

CS & IT ENGINEERING



C Programming

Data Types and Operators

Lec- 03



By- Pankaj Sharma sir



TOPICS TO BE
COVERED



Introduction to C Programming-3

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

```
void main() {
```

```
    short int a ;
```

```
    signed short int b ;
```

```
    short c ;
```

```
    signed short d ;
```

```
}
```

declaration

✓

✓

✓

✓

} signed short int type

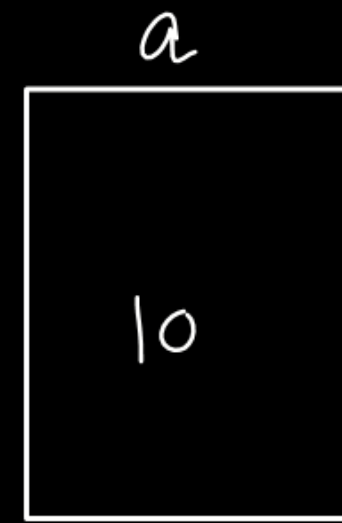
```
#include <stdio.h>
void main() {
```

```
    signed short int a = 10;
```

```
    signed short int b; declaration
```

```
}
```

declaration,
initial value



2016



2048

Garbage

```
#include <stdio.h>

void main() {

    short int a = 20;
```

```
printf("a");
```

}

Text format



pf \Rightarrow " "
 \downarrow
 as it is
 point है ना
 अपना दिखावा नही
 करता है

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

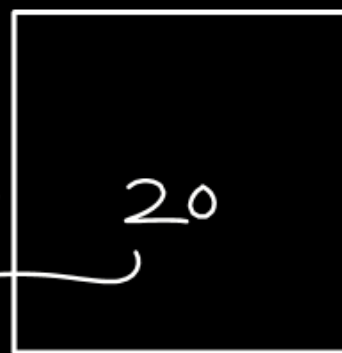


Specify the format



format specifiers

a



2016

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

```
void main() {
```

```
short int a = 20;
```

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

format specifier



```
}
```

20



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

```
void main() {
```

```
    short int a = 20;
```

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

```
}
```

The value is 20

a

20

2048

format specifier

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

```
void main(){
```

```
    signed short int a = 20;
```

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

```
}
```



20 is 20

=


```
#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

/d → short int, int

/ld → long int

/lld → long long int

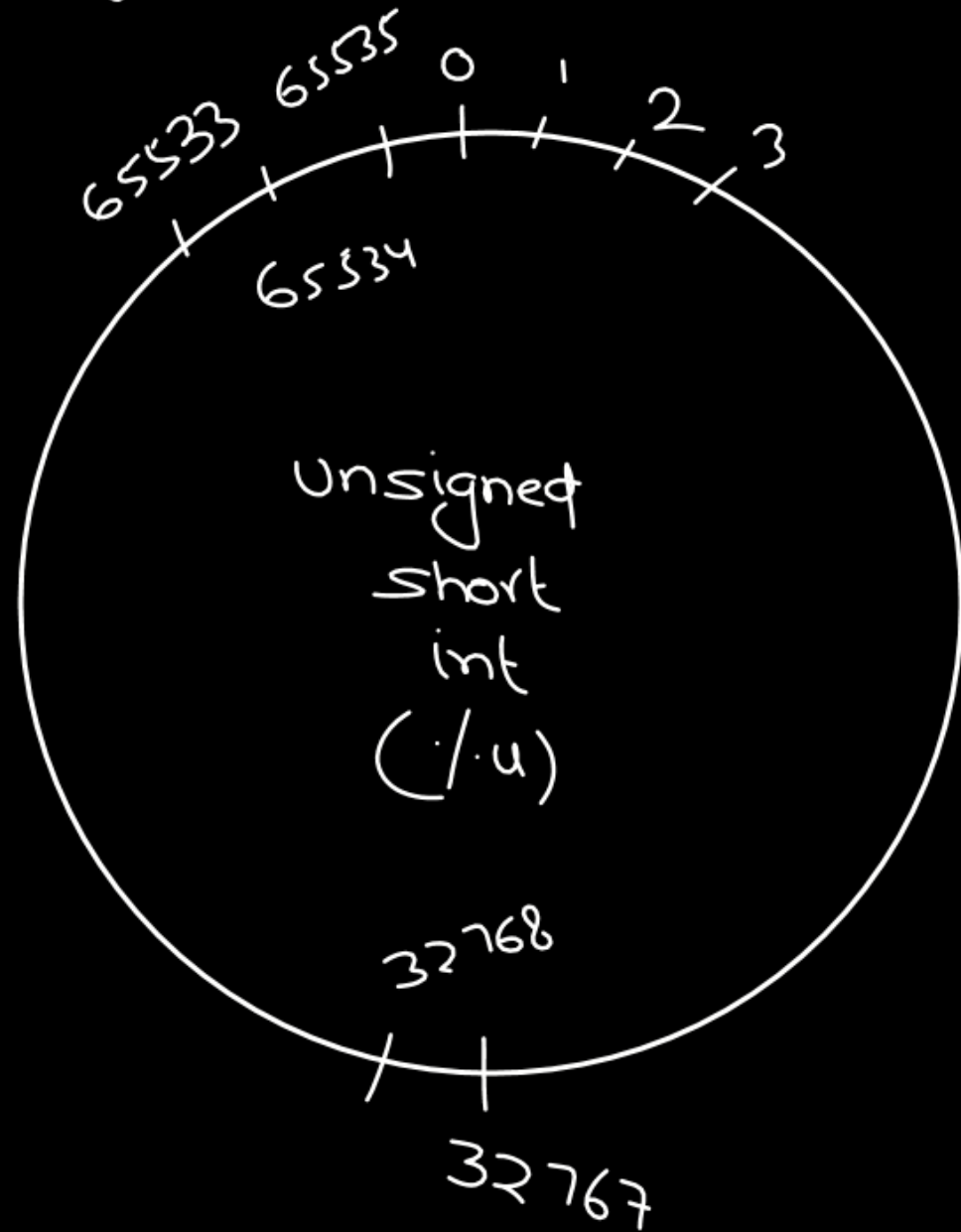
/c → char

/u → unsigned short int,
unsigned int

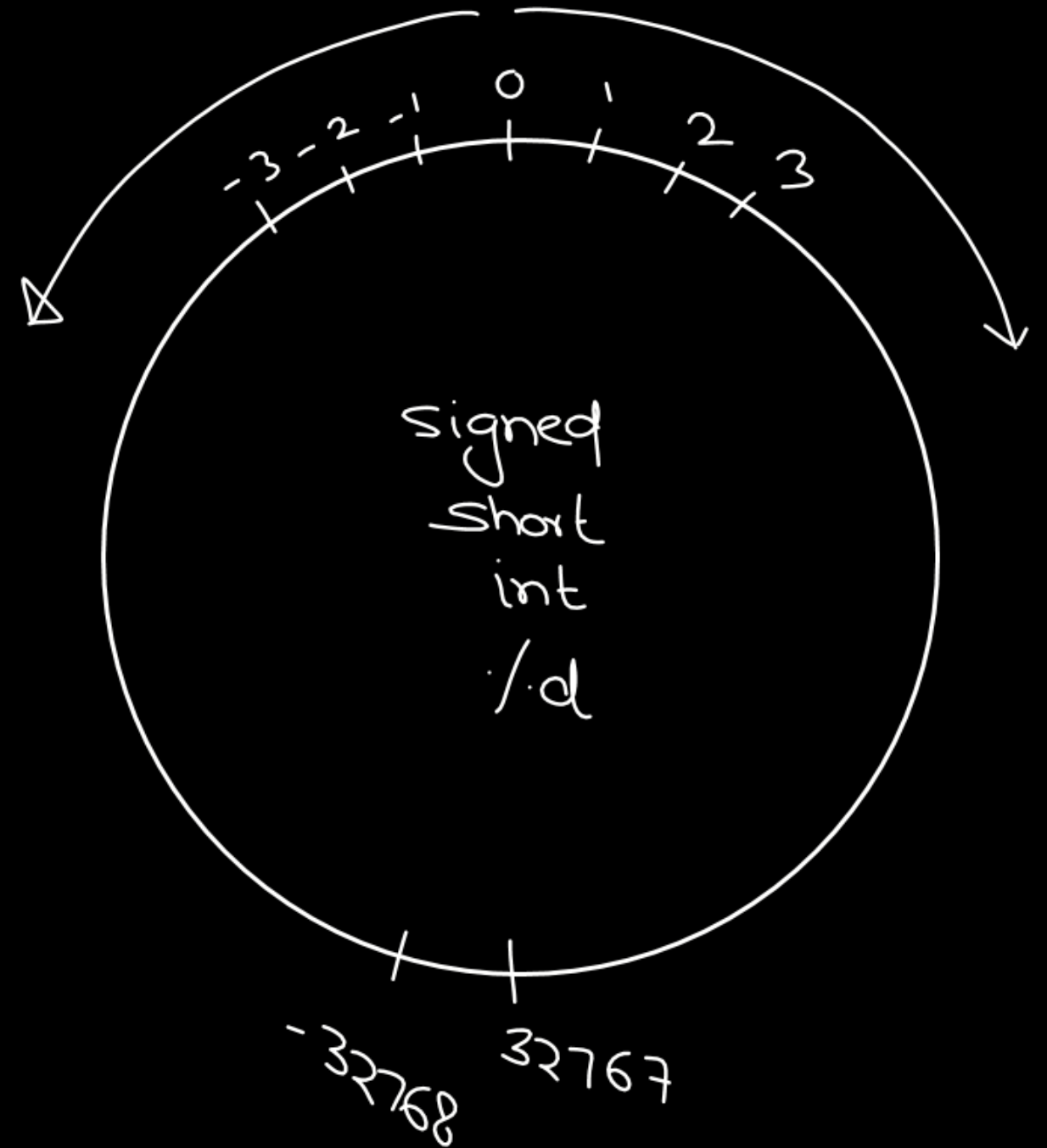
short int \rightarrow 2 byte



unsigned 0 to 65535



signed: -32768 to +32767




short int \rightarrow 2 byte



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

```
void main(){
```

 short int i = -4;

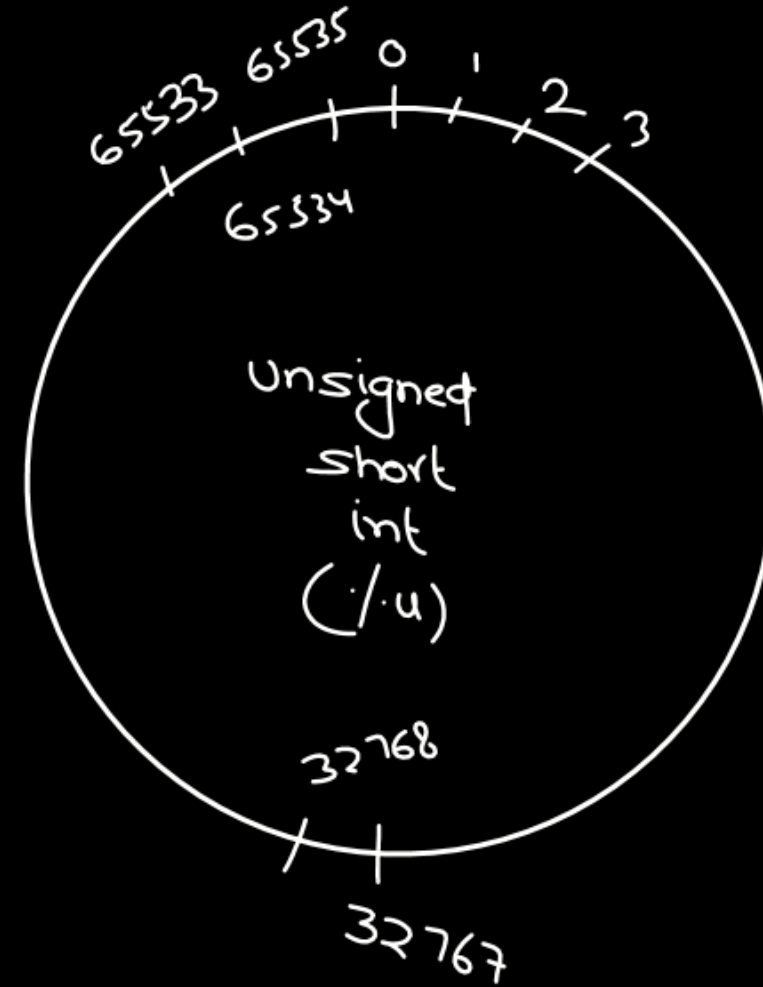
```
printf("%d", i);
```

