



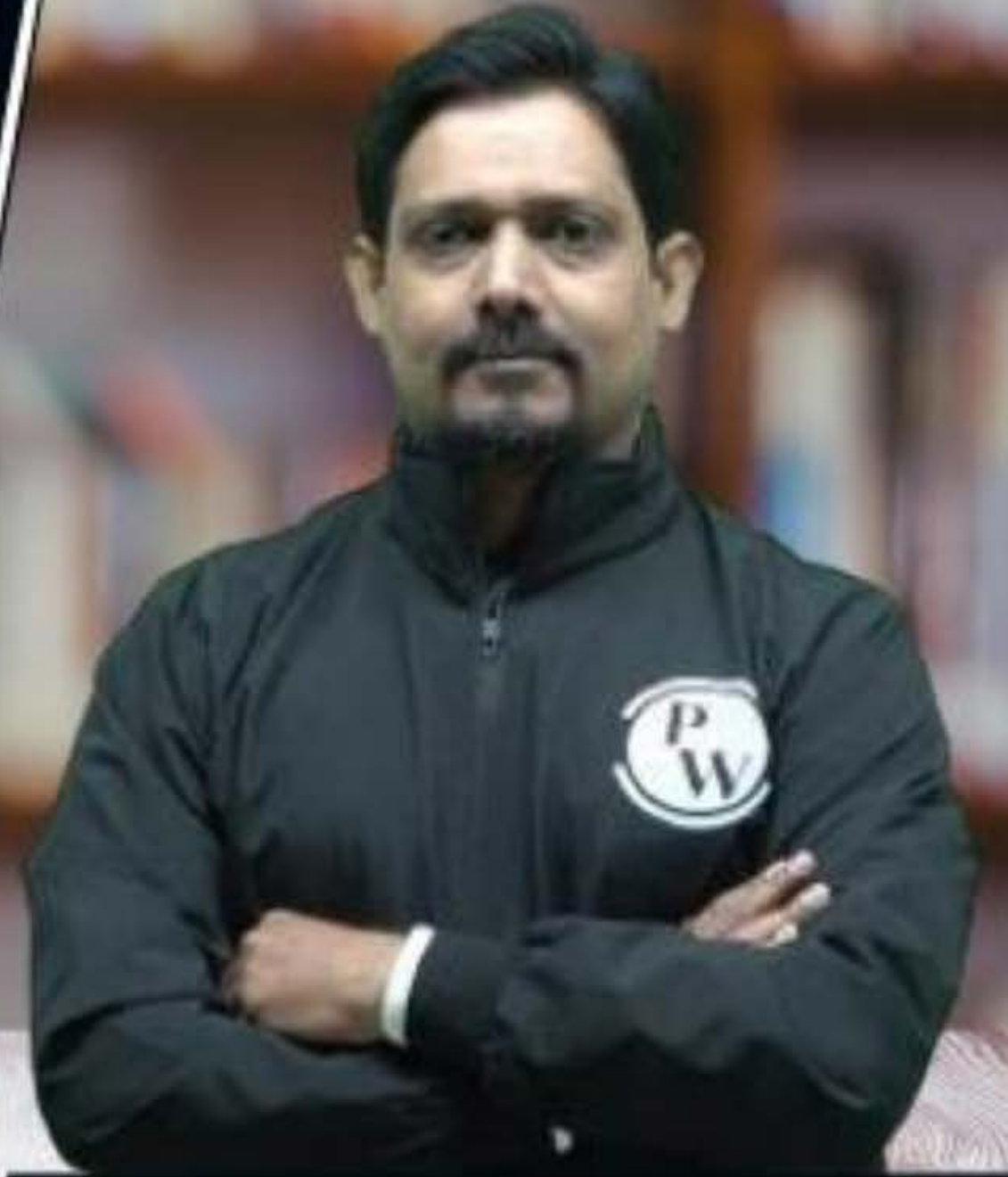
Computer Science & IT

C programming



Data Types & Operators

Lecture No. 02



By- Abhishek Sir

Recap of Previous Lecture



topic

Range of Signed

topic

Range of unsigned

topic

practice

topic

topic

topics to be Covered



topic

Data types

topic

Singed No. (prachice)

topic

topic

topic



Topic : Data Type

primary Data type

- Integer
- char
- floating point
Real No.

