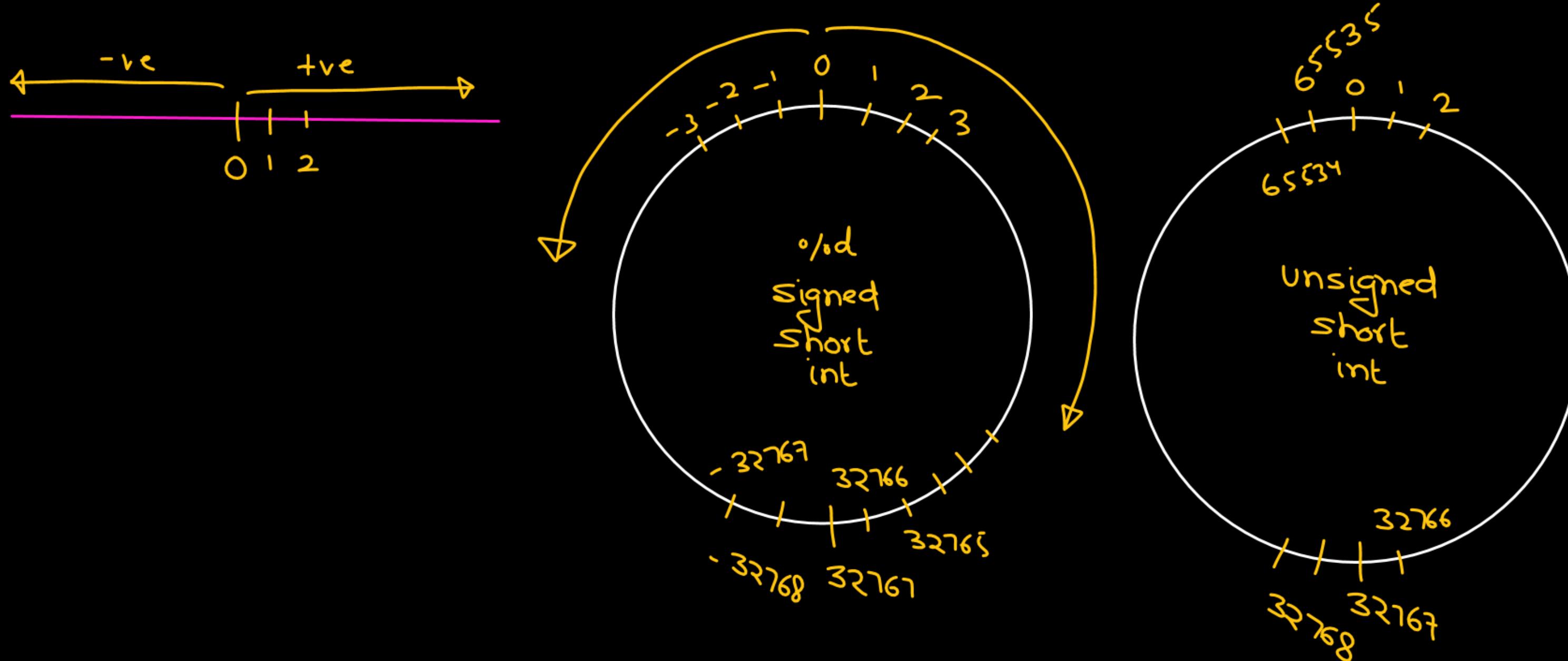
```
#include<stdio.h>
Void main(){
    int a,b,c;
    a=10;
    b=20;
    c=a+b; ✓
    printf("The sum of %d and %d is %d",a,b,c);
    printf("The sum of %d and %d is %d",b,a,c);
}
```



The sum of 10 and 20 is 30
The sum of 20 and 10 is 30

$\text{Short} \rightarrow 2\text{byte}$ (cyclic property)

Signed -32768 to +32767
Unsigned : 0 to 65535

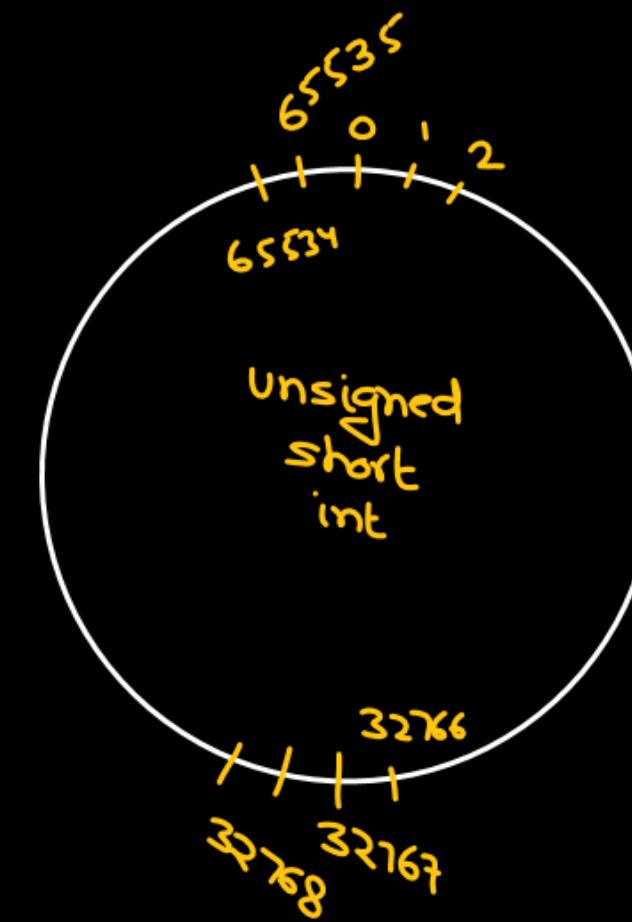
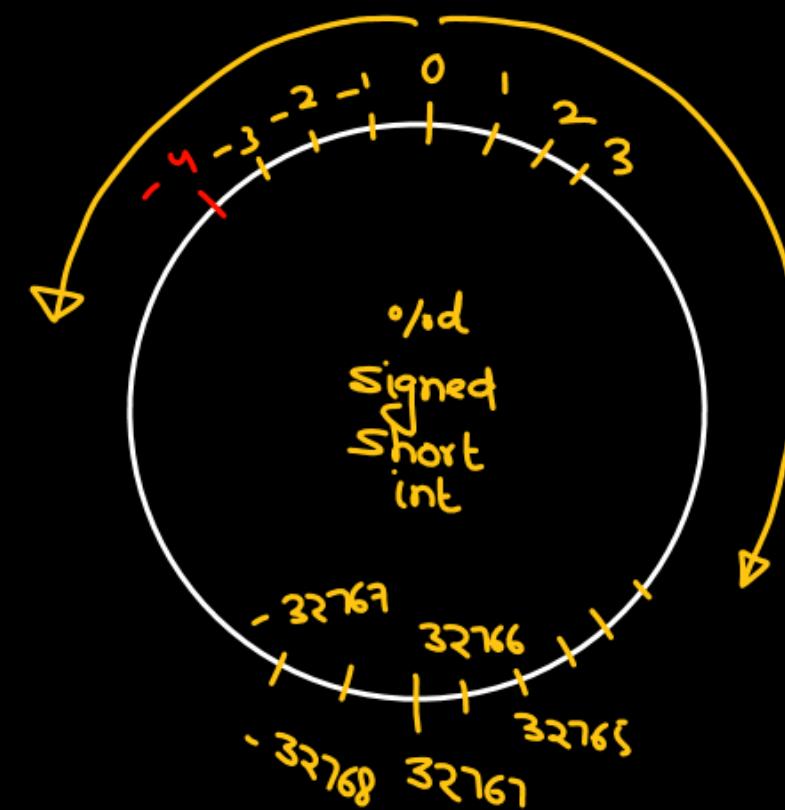


```
#include<stdio.h>  
void main( ){
```

(cyclic property)

 Short int i = -4;
printf ("%d", i);
}
i
-4

-4



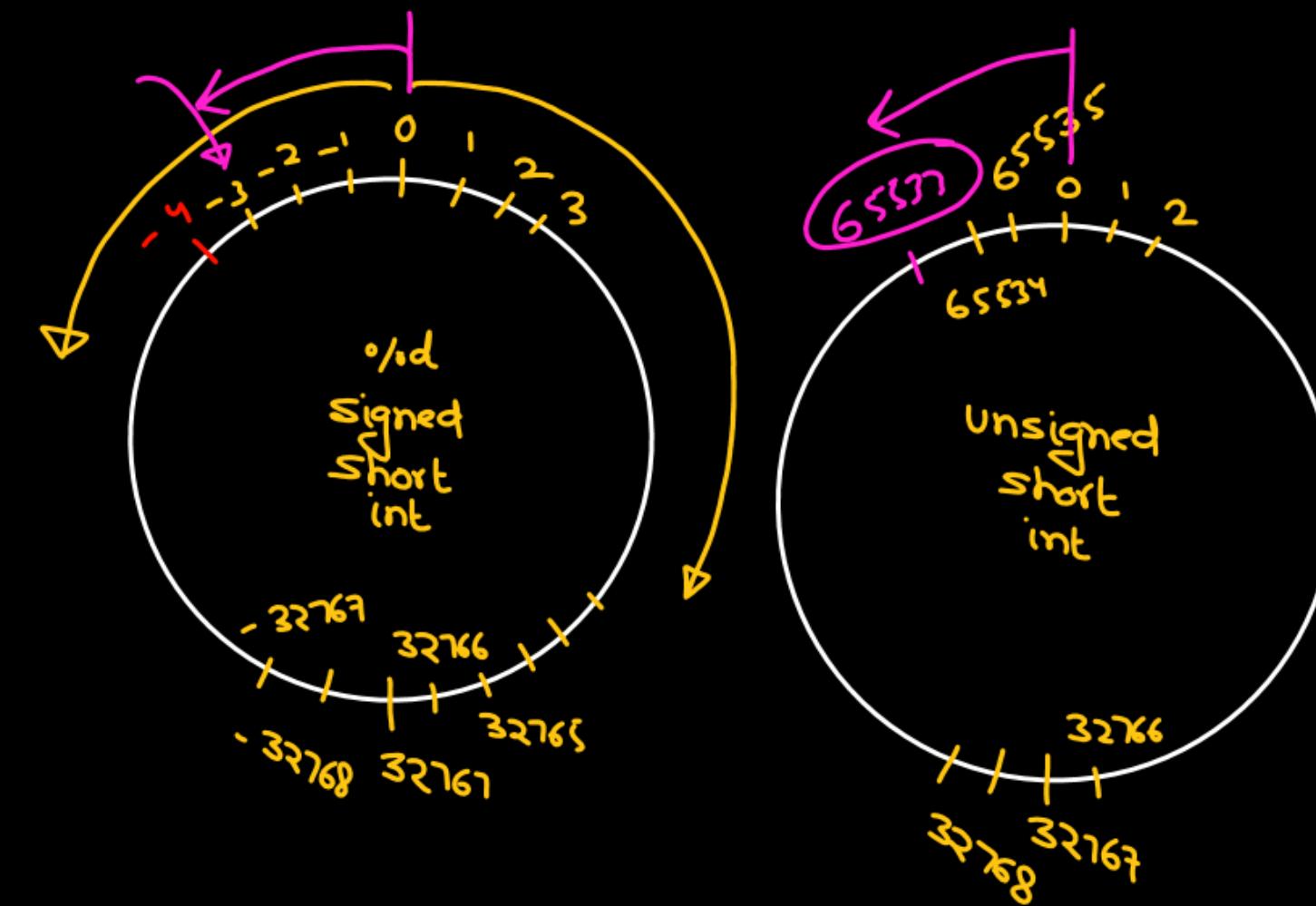
```
#include<stdio.h>
void main(){
```

Unsigned Short int i = -3; ✓

```
printf("%u", i);
}
```

65533
2196

(cyclic property)