```
}
```

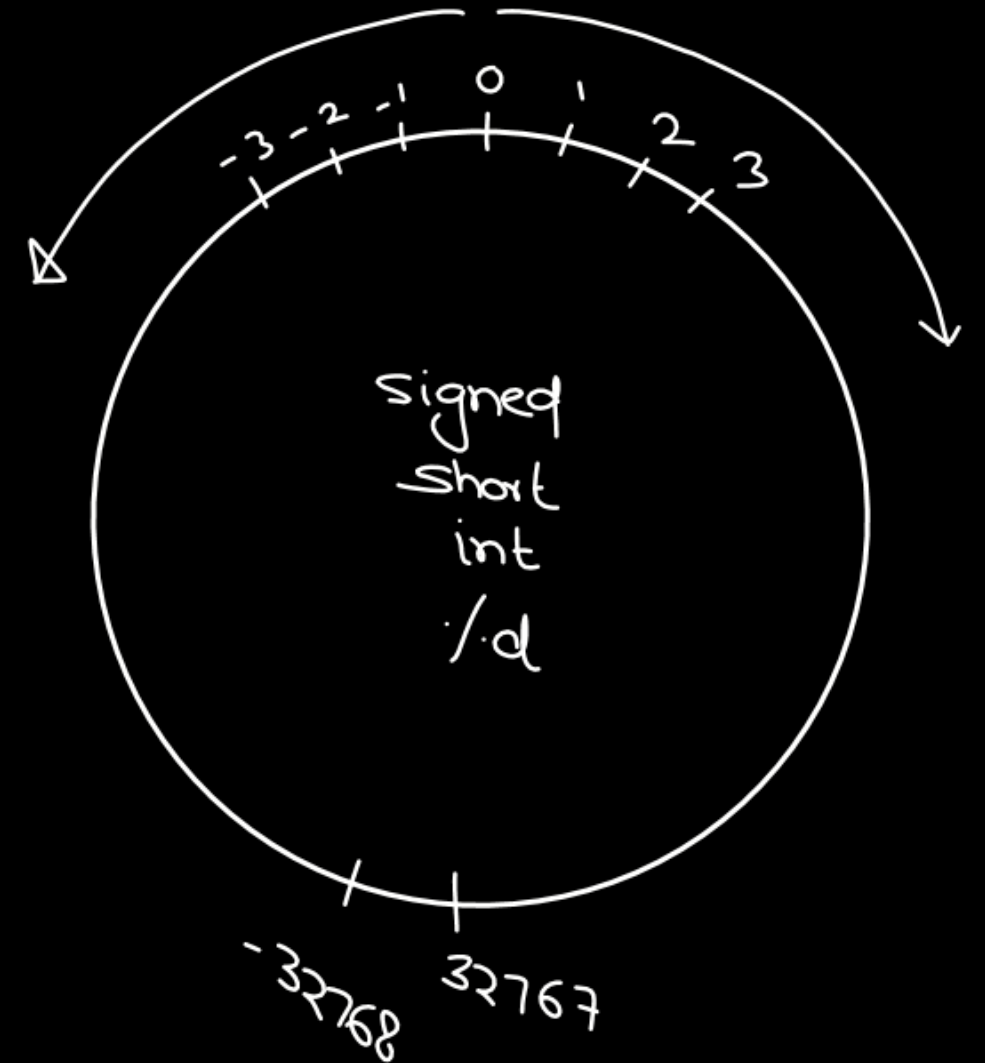
o/p: -4



unsigned 0 to 65535



signed: -32768 to +32767




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

```
void main() {
```

```
unsigned short int i = -2;
```

```
printf("/u", i);
```

```
}
```

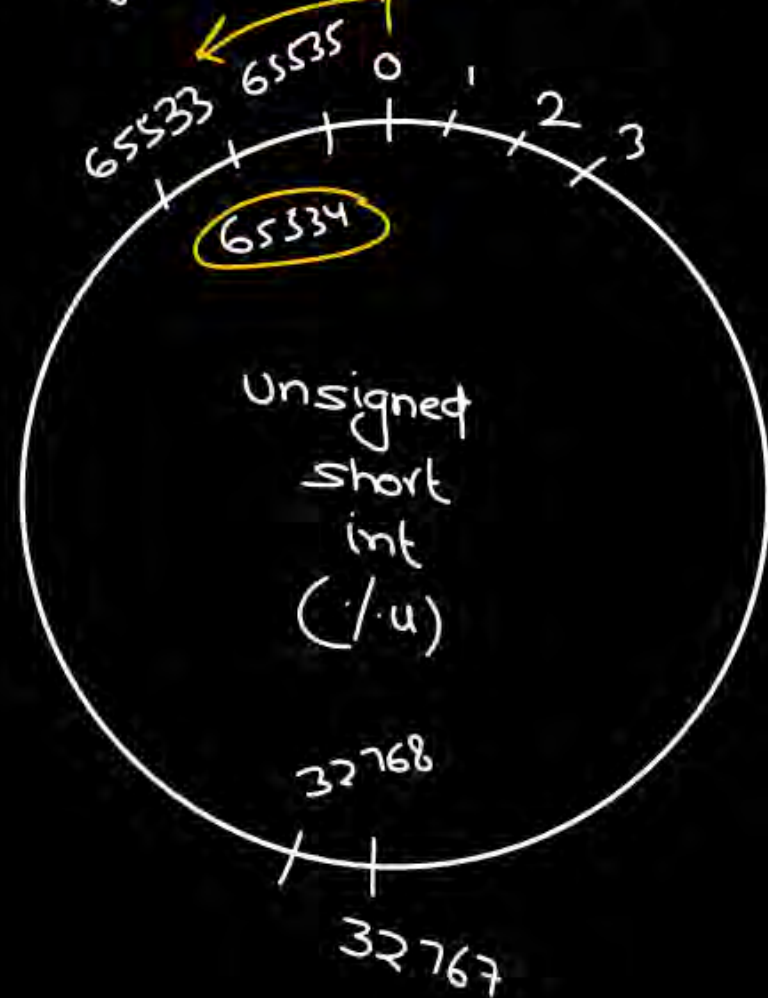
O/P : 65534

+ve value

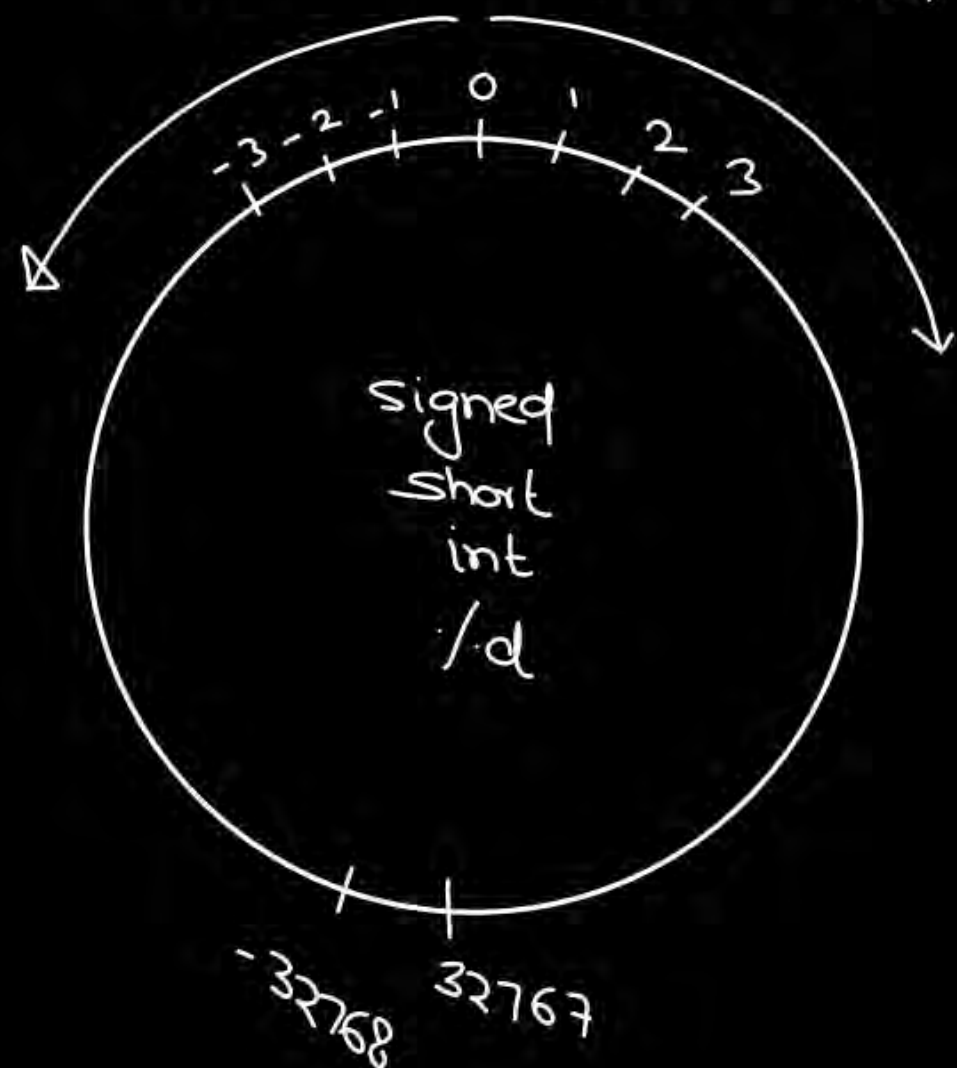
short int \rightarrow 2 byte



unsigned 0 to 65535



signed: -32768 to +32767