User Define

- structure
- union
- enum
- typedef

Derived

- function
- Array
- pointer





Topic : Data Type



Integer — Singed Integer $n \text{ bit} - -2^{n-1} \text{ — } 2^{n-1} - 1$

unsigned Integer $\xrightarrow{\text{'ld'}}$ long int $0 \text{ — } 2^n - 1$
8B 64 Bits

%d int — 4B (32 bits)

%u unsigned — 4B (32 bits)

%hd short int — 2B

%hu short unsigned int — 2B

%lu long unsigned int 8B = 64 bits



Topic : Data Type



Signed cyclic property



4 bits

0111

, 1000 \leftarrow (-8)

$$0 - 2^{n-1} - 1$$

$$0 - 2^3 - 1 = \underline{-7}$$

Singed

$a = 8; \rightarrow \text{print}(-8)$

$a = 10; \rightarrow \text{print}(\underline{-6})$

$a = \textcircled{12} \rightarrow \text{print}(-4)$



Topic : Data Type



Signed cyclic property



if No. goes out of positive No. range then No. is Negative

if No. is k

$k=8$

$$\boxed{-(2^4 - k)} \quad (\text{I}) \quad -(2^4 - 8) \\ -(16 - 8) = -8$$

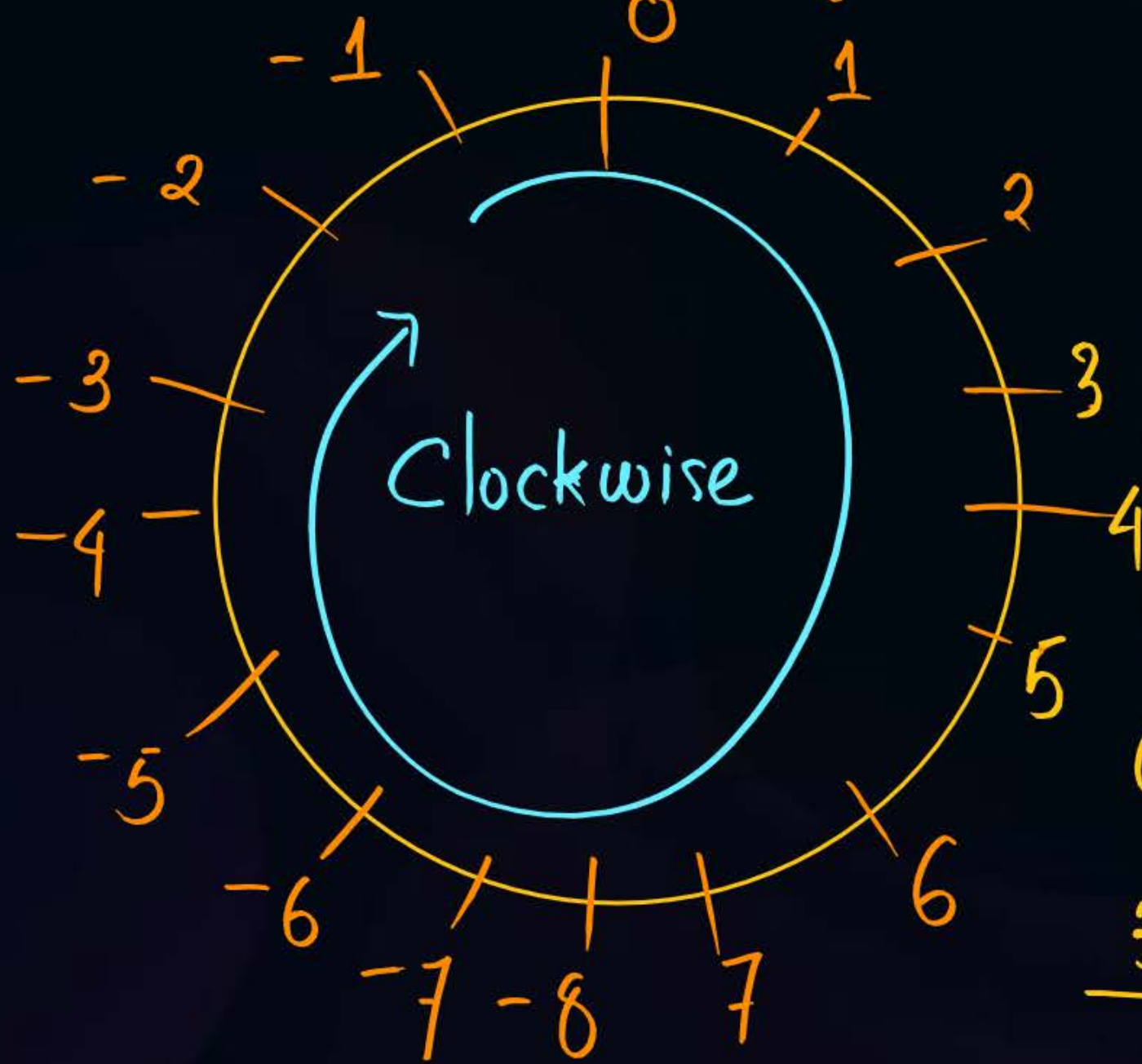
$$(\text{II}) \quad k=10 \Rightarrow -(2^4 - 10) = -(16 - 10) = -6$$