(cyclic property)

```
#include<stdio.h>
```

```
void main(){
```

Signed short i = 32769;

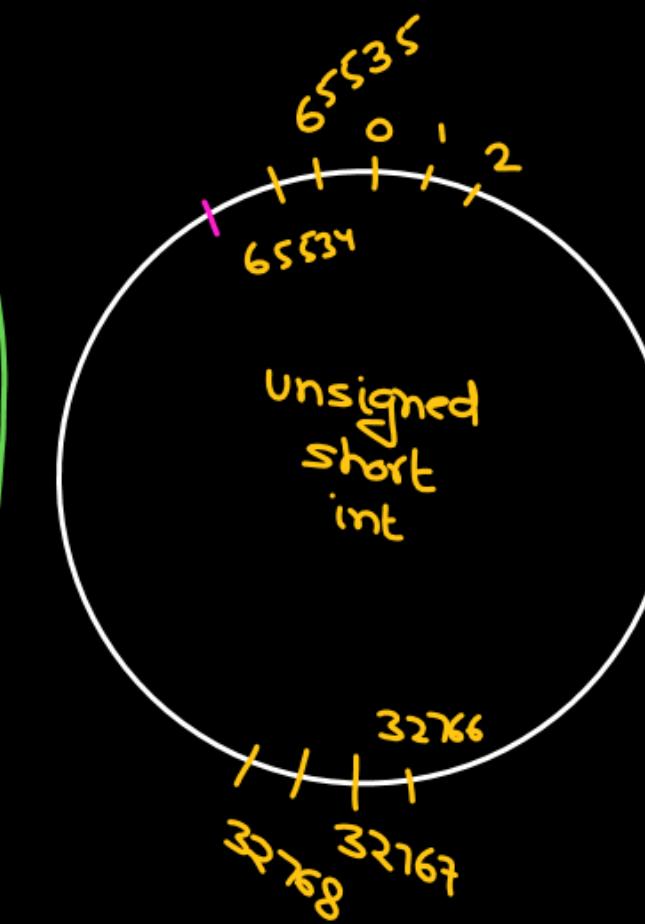
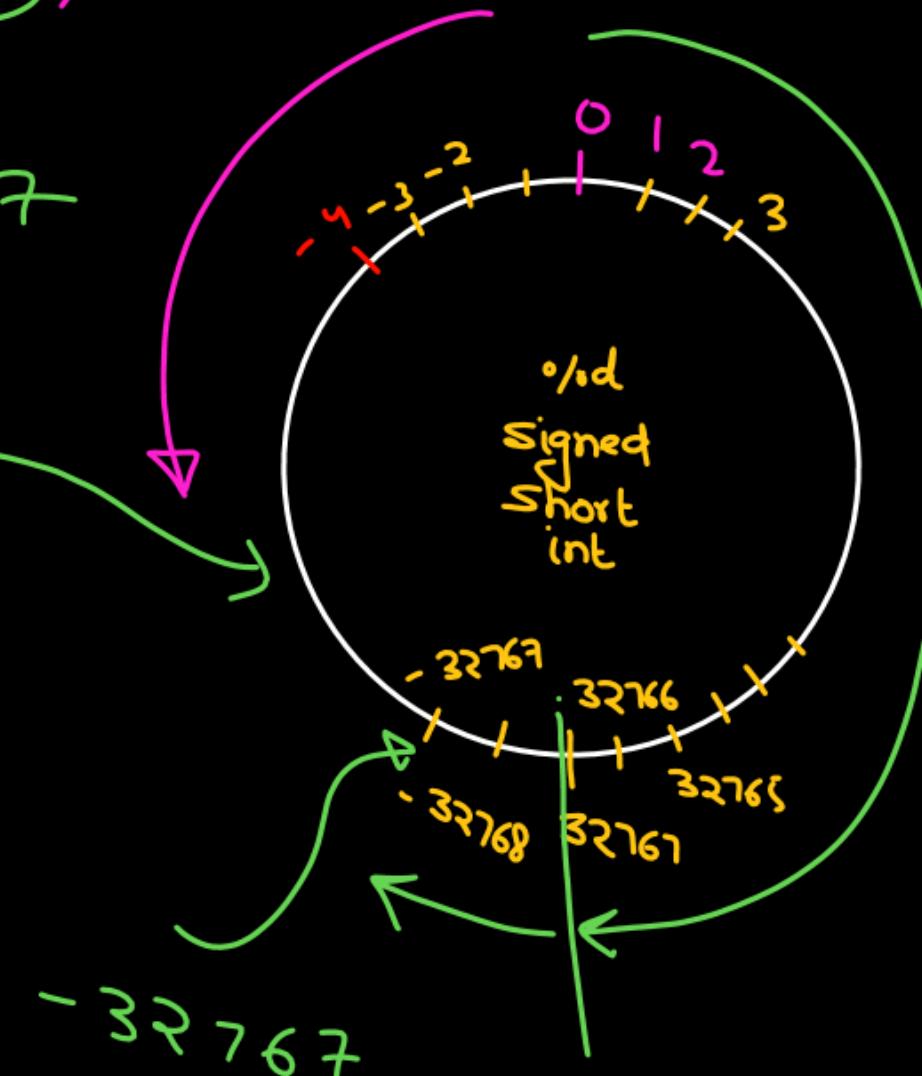
```
printf("%d", i); -32767
```

```
}
```

i

```
-32767
```

→ 2 more than
32767



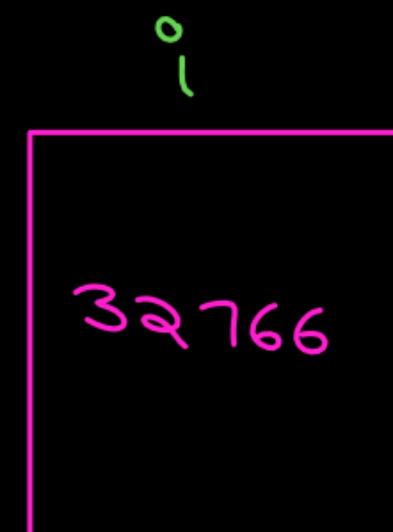
```
#include<stdio.h>
```

```
void main(){
```

```
    signed short int i = -32770;
```