```
#include <stdio.h>
void main() {
    short int i = 32769;
    printf("%d", i);
}
```

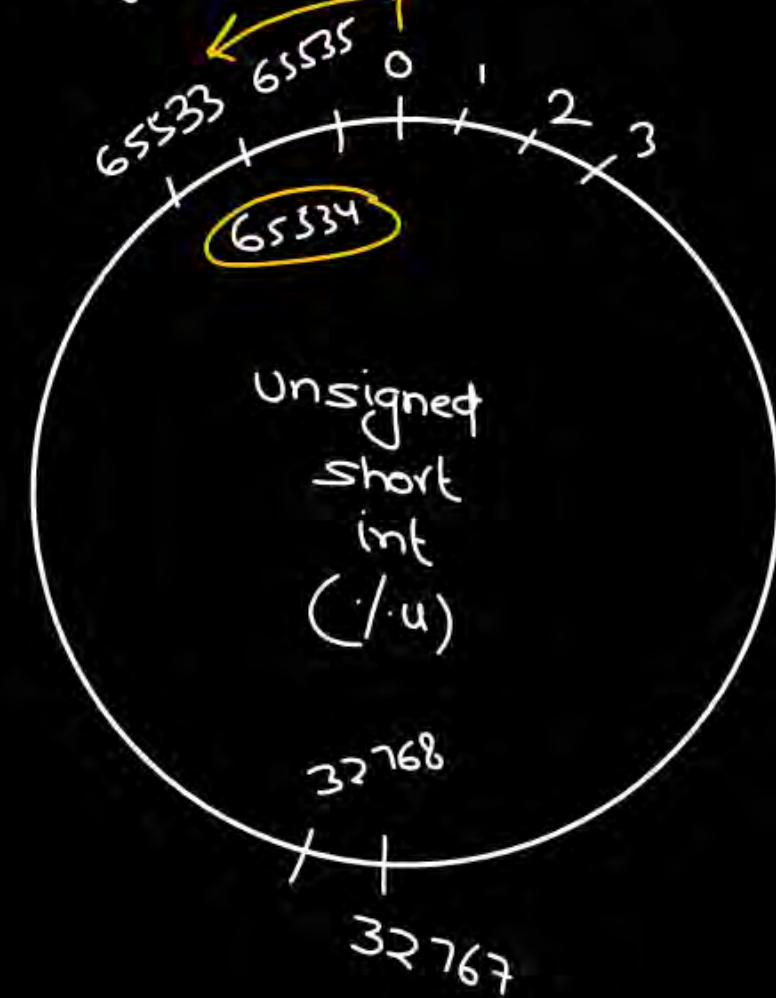
-32767

2016
-32767

short int → 2 byte

→ +ve value

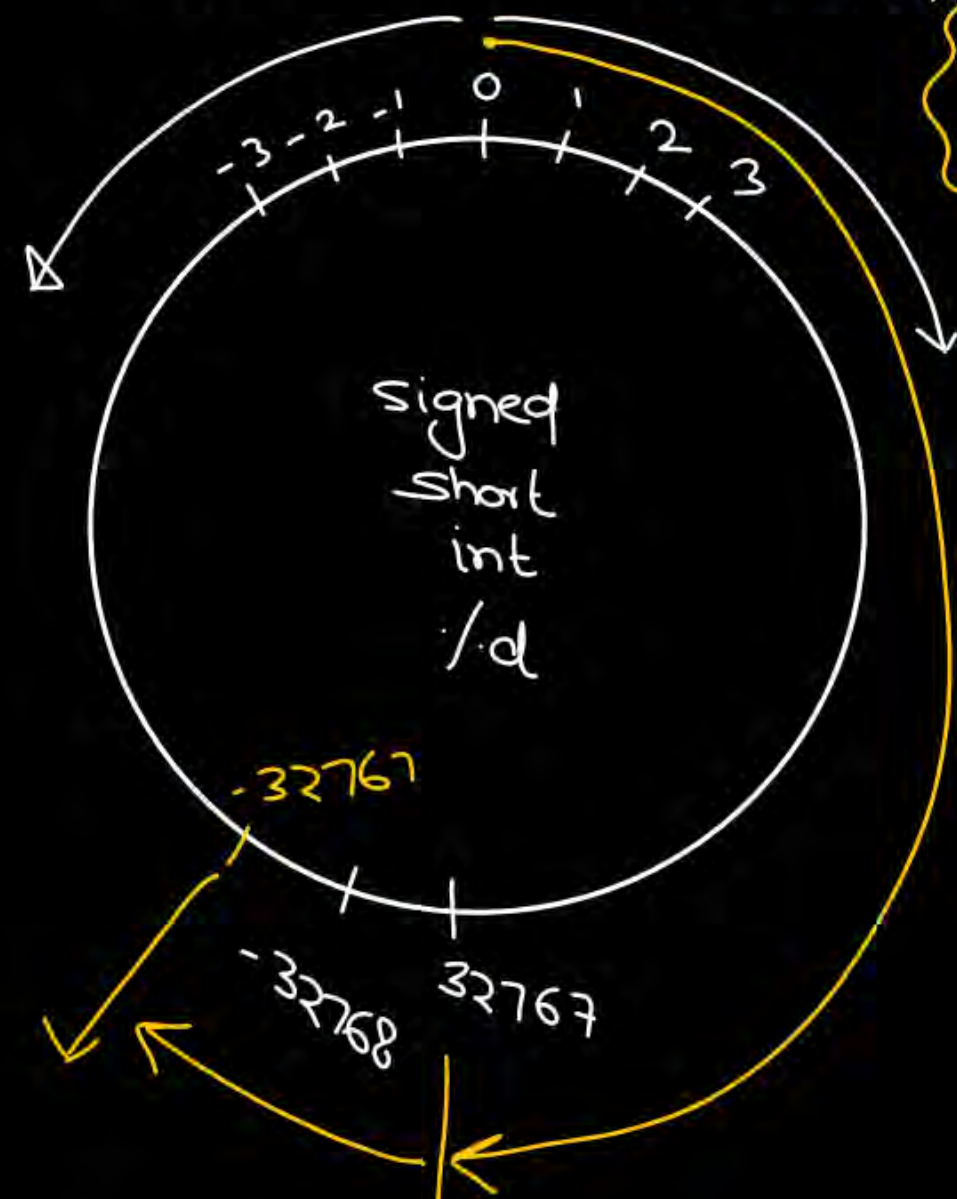
unsigned 0 to 65535



Cyclic
prop

→ short int, int, long int, char

signed: -32768 to +32767



32769 is
2 more than this

short int \rightarrow 2 byte



Cyclic
prop

\rightarrow short int, int,

long int, char

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

```
signed void main() {
```

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

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

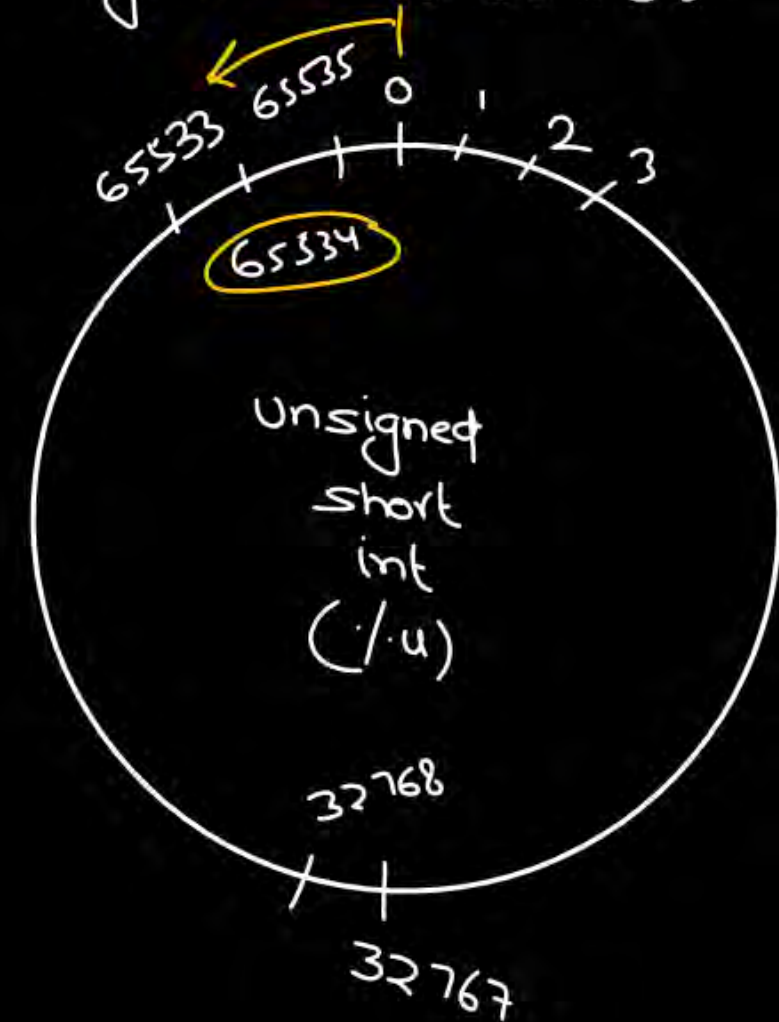
```
}
```

O/P : 32766

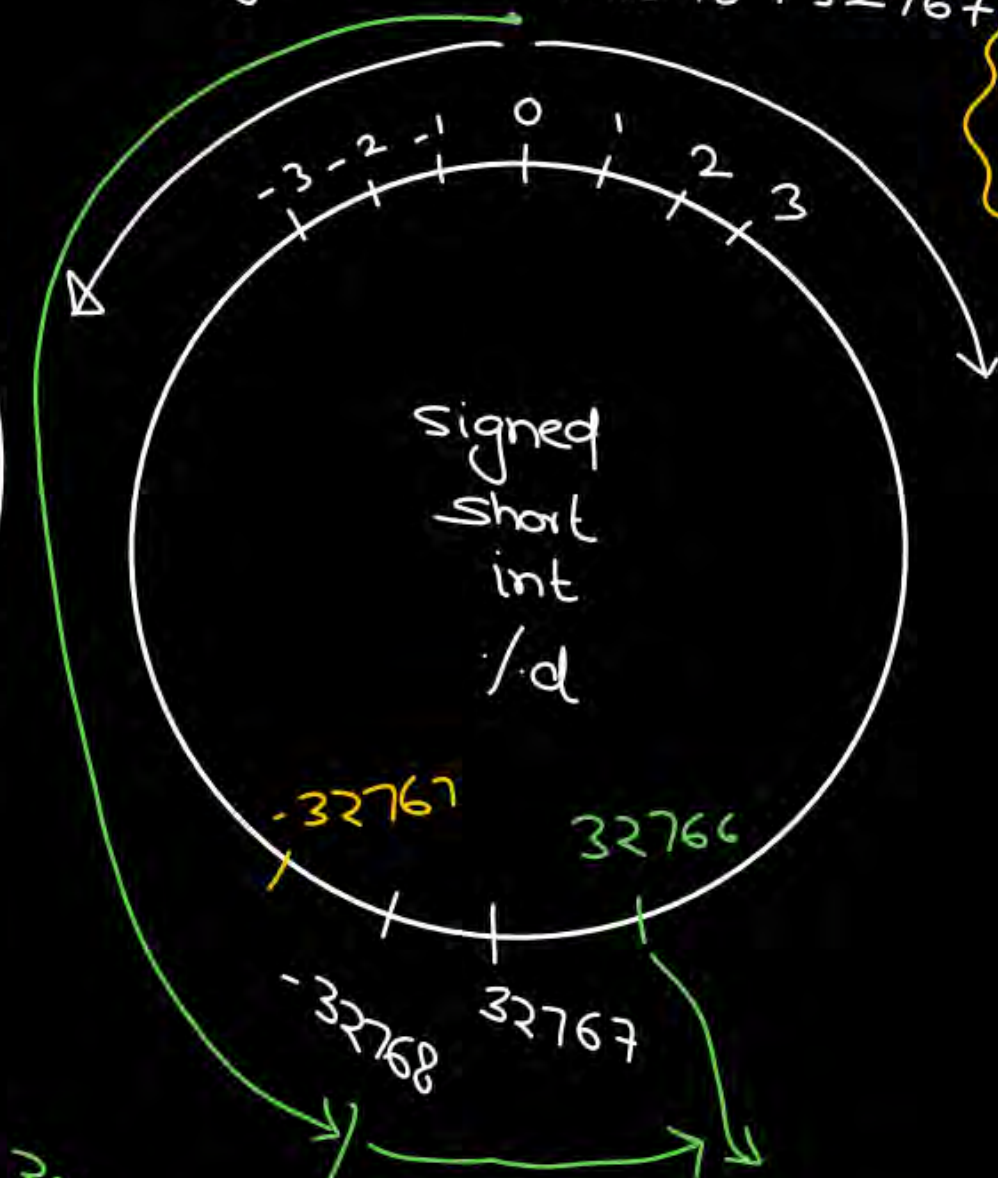


2016

unsigned 0 to 65535



signed: -32768 to +32767



-32770 is
less than
this

short int \rightarrow 2 byte



Cyclic
prop

\rightarrow short int, int,

long int, char

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

signed void main() {

short int i = -32770;

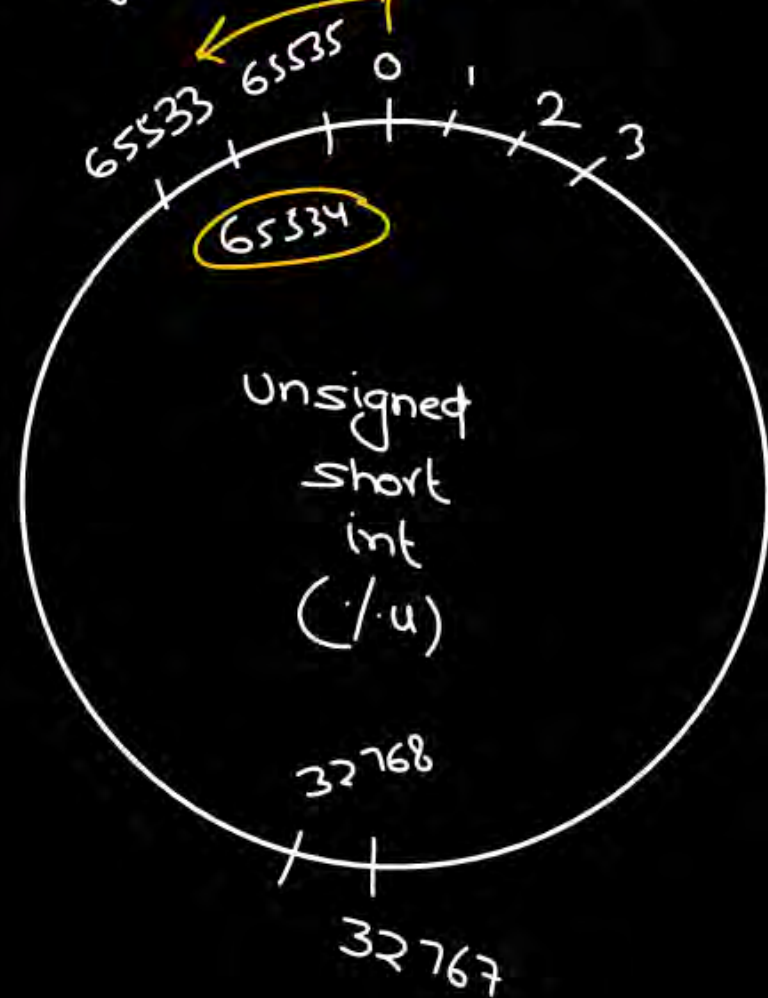
printf("%u", i);

}

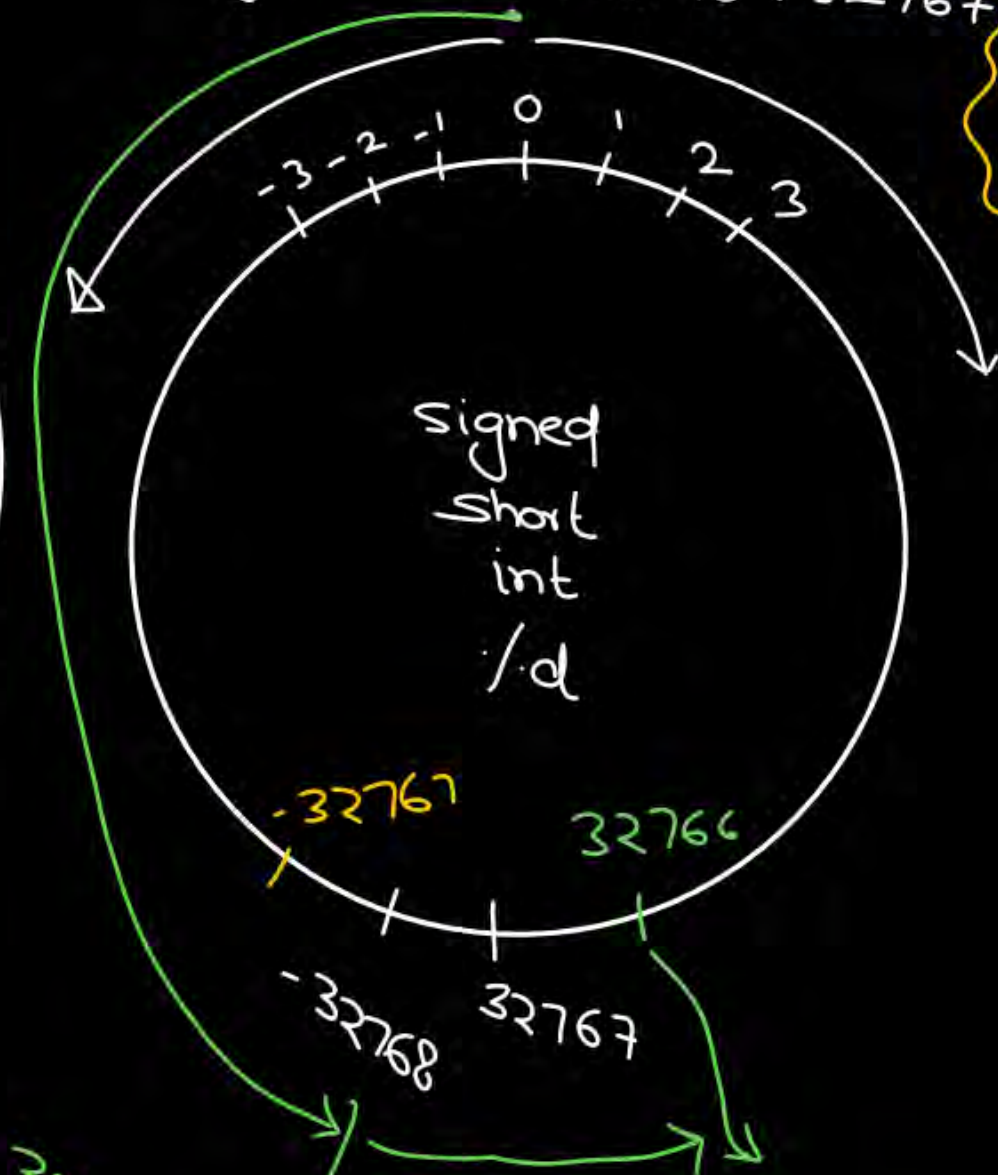
O/P: 32766



unsigned 0 to 65535



signed: -32768 to +32767



-32770 is
less than
this

short int \rightarrow 2 byte



Cyclic
prop

\rightarrow short int,
int,

long
int
char