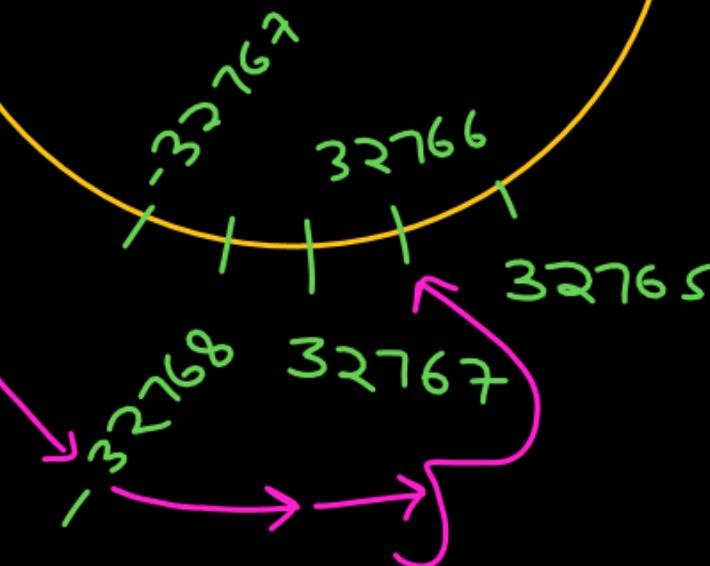
```
    printf("%d", i);  
}
```

O/P : 32766



2 less than
-32768

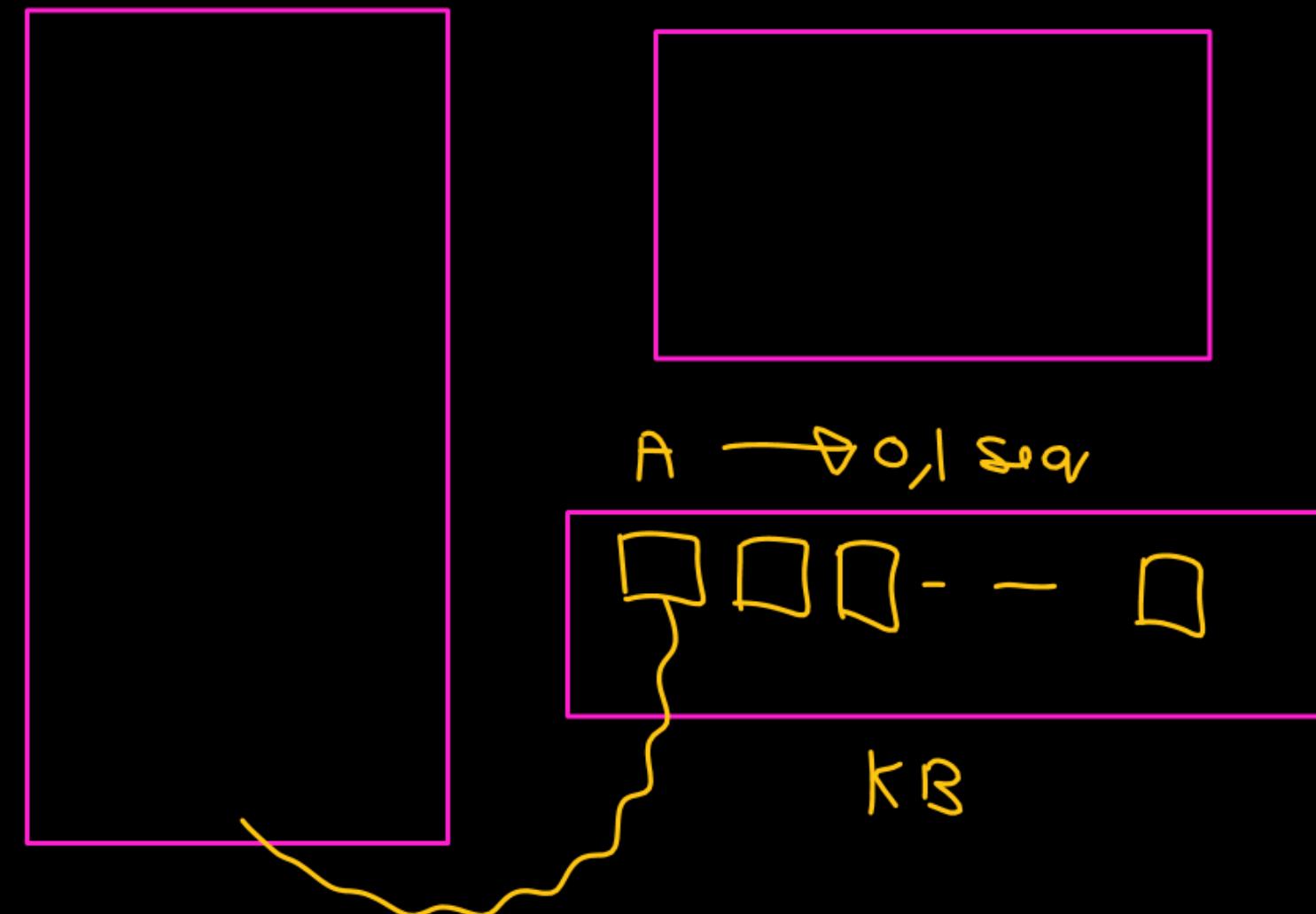
signed



How to take i/p from keyboard

① format

scanf



A - 0|00000|
@ → —
g - 0000|00|

```

#include<stdio.h>
void main(){
    int a ;
    printf("Enter a number");
    scanf(".%d", &a);
    printf(" The value is %.d",a);
}

```

$\&$ \Rightarrow Address of
operator

1024

a	
10	
Garbage	

Enter a number 10
The value is 10

#include<stdio.h>

void main(){

int a, b;

✓ printf("Enter 1st Number");

→ scanf("%d", &a);

printf("Enter 2nd Number");

scanf("%d", &b);

✓ printf("The no. is %d", a);

✓ printf("Second no. is %d", b);

}

Fetch/Read 2 no.

a

10

1024

b

20

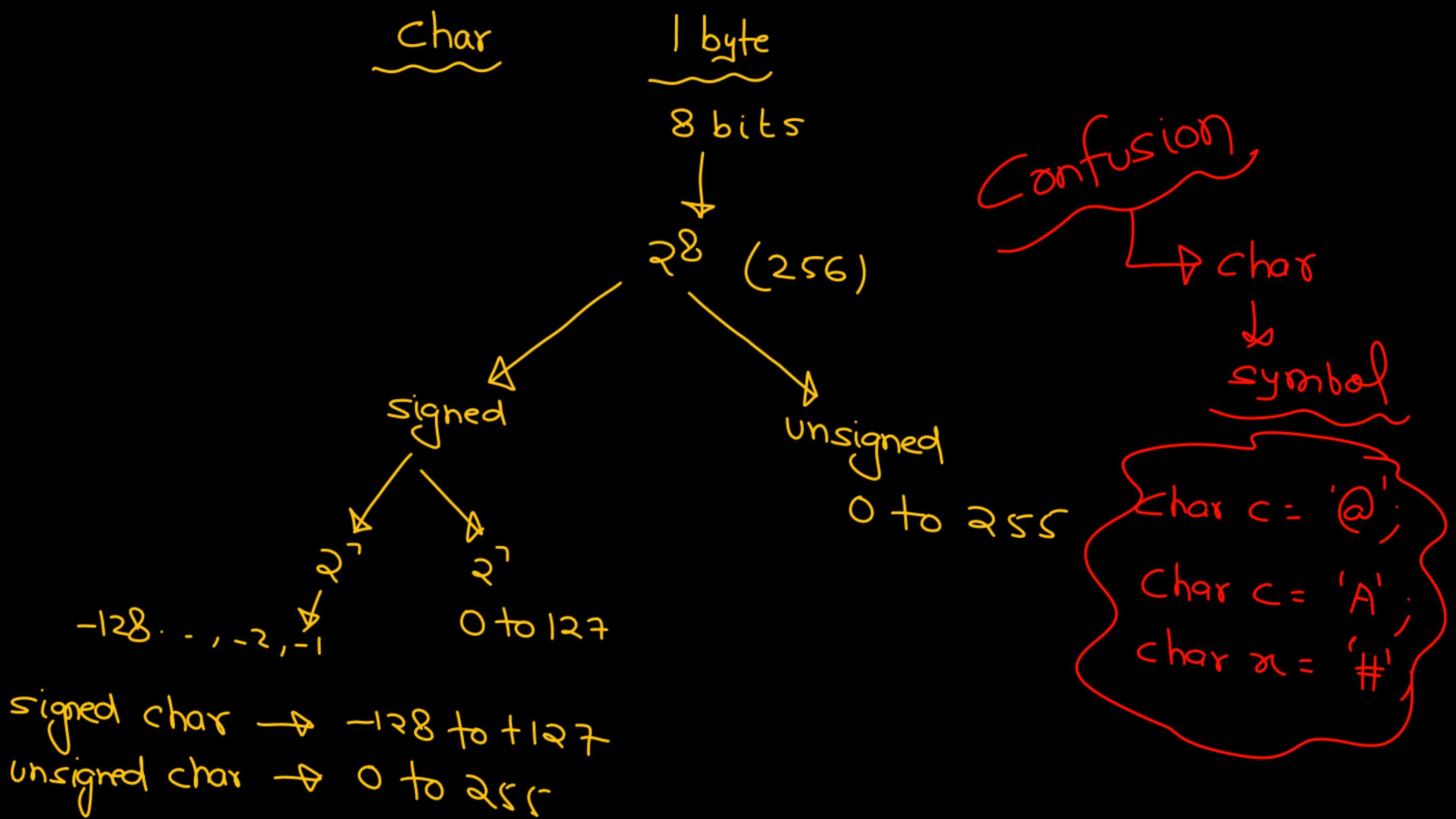
2196

scanf("%d %d", &a, &b);

Enter 1st Number 10 ↵

Enter 2nd Number 20 ↵

_



```
void main(){  
    int a = 10;  
    @  
}
```

010 - 0 ~
00001010

10100 - -0

Number Number
10 → 00001010
= → 00. - 10
{ → 00100. ..

CHARACTER SYSTEM

language

characters
(symbol)

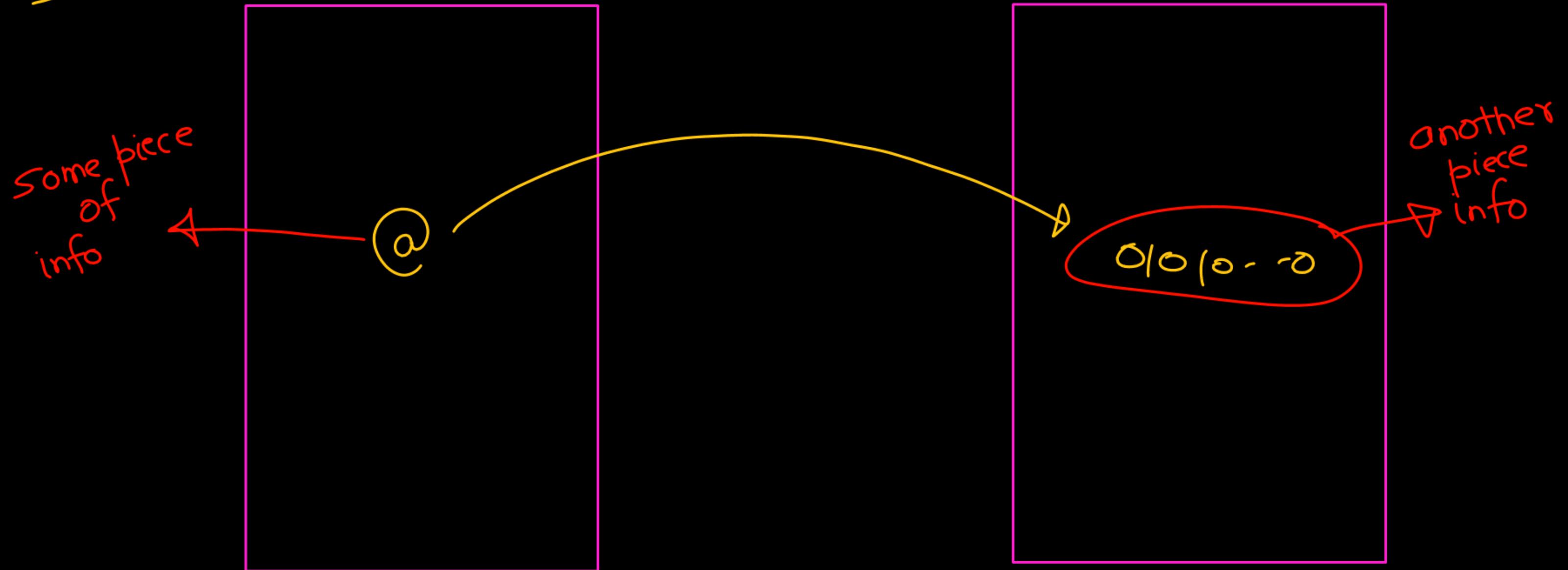
char. system

M/c

using a
char. system
we can rep.
all the symbols
by +ve
integer constant

English-type

ASCII



A — 65

B — 66

C — 67

:

:

:

Z — 90

a — 97

b — 98

c — 99

:

:

:

₹ — 122

0 — 48

1 — 49

2 — 50

:

:

:

(i) Symbols
+ve Integer
Constant

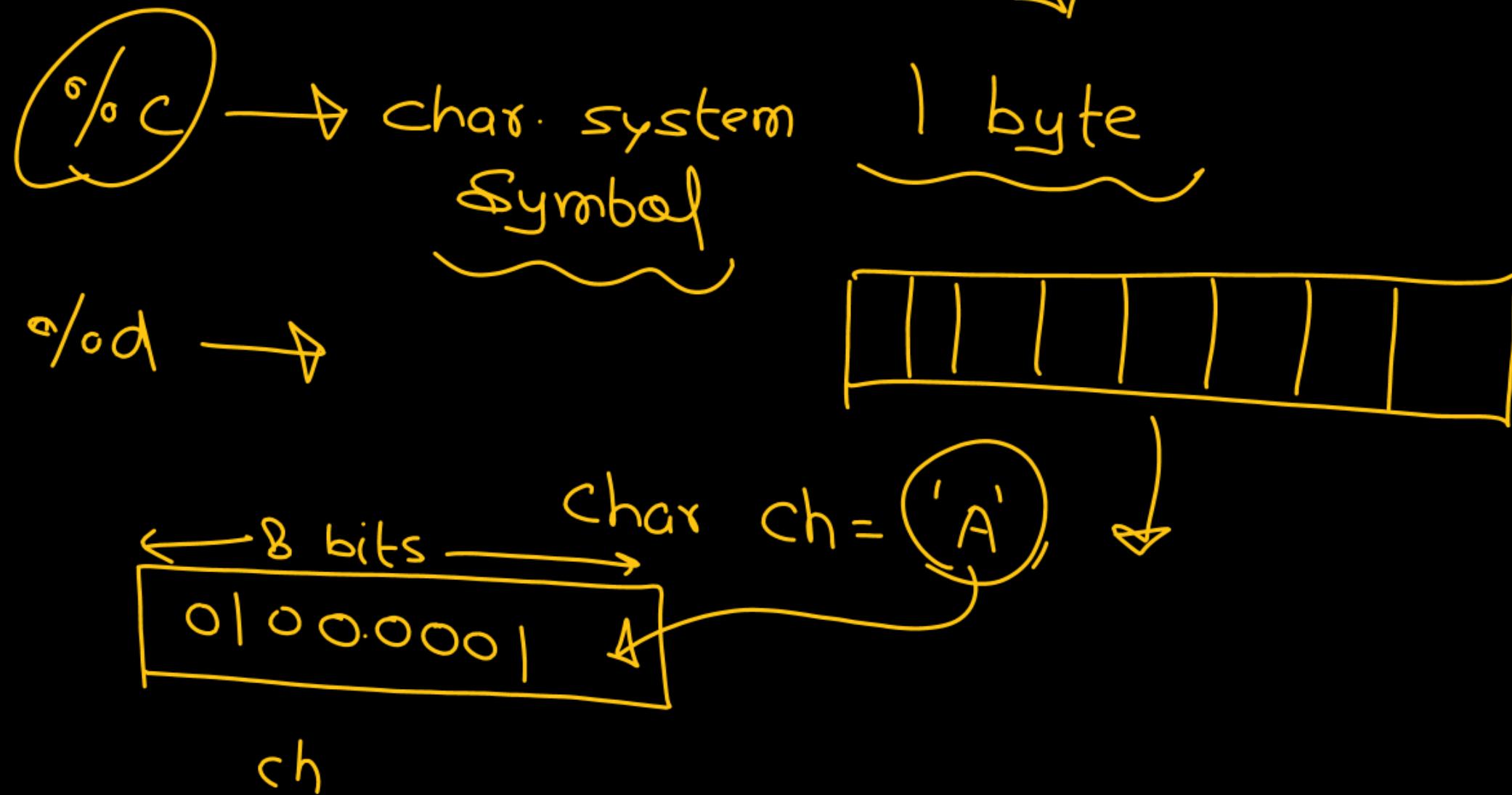
(ii) 1 byte \Rightarrow 256

0 to 255

Character System

int i = -1;

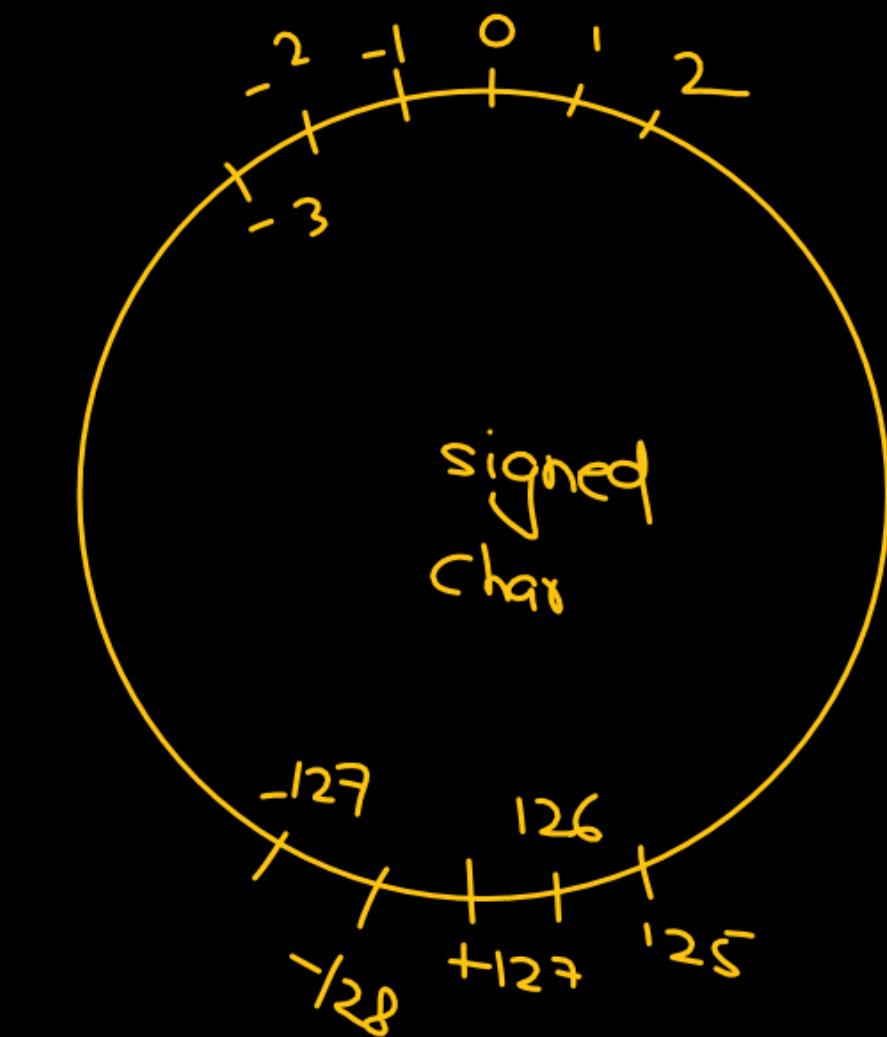
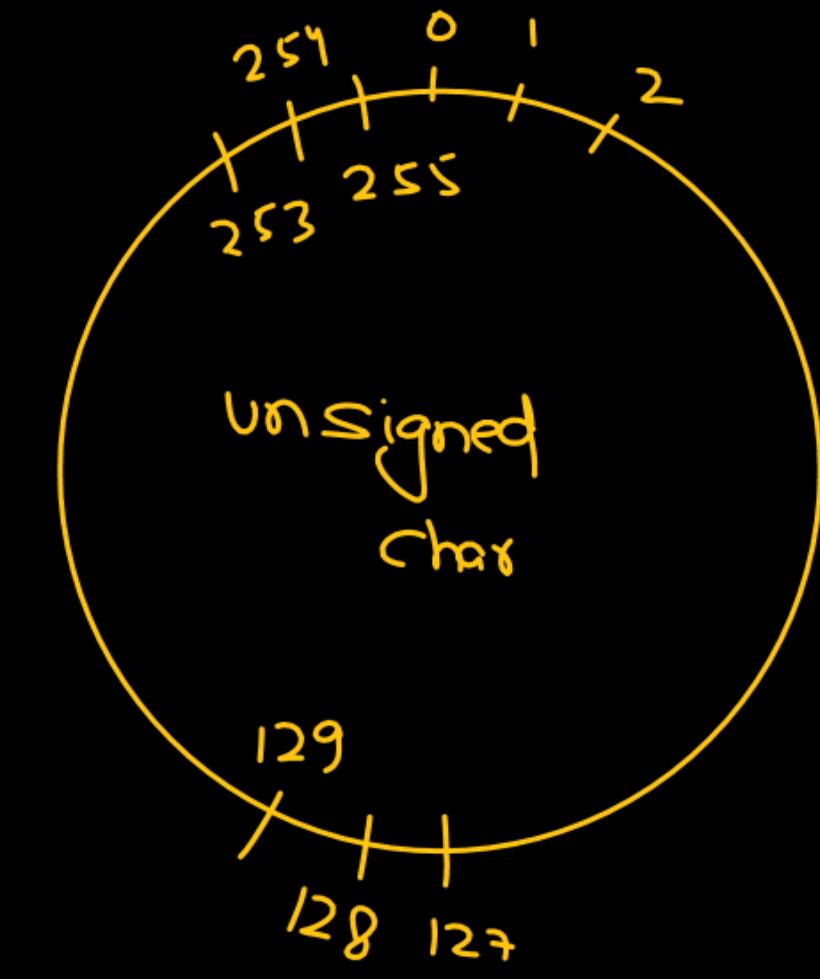
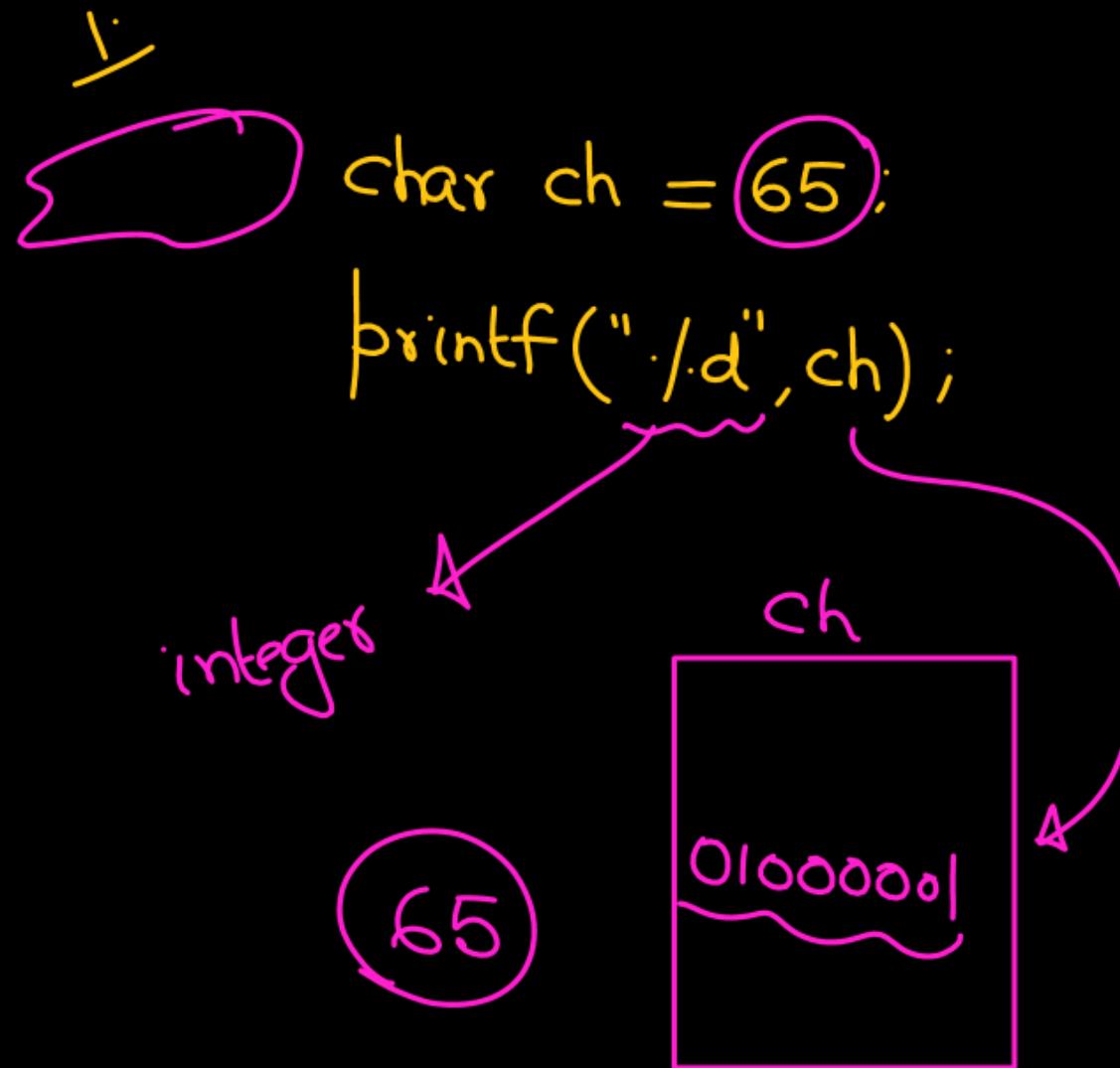
char ch = '@';