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

① unsigned short int a = -3;

printf("/u", a); ✓ 65535

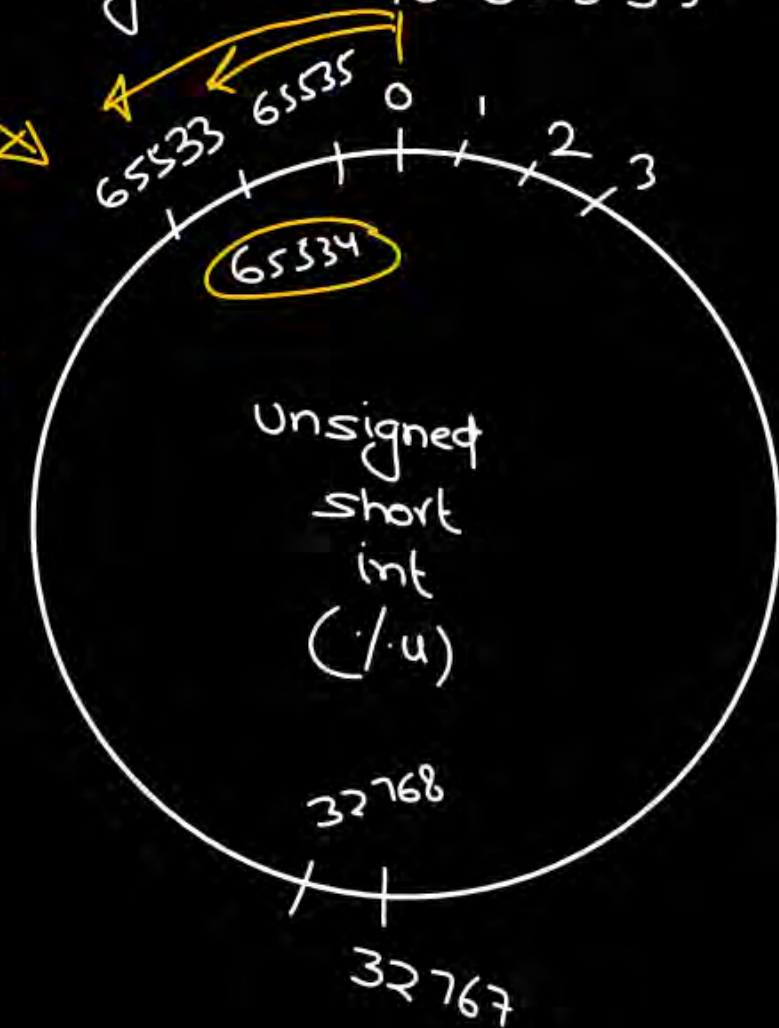
printf("/d", a);

}

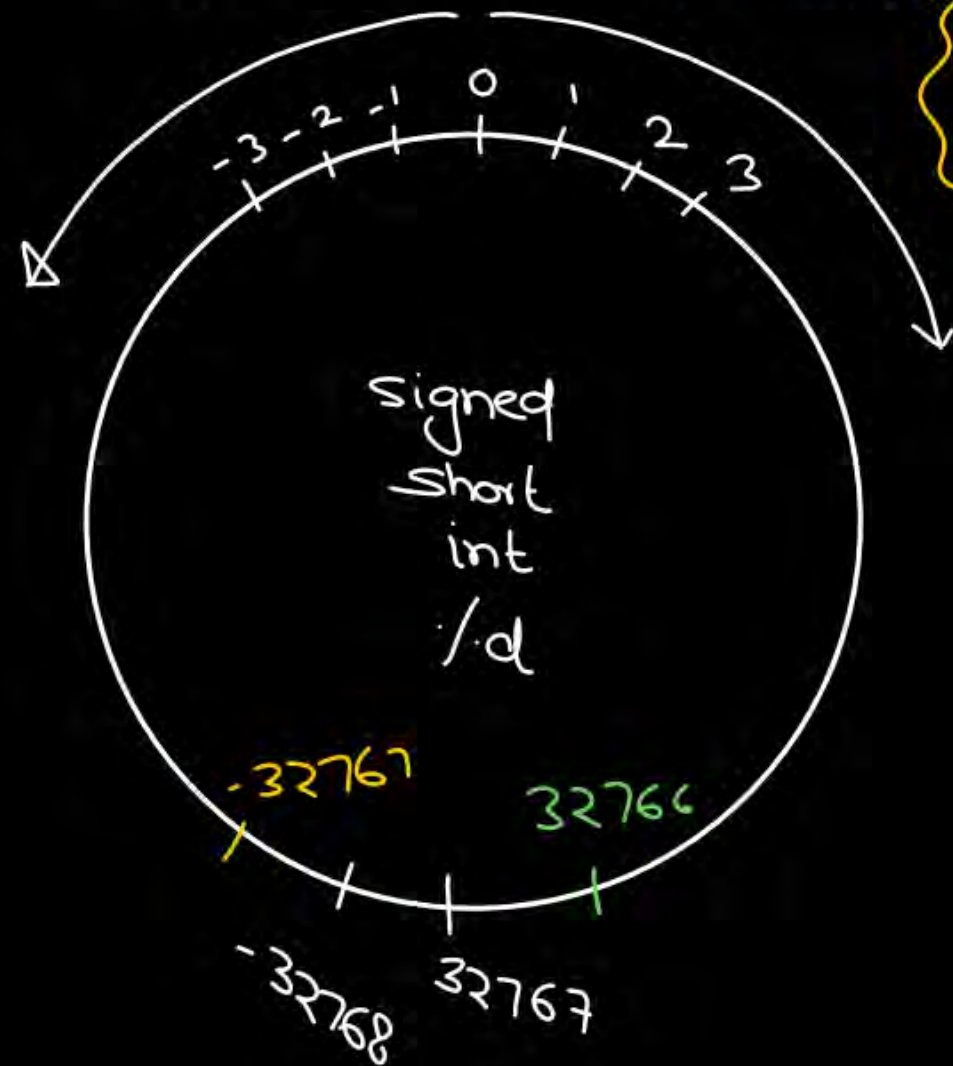
a

65533

unsigned 0 to 65535



signed: -32768 to +32767



short int \rightarrow 2 byte



Cyclic
prop

\rightarrow short int,
int,

long
int
char

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

① unsigned short int a = (-3);

printf("/u", a); ✓ 65535

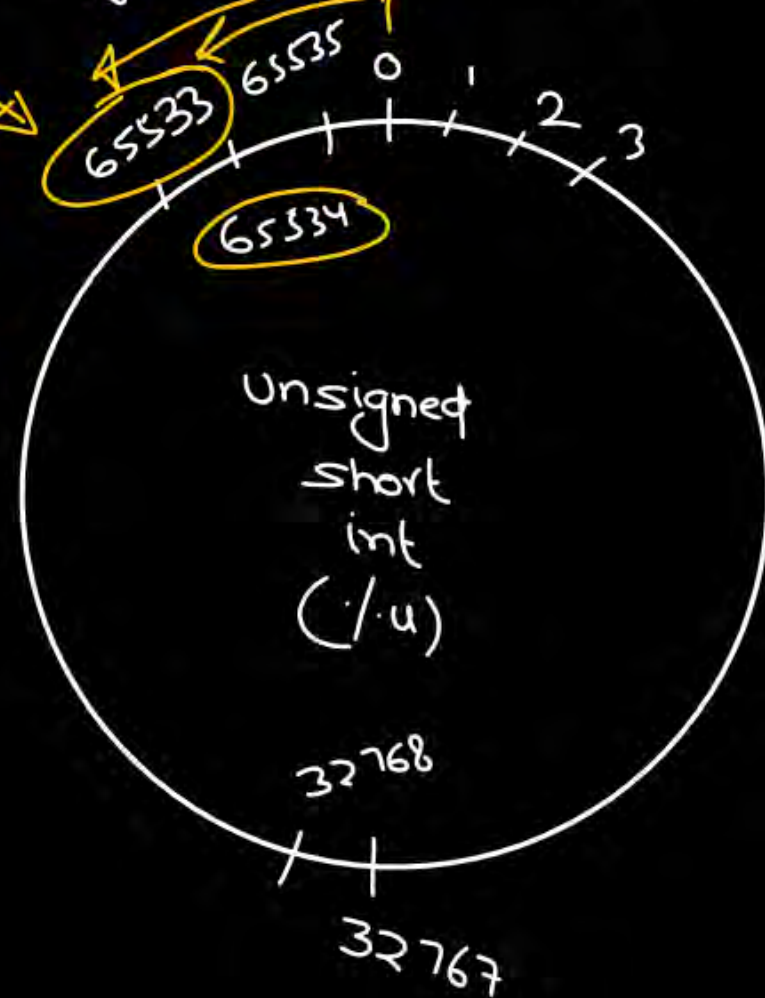
printf("/d", a); -3

}

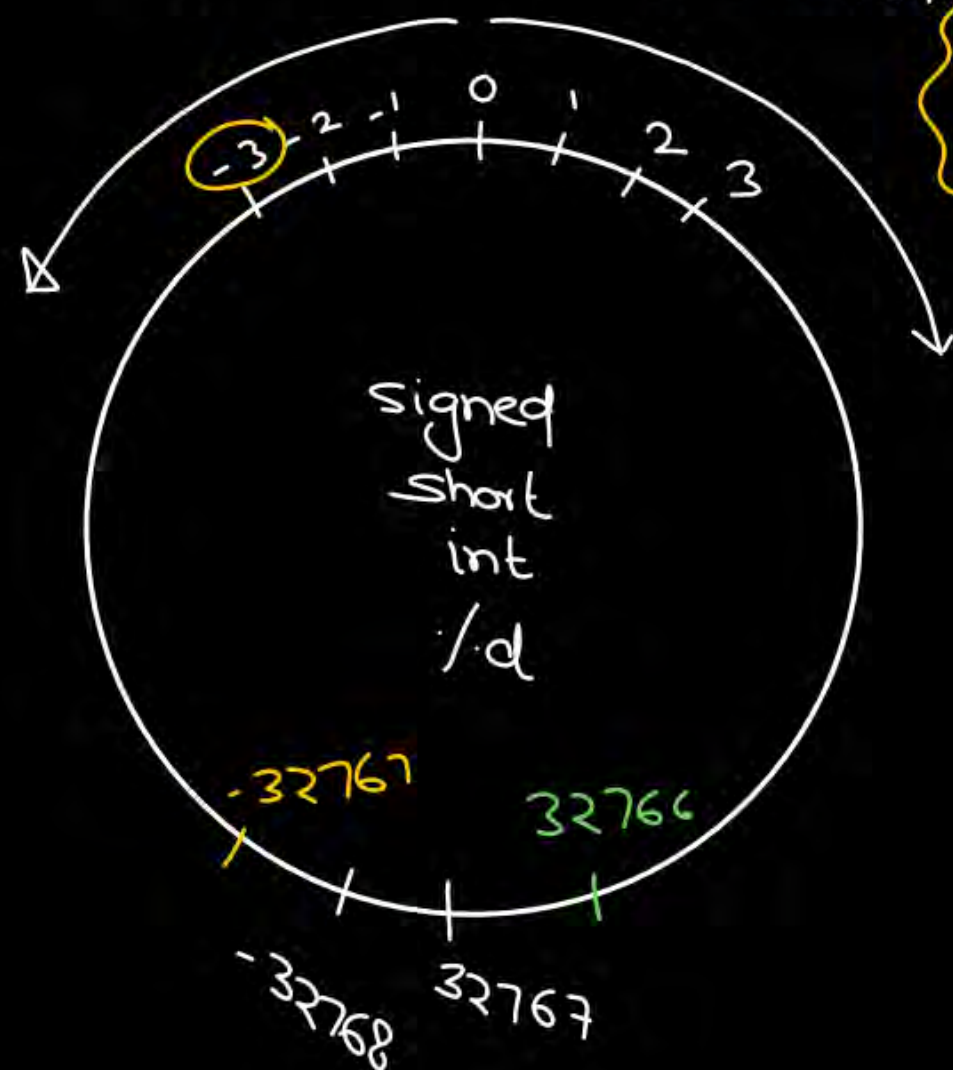
a

65533

unsigned 0 to 65535



signed: -32768 to +32767



short int \rightarrow 2 byte



Cyclic
prop

\rightarrow short int, int,

long int, char

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

short int a = -3

```
printf("/u", a);
```

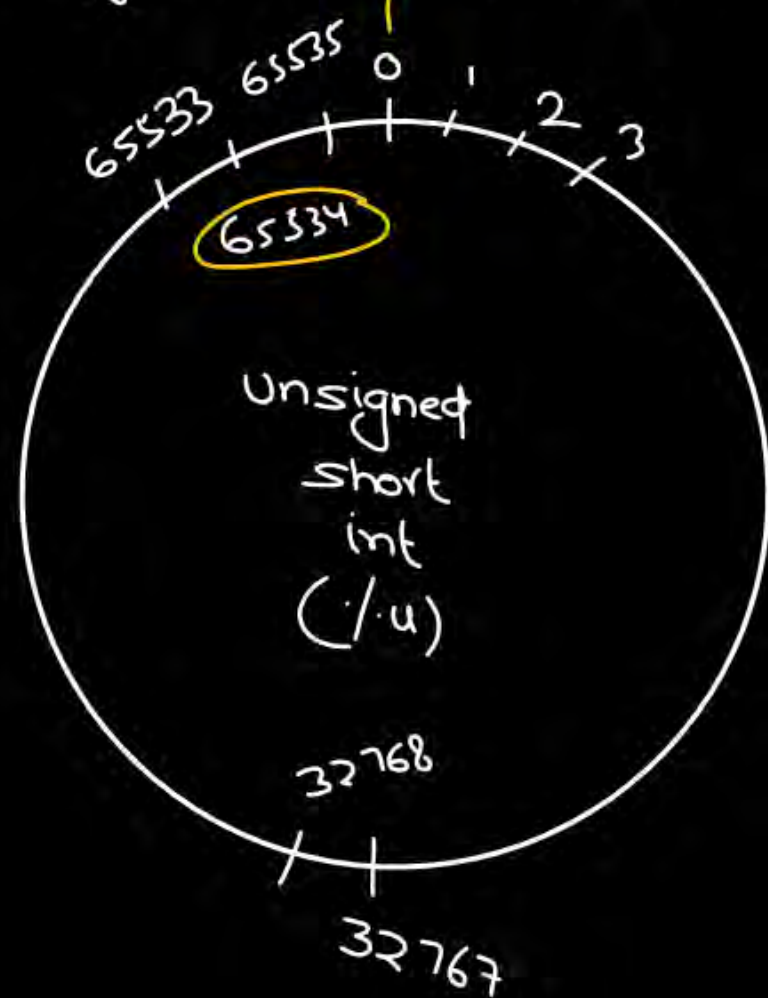
```
printf("/d", a);
```