char → 1 byte 8bit 2^8

unsigned char \Rightarrow 0 to 255

signed char \Rightarrow -128 to +127



char → 1 byte 8bit 2^8

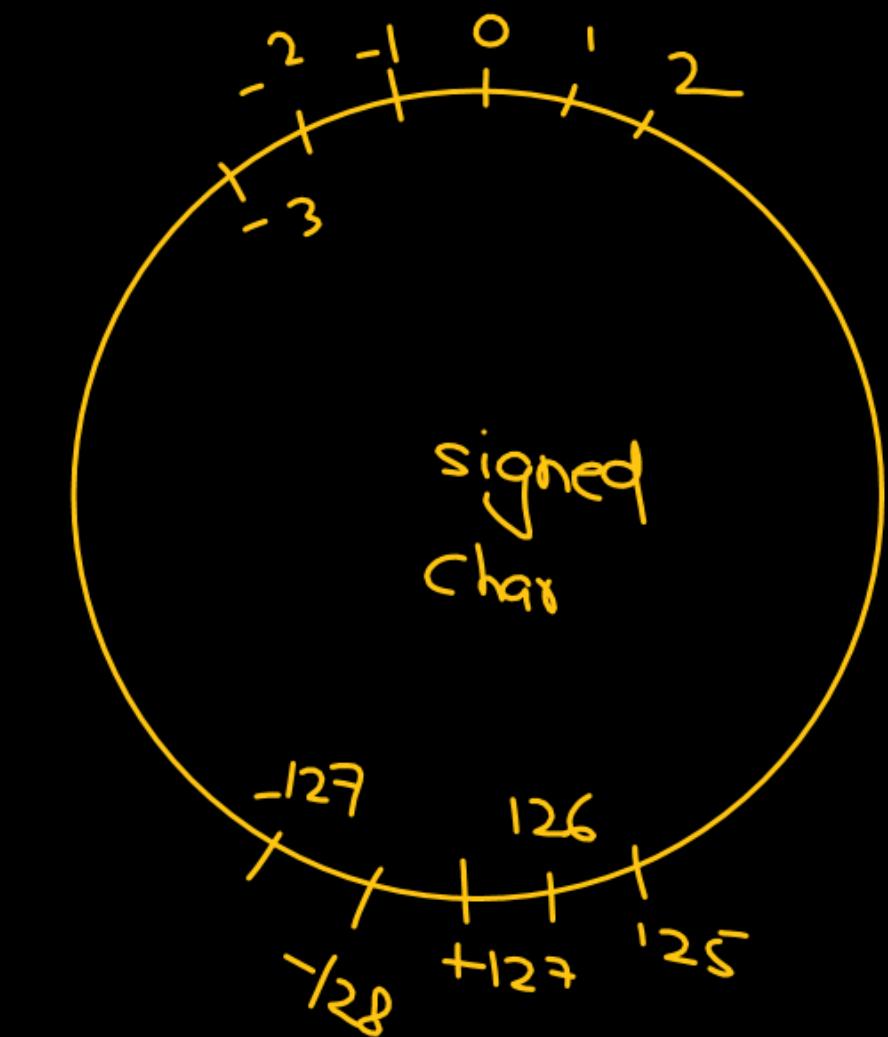
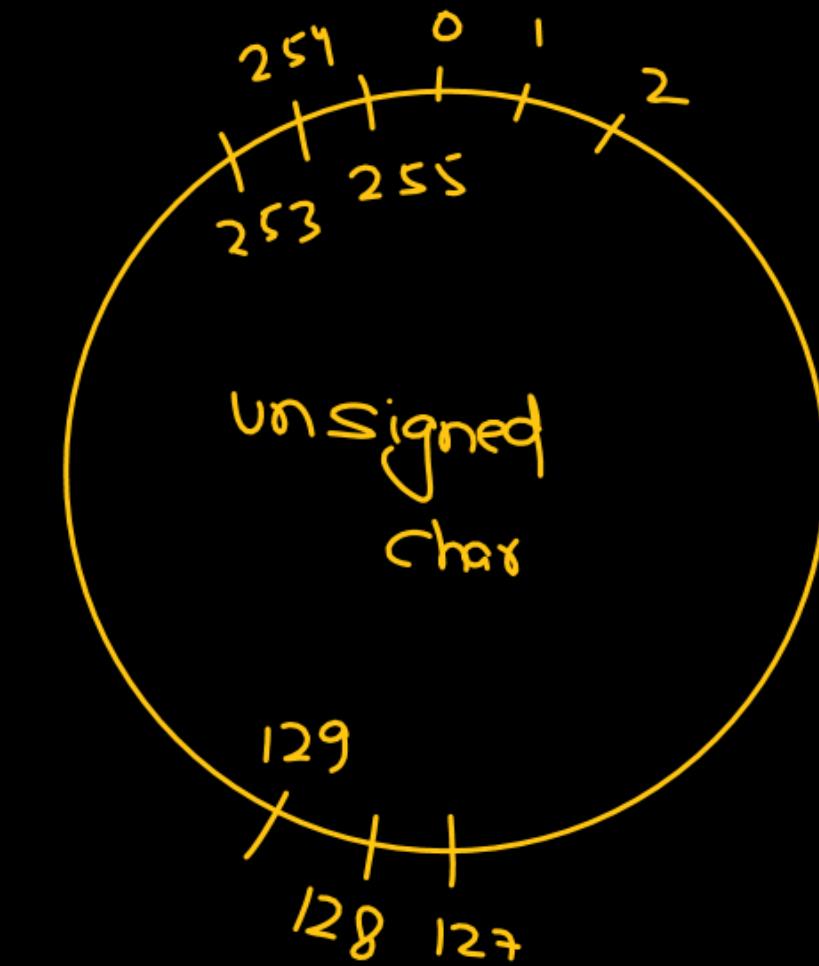
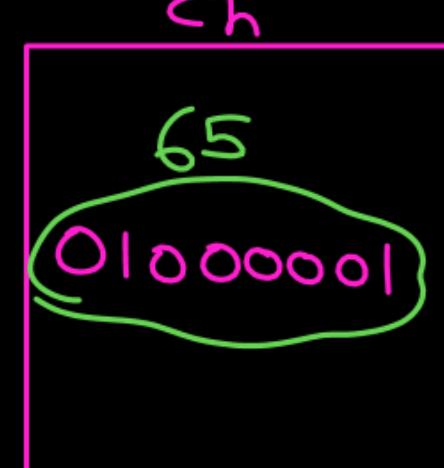
Unsigned char \Rightarrow 0 to 255

`signed char` \Rightarrow -128 to +127

2 char ch = 65 ;

```
printf("./c", ch);
```

Char. system
Symbol



char → 1 byte 8bit 2^8

unsigned char \Rightarrow 0 to 255

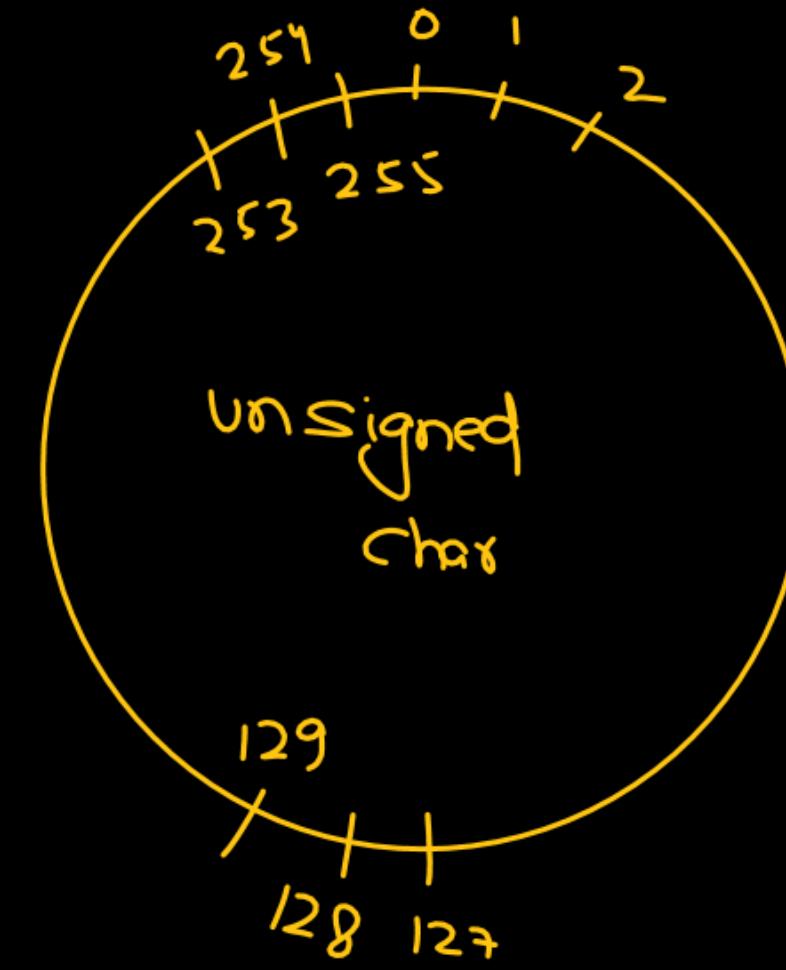
signed char \Rightarrow -128 to +127

3.
char ch = 'A';

printf("./c", ch); A

Symbol

ch
01000001 → 65



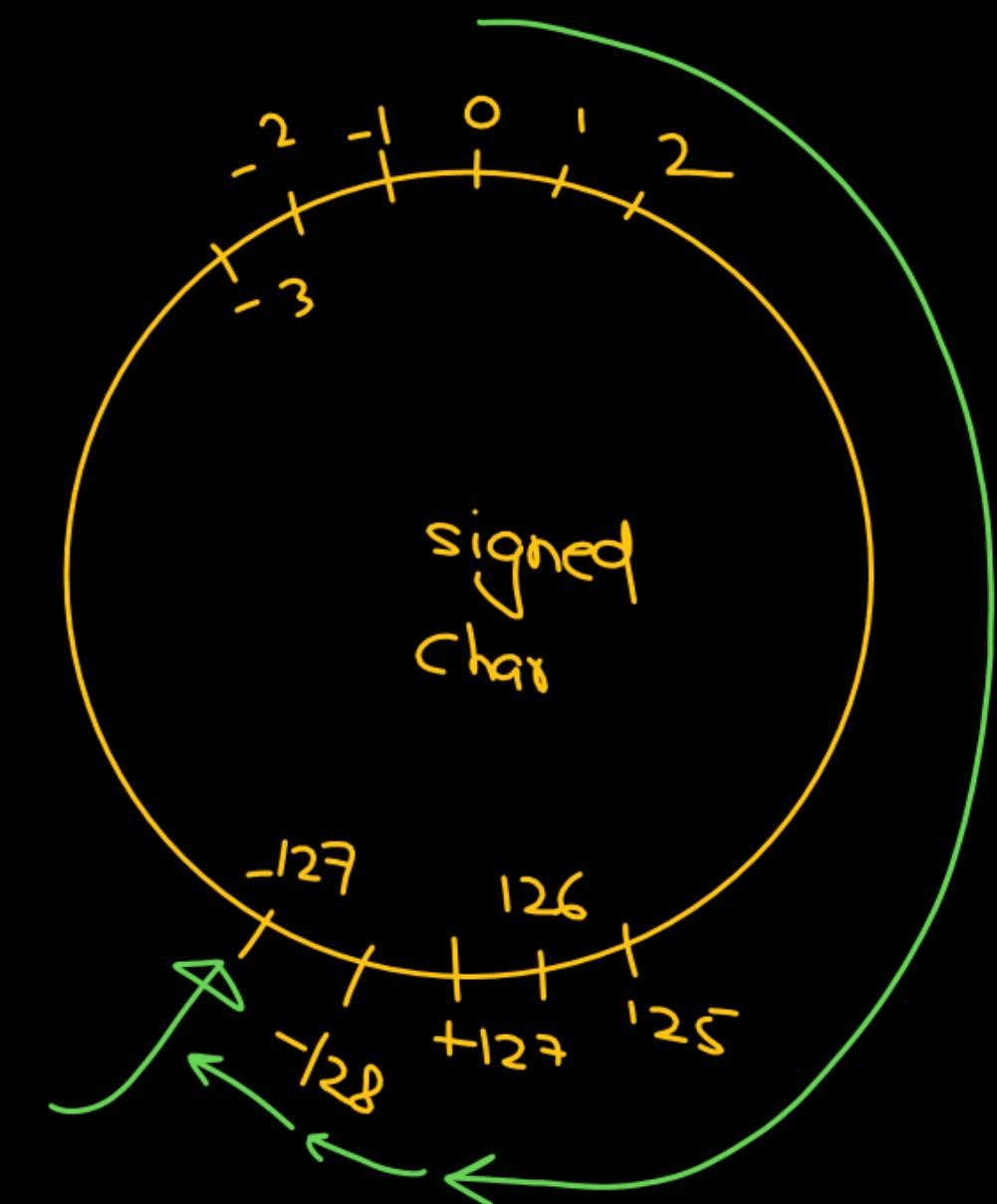
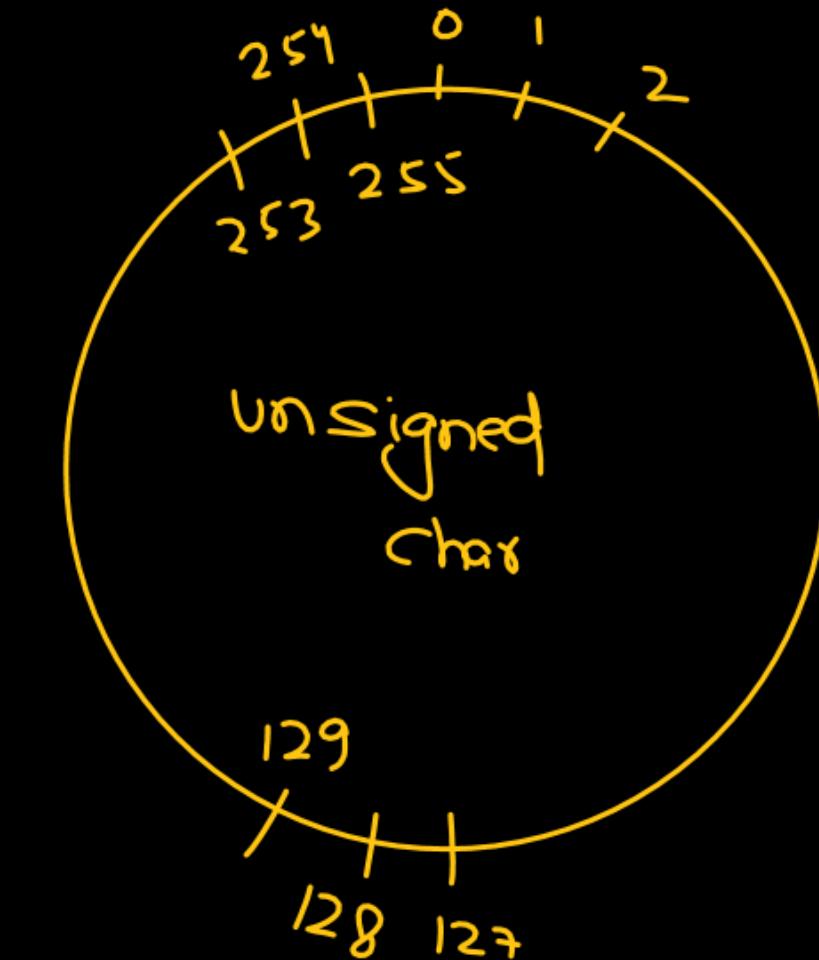
char ch = 129;
by default signed

printf("%d", ch);