```
}
```

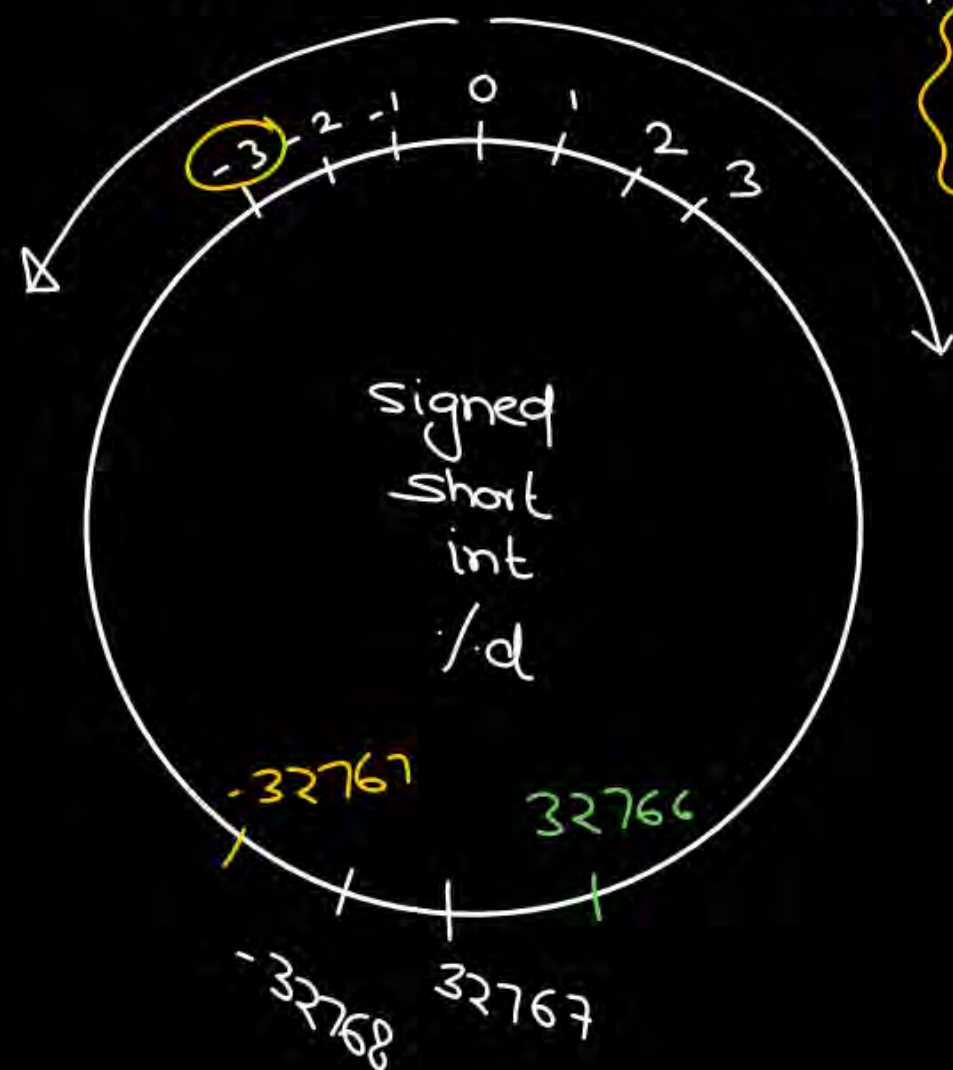
a



unsigned 0 to 65535

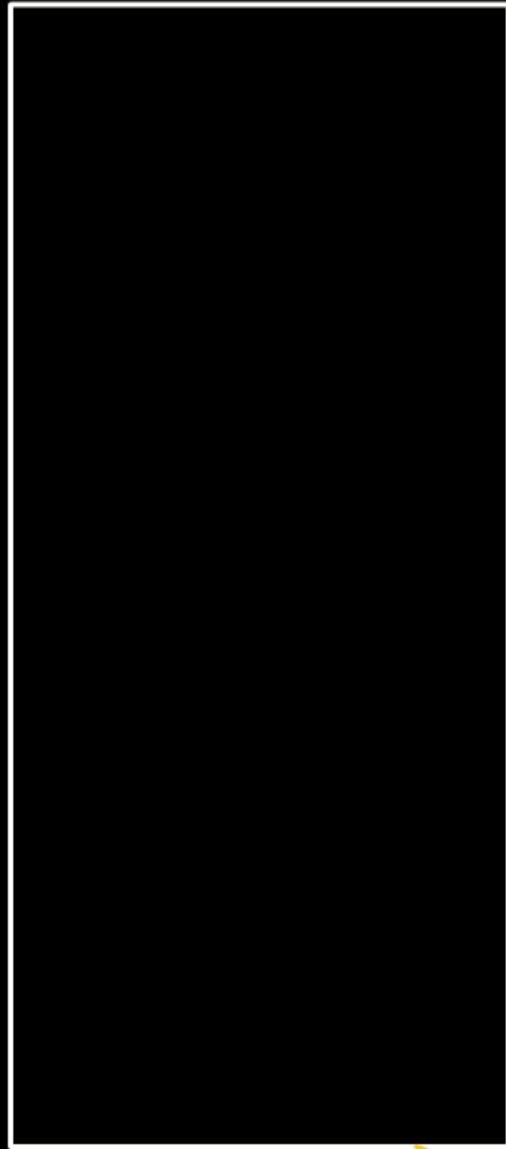


signed: -32768 to +32767



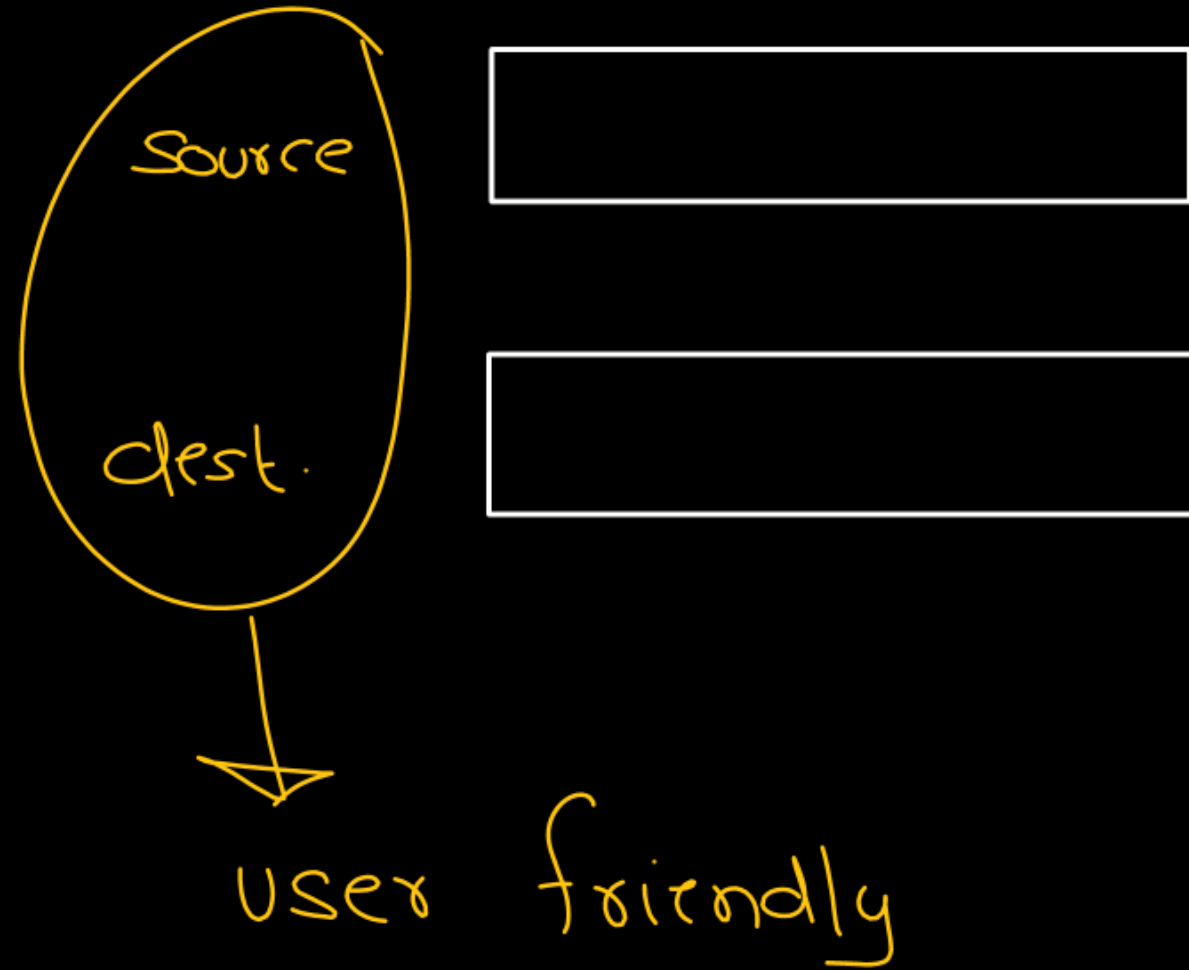
① format specifier

How to take a i/p from keyboard (scanf)



0, 1, 2

✓ A - [0|000000|] - 65 ✓

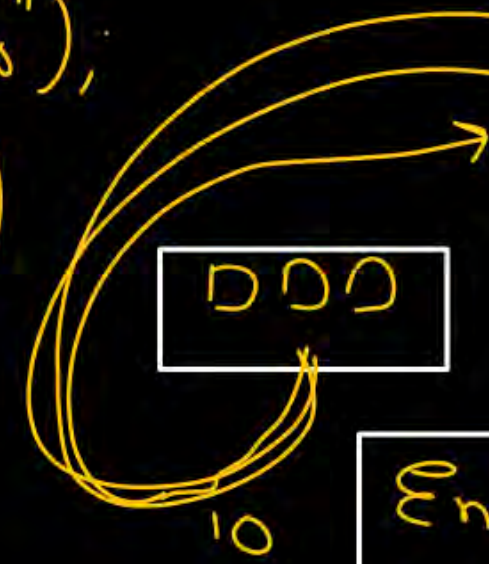


```
#include <stdio.h>
void main() {
```

```
    int a ;
```

```
    printf("Enter a number");
```

```
    scanf("/d", address)
```



Enter a number 10

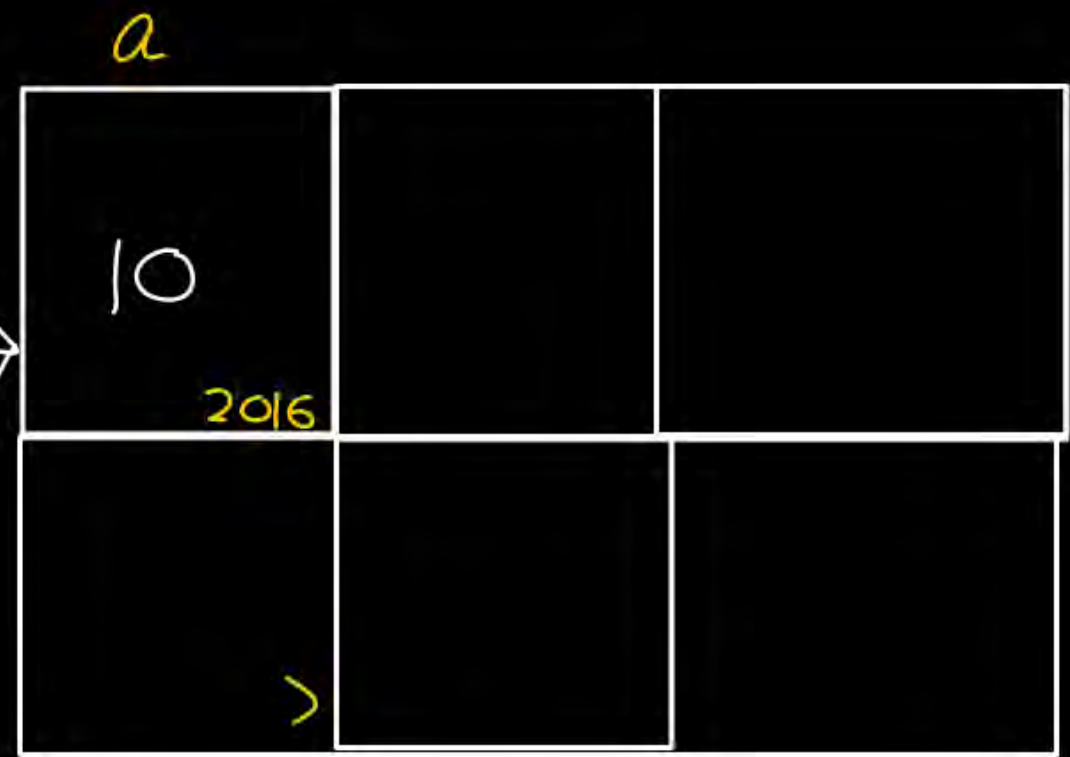
#include <stdio.h>
void main() {

int a;

printf("Enter a number");
scanf("%d", &a);

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

&



000

10

Enter a number 10
The value is 10

28 lect → 28 question
1st sem
printer
↓
format

Addition of 2 numbers Chapter

Take 2 no. from user

print their sum

a
10
~~5~~
2016

b
20
~~5~~
3048
c

30
~~5~~
1086

printf ("Enter First number");

scanf ("%d", &a);

printf ("Enter second number");

scanf ("%d", &b);

c = a + b;

printf ("The sum of %d and %d is %d", a, b, c);

Enter First number 10
Enter second number 20
The sum of 10 and 20 is 30

Enter

Char
↓

1 byte

8 bits

2^8 possible values

256

$\overset{1}{\text{Char}} \rightarrow \text{Symbol}$

Char c = '@';

Signed char

256

128

128

-128, ..., -1

0 to 127

-128 to +127

unsigned char

0 to 255

```
void main() {
```

```
  int a = 10;
```

```
}
```

10 → 00001010

No.

No.

01010100

00100000

00001010

01- -

1- -

CHARACTER SYSTEM

Language

character
(symbol)

using a

Character
System, we represent
symbols of a lang
by the integer
Constant

M/C

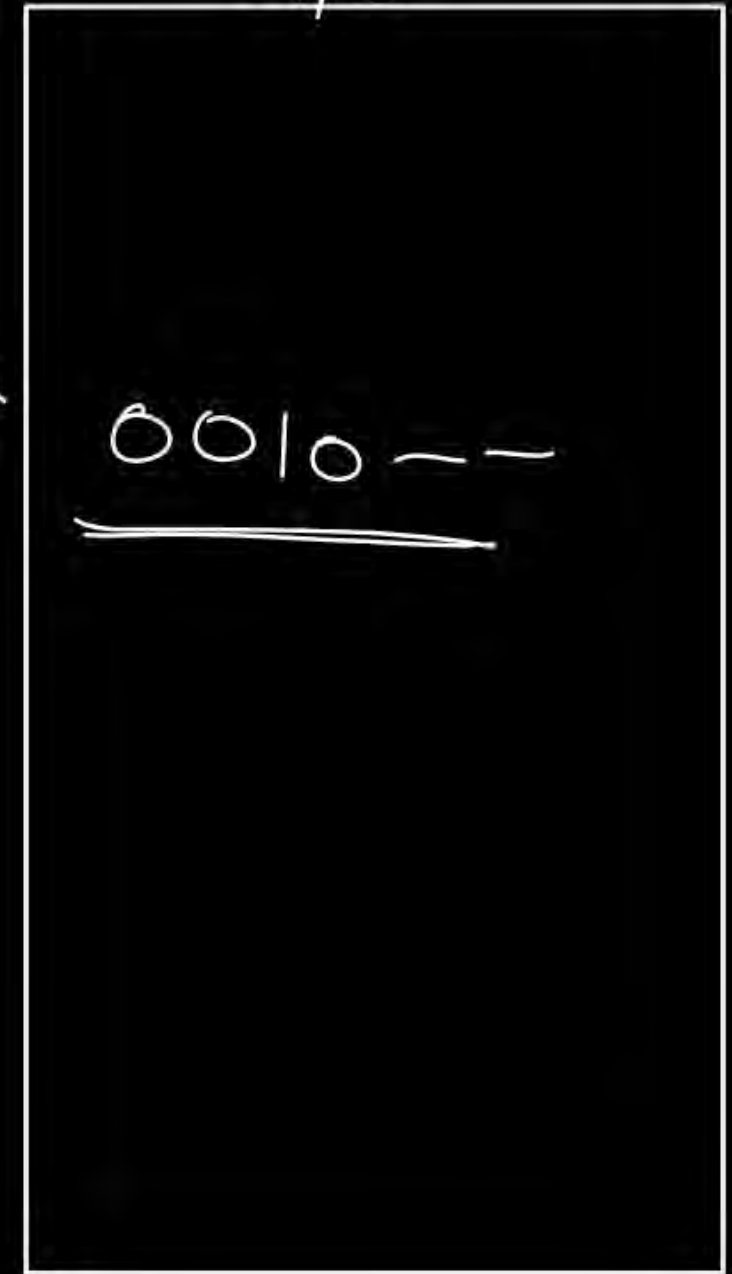
CHARACTER SYSTEM

English type

ASCII

M/C

Language



@

one form of
information



language → Symbols

A — 65	a — 97	0 — 48	@ —
B — 66	b — 98	1 — 49	# —
C — 67	c — 99	.	
.	.		
	.		
	.		
Z — 90	z — 122		

256

1 byte

Java

65536

UNICODE

ASCII

by default signed
char c = 65;



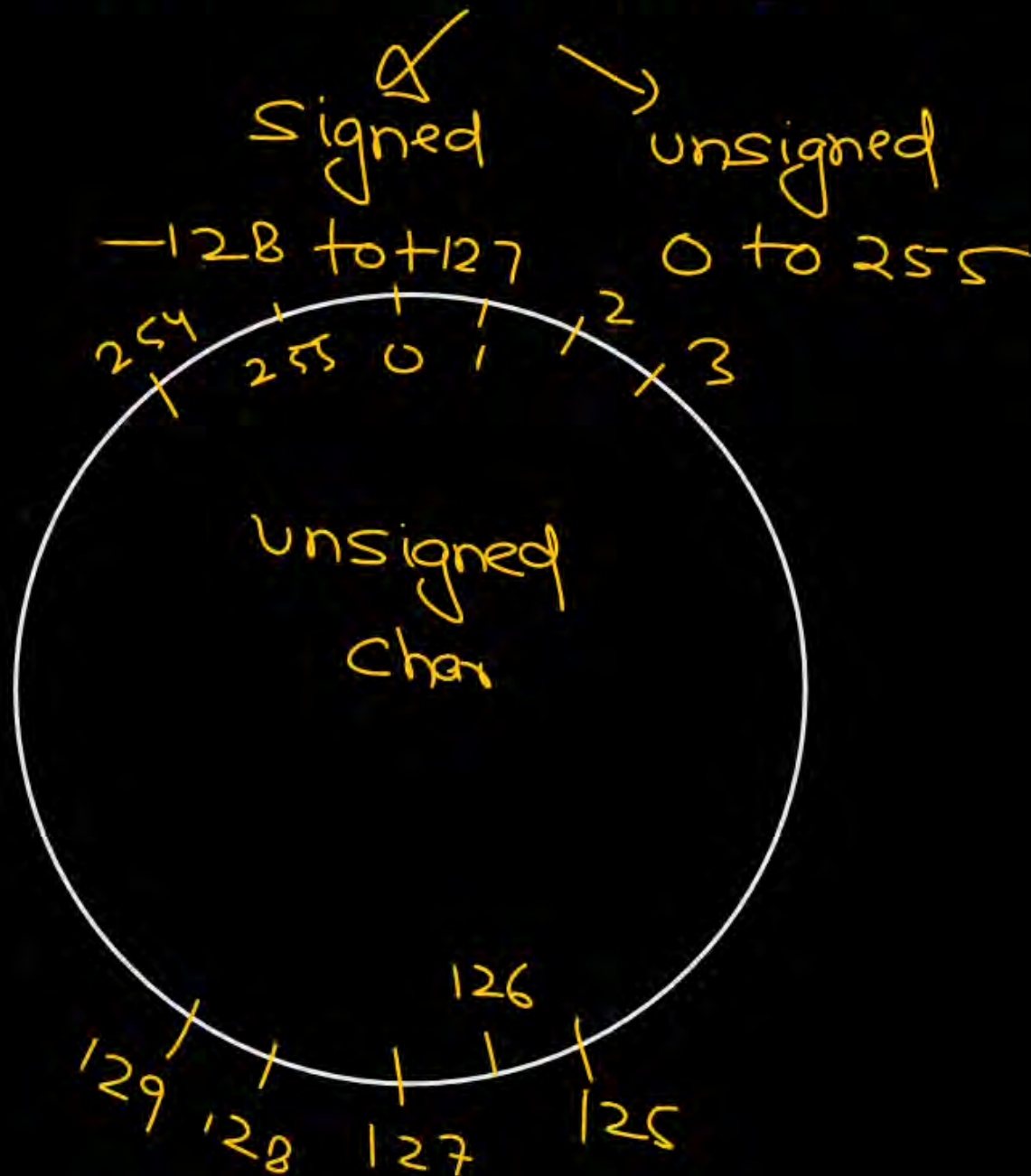
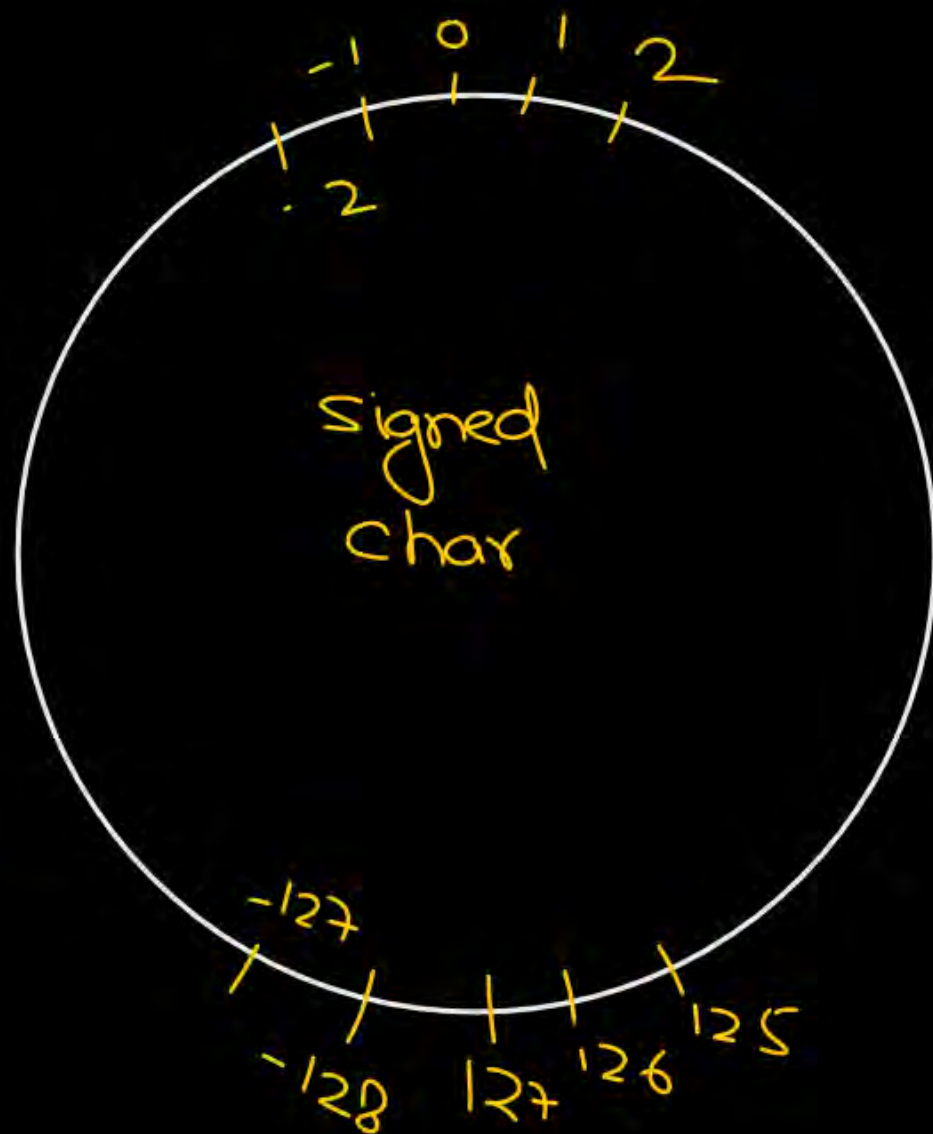
printf("%d", c);

signed integer
short

65

/c → character System

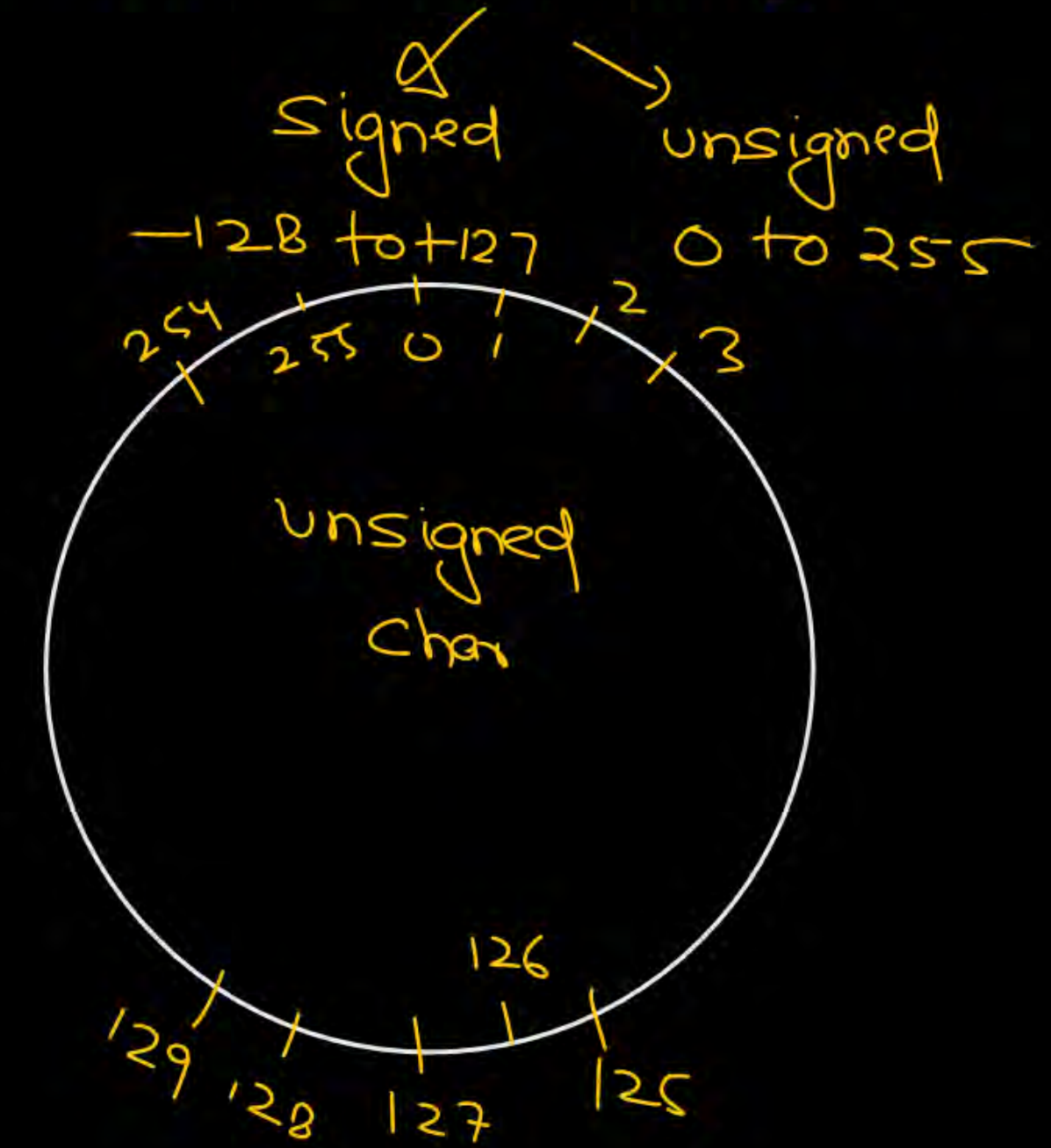
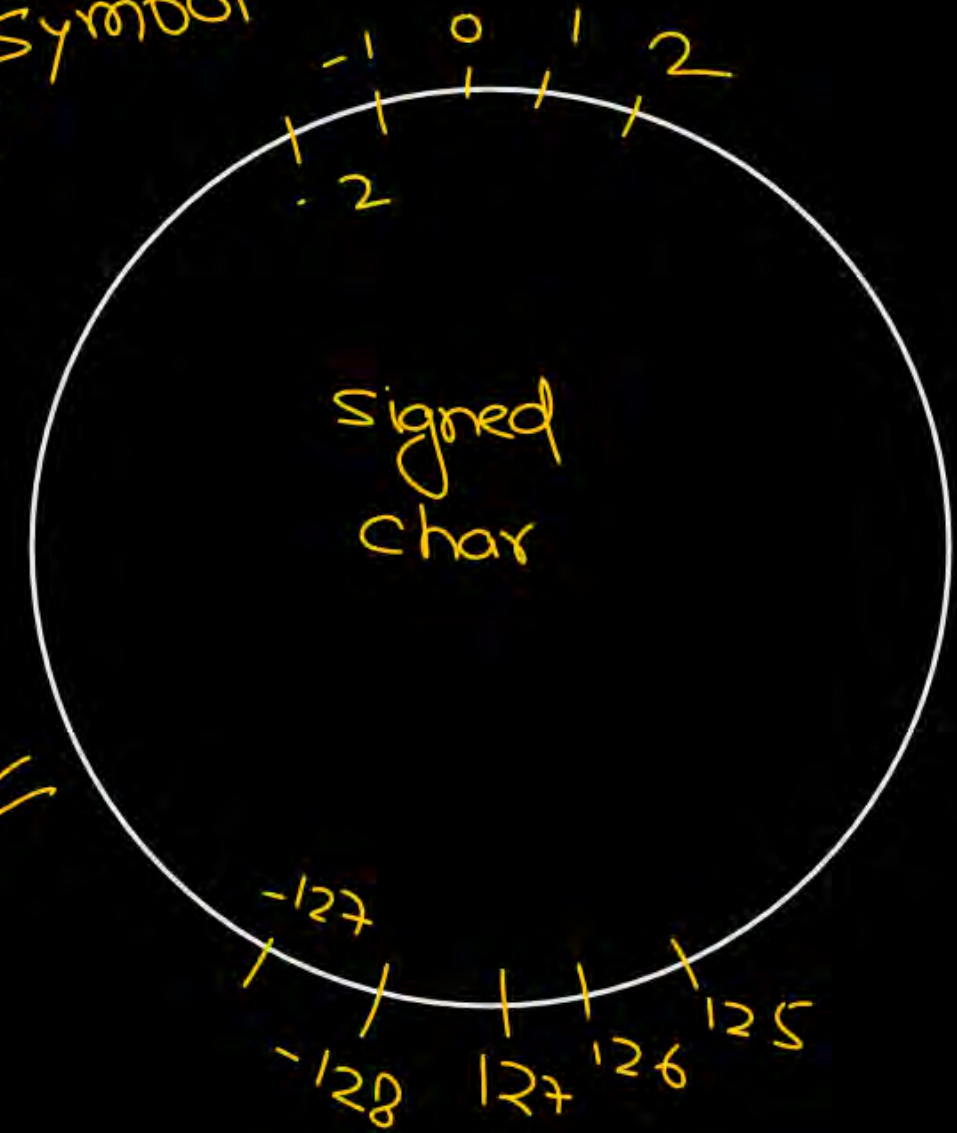
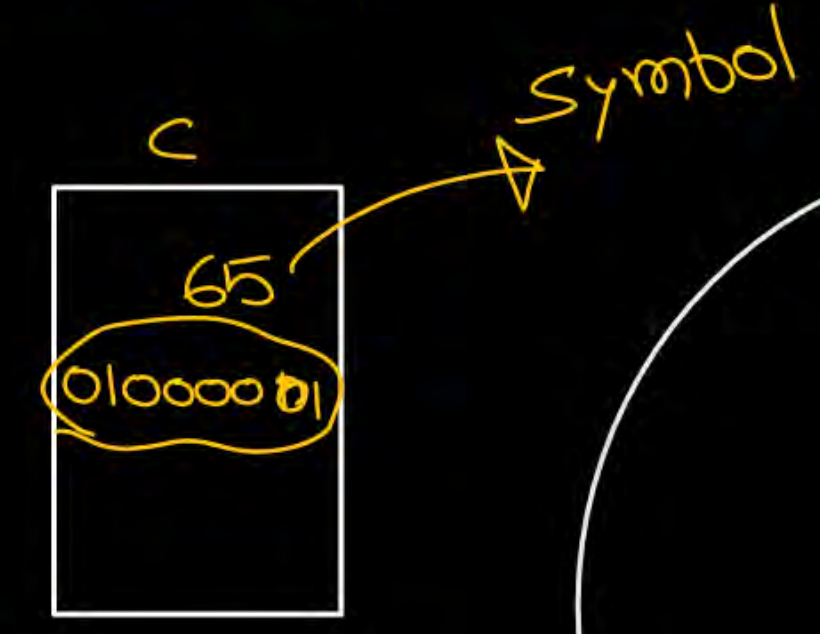
char - 1 byte - 8 bits - 256 values



/c → character System

char - 1 byte - 8 bits - 256 values

by default signed
char c = 65;



printf("/c", c); All
Character System

/c → character System

char - 1 byte - 8 bits - 256 values

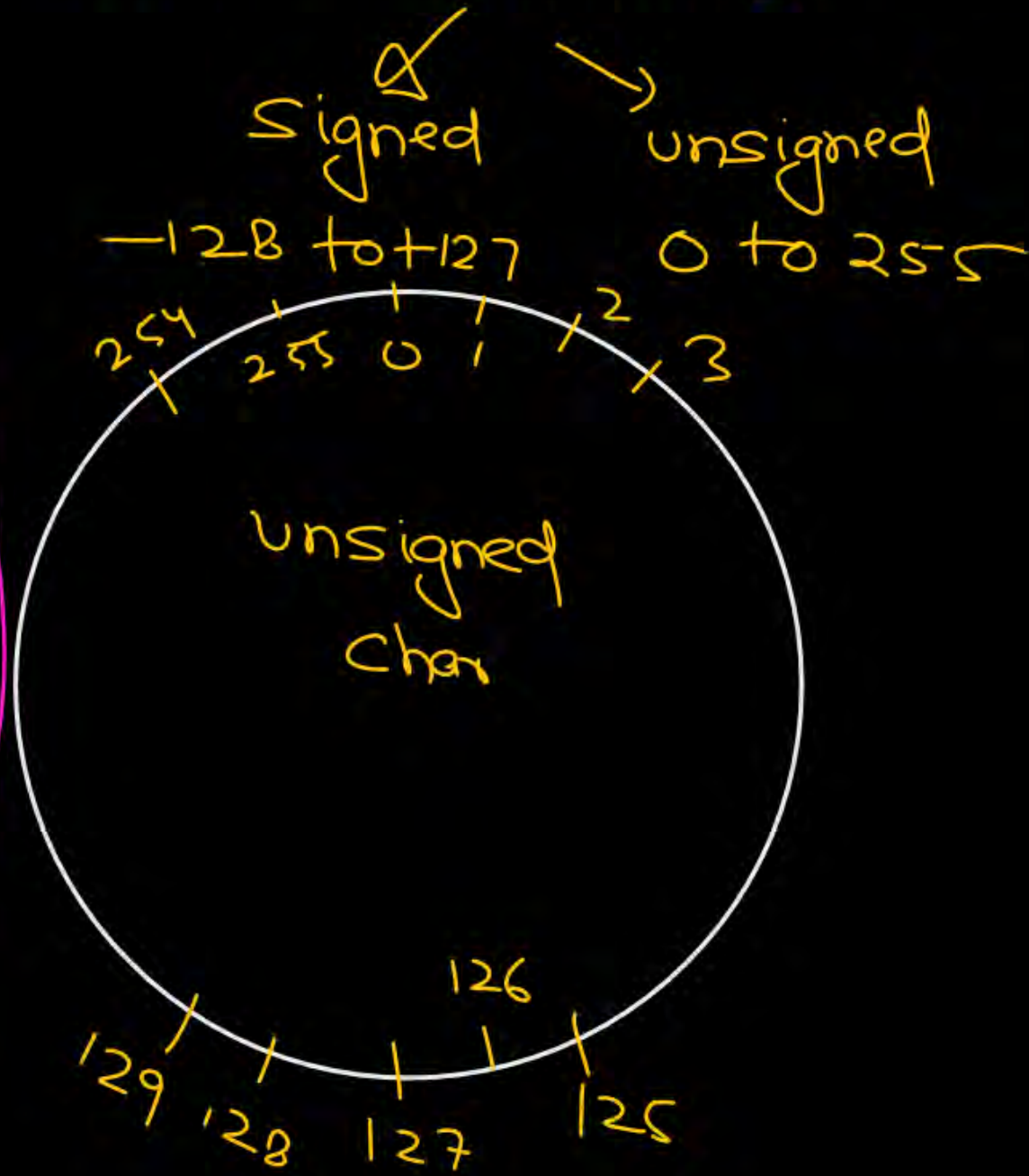
char c = 128;

printf("%d", c);

signed integer

-128

-128



/c → character System

char - 1 byte - 8 bits - 256 values

char c = -130 ;

printf("%d", c);

signed integer