O/P: -127

ch
-127

char → 1 byte 8bit 2^8
unsigned char ⇒ 0 to 255
signed char ⇒ -128 to +127



char → 1 byte 8bit 2^8

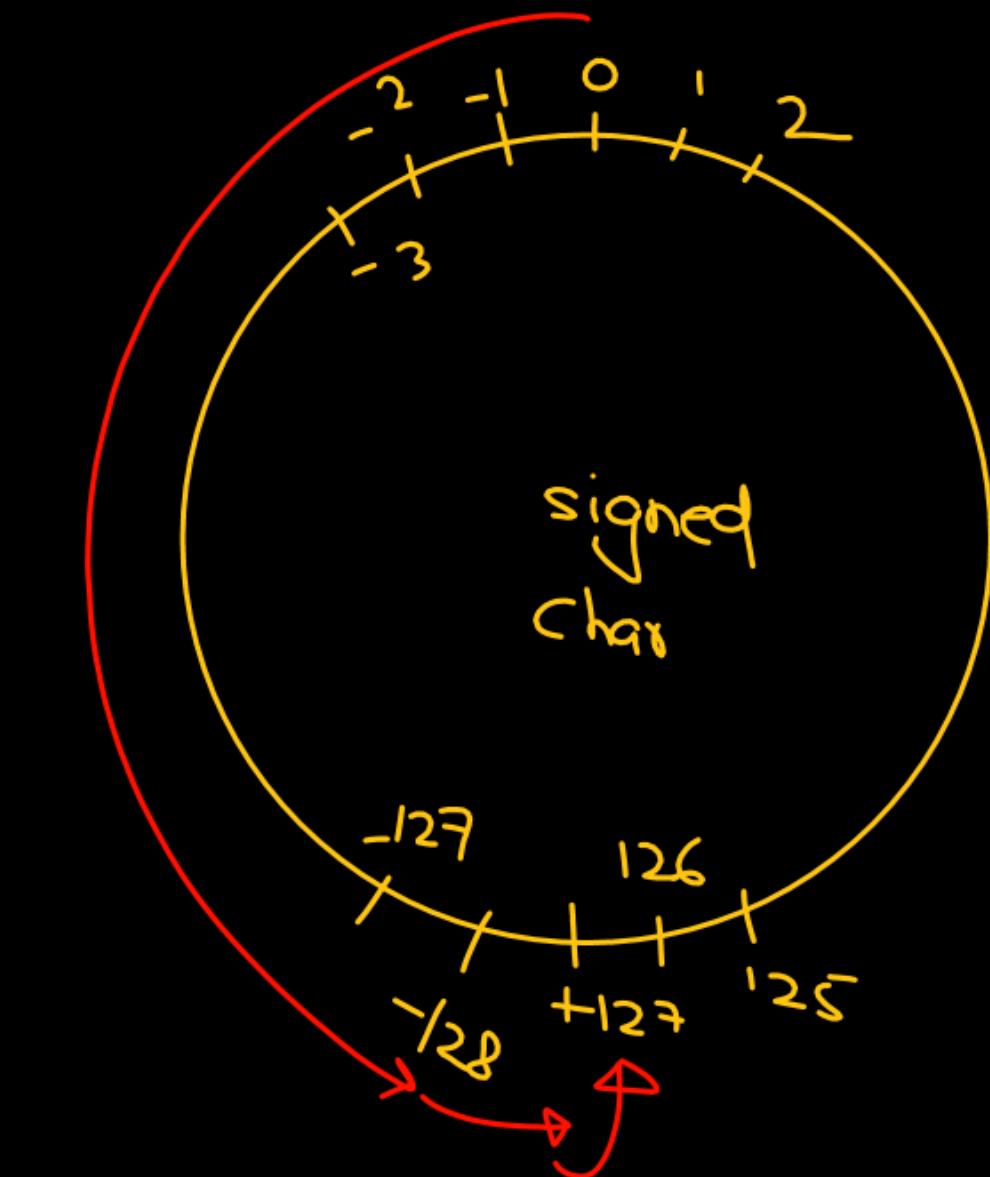
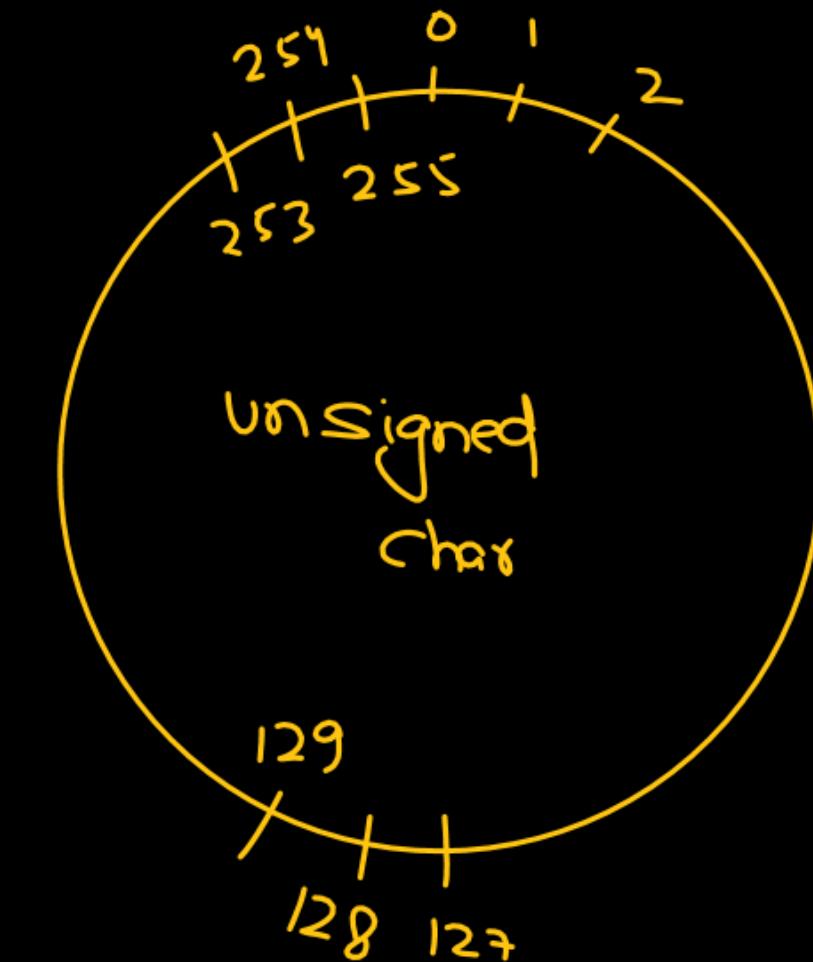
unsigned char \Rightarrow 0 to 255

signed char \Rightarrow -128 to +127

signed
char ch = -129;
printf("%d", ch);

O/P: 127

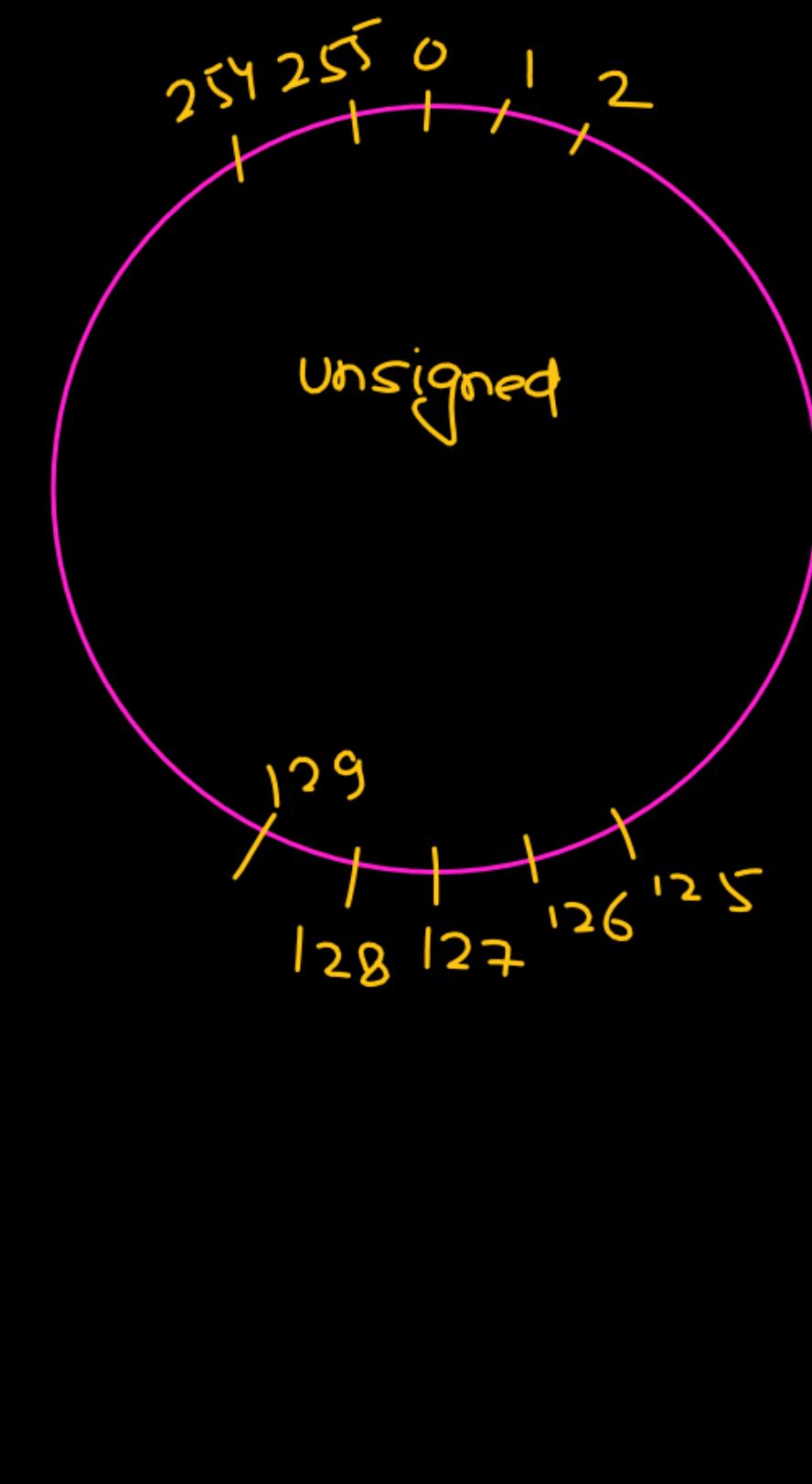
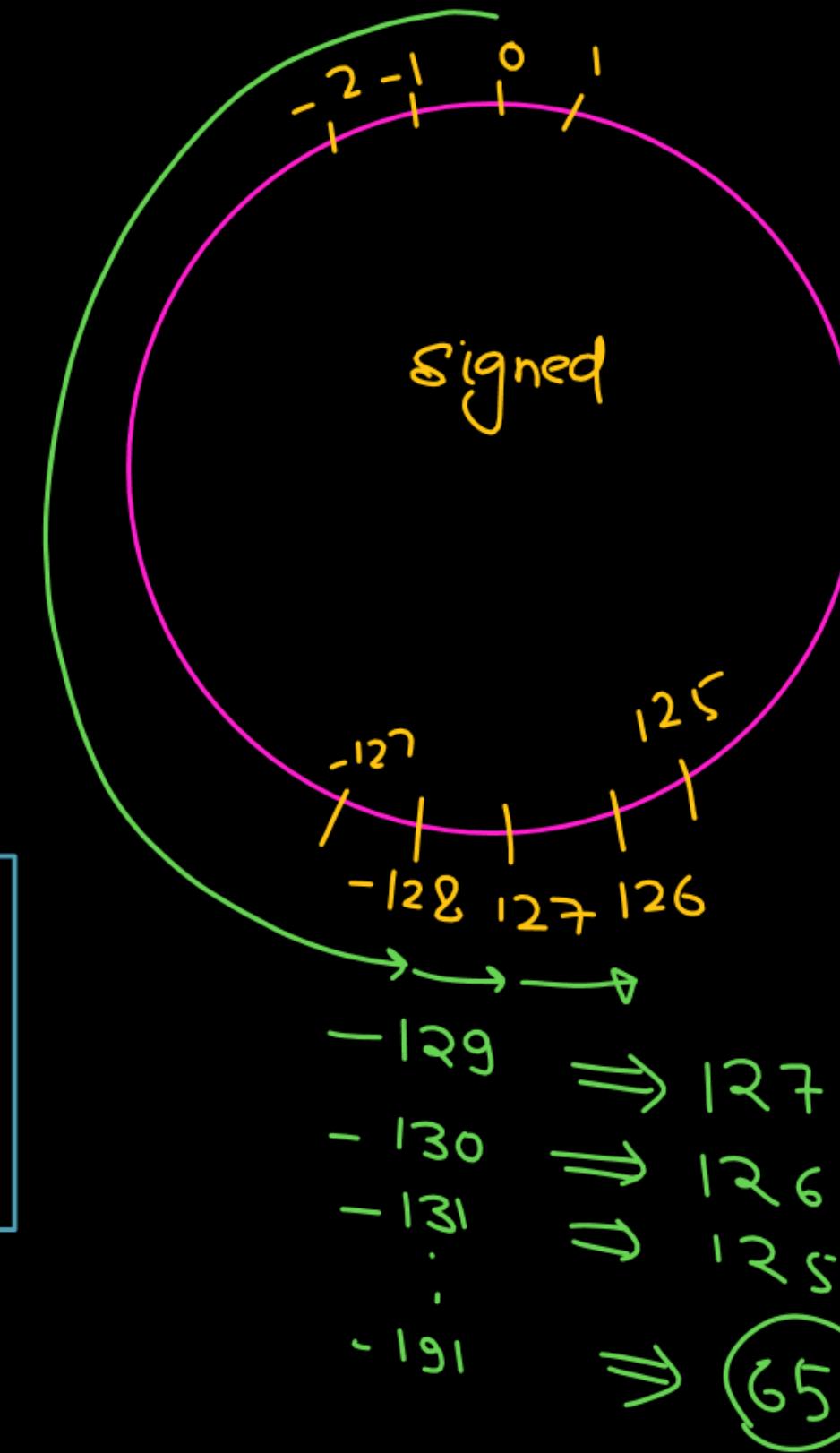
ch
127

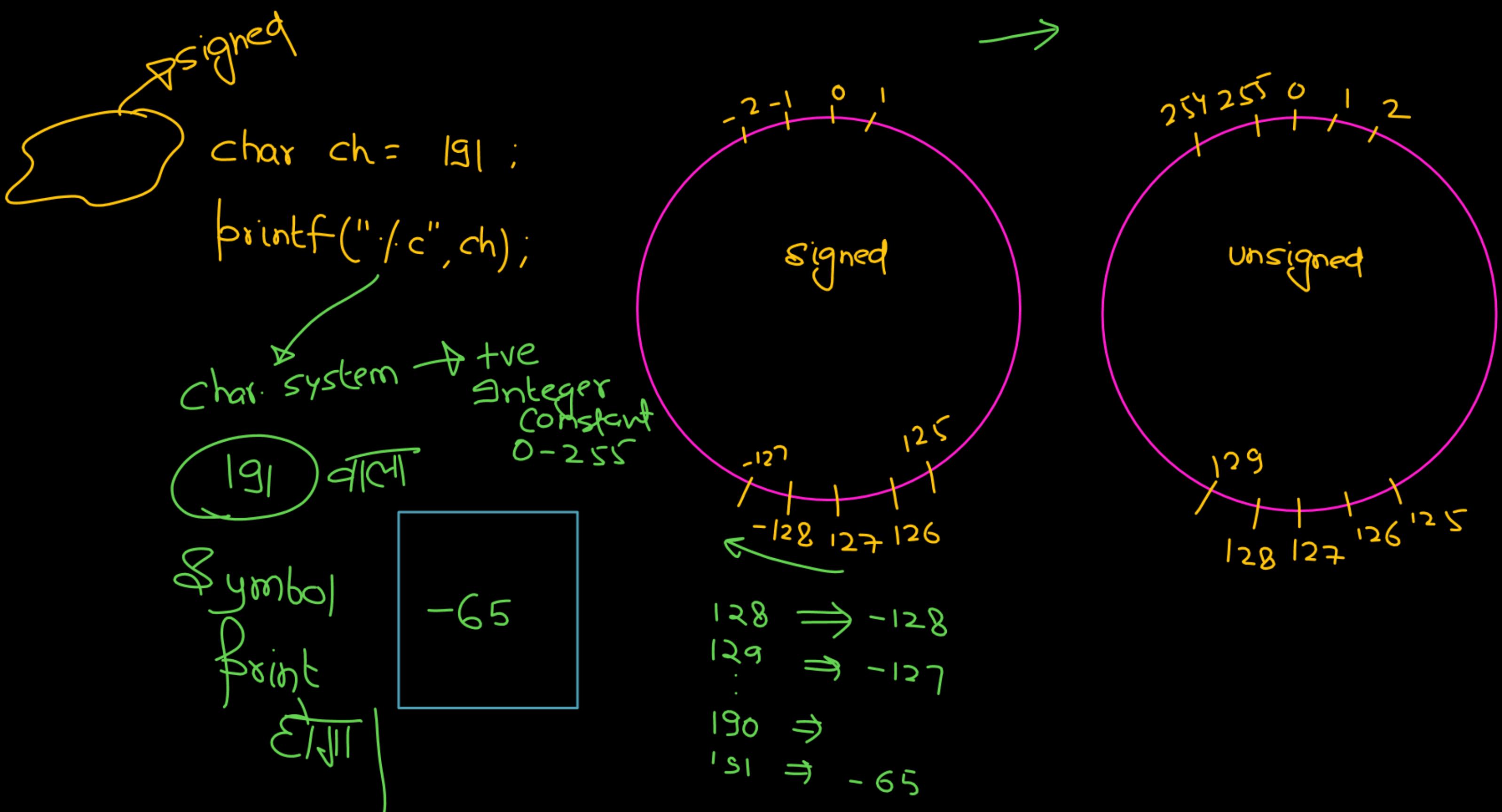


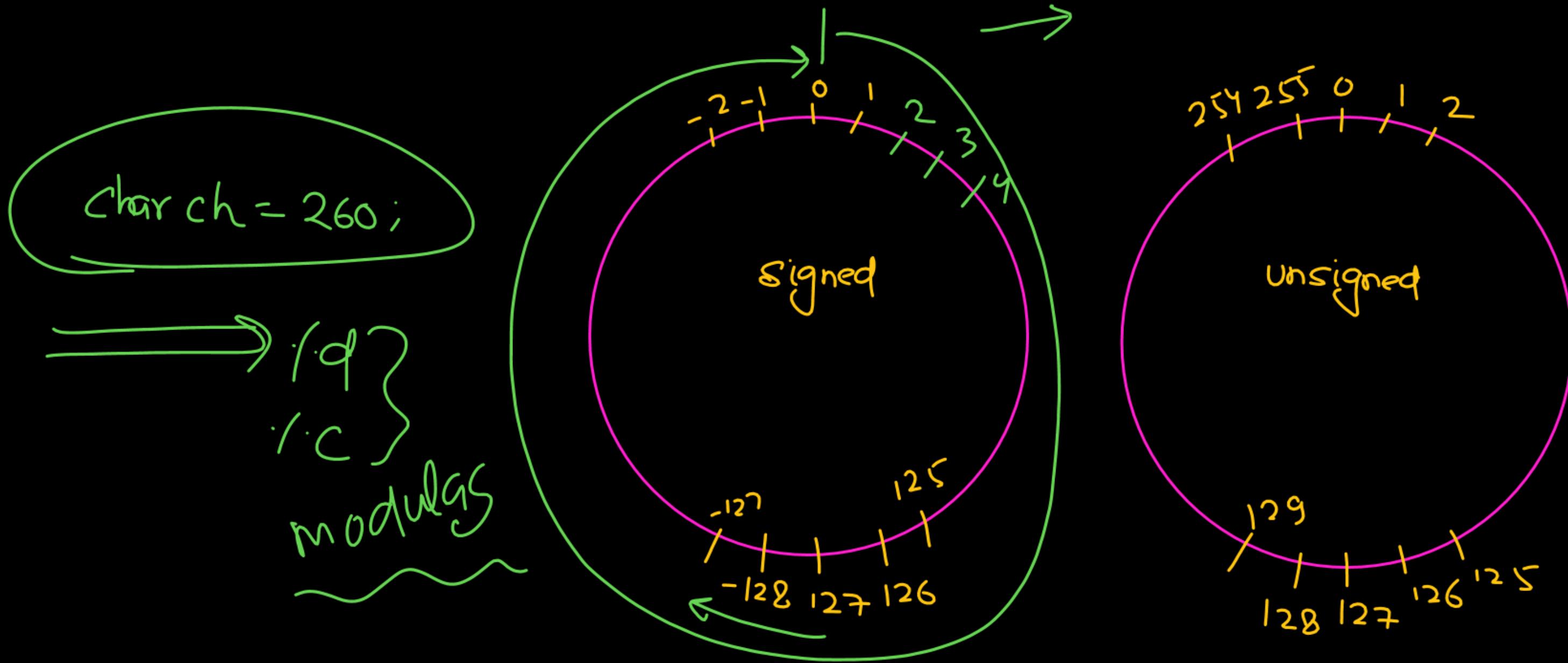
```
char ch = -191 ;  
printf("./c", ch)
```

A

65







$128 \Rightarrow -128$
 $129 \Rightarrow -127$
 \vdots
 $190 \Rightarrow$
 $151 \Rightarrow -65$

```
#include<stdio.h>
void main(){
```

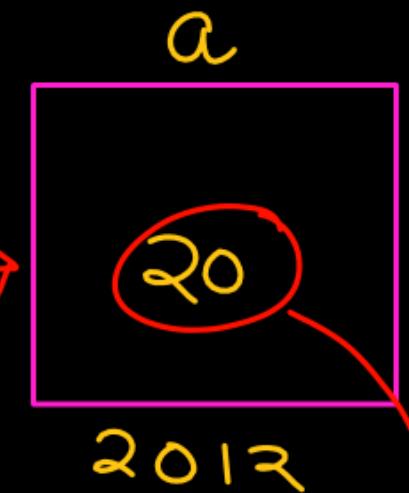
...

```
int a = 20;
```

```
printf(" The value is %.d",a);
}
```

{

- %d → short int,
int
- %f → float
- %c → char



%u → unsigned int

The value is 20

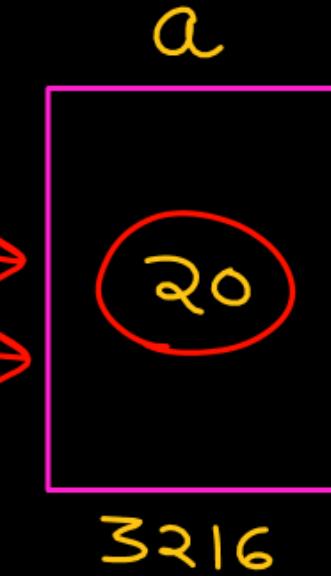
```
#include<stdio.h>
void main(){
```

```
    int a=20;
```

```
    printf("%d is %d",a,a);
```

```
}
```

20 is 20



Concept

superman



%d

%lld

Karan - H_2O
→ $CaCl_2$