$$(\text{III}) \quad k=12 \Rightarrow -(2^4 - 12) = -(16 - 12) = -4$$



Topic : Data Type

Signed cyclic property



Q. 17 Signed No
if No. K is given

$$K \% 2^4 = 17 \% 16 = \textcircled{1}$$

1 in Range of
positive No

17 Signed value 1
Signed value 4 bit will be?

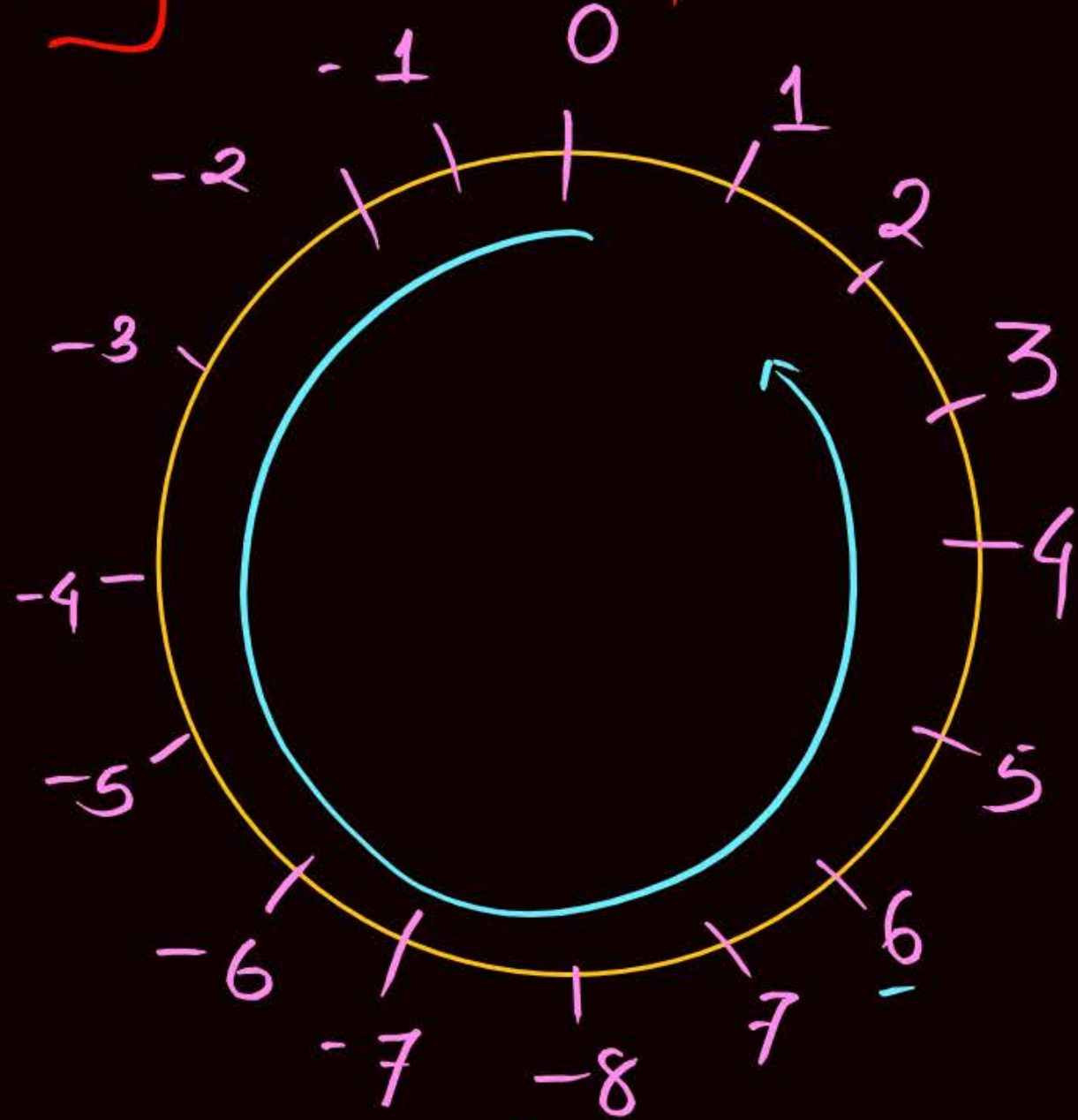
Q. $K=32$
 $\underline{32 \% 16 = 0}$

Q. $K=28$ Signed value in 4 bits representation.

1. Take modulo $\frac{28}{16}$ Q 1 R 12 ← modulo — %
returns Remainder

No 12 Not in Range $-(2^4 - 12) = -(16 - 12) = \underline{-4}$
Ans.

Negative No. overflow



Signed number (4bit)

(I) $k = -5$

Signed value = -5

(II) $k = -10$

Signed value will be
Negative No. out of range

Signed value is 6

if overflow occurs

(III) $k = -15 = (2^4 - 15) = \underline{1}$

$$(2^4 - |k|) = (16 - 10) = \underline{6}$$

Signed No. unsigned value (Negative No)

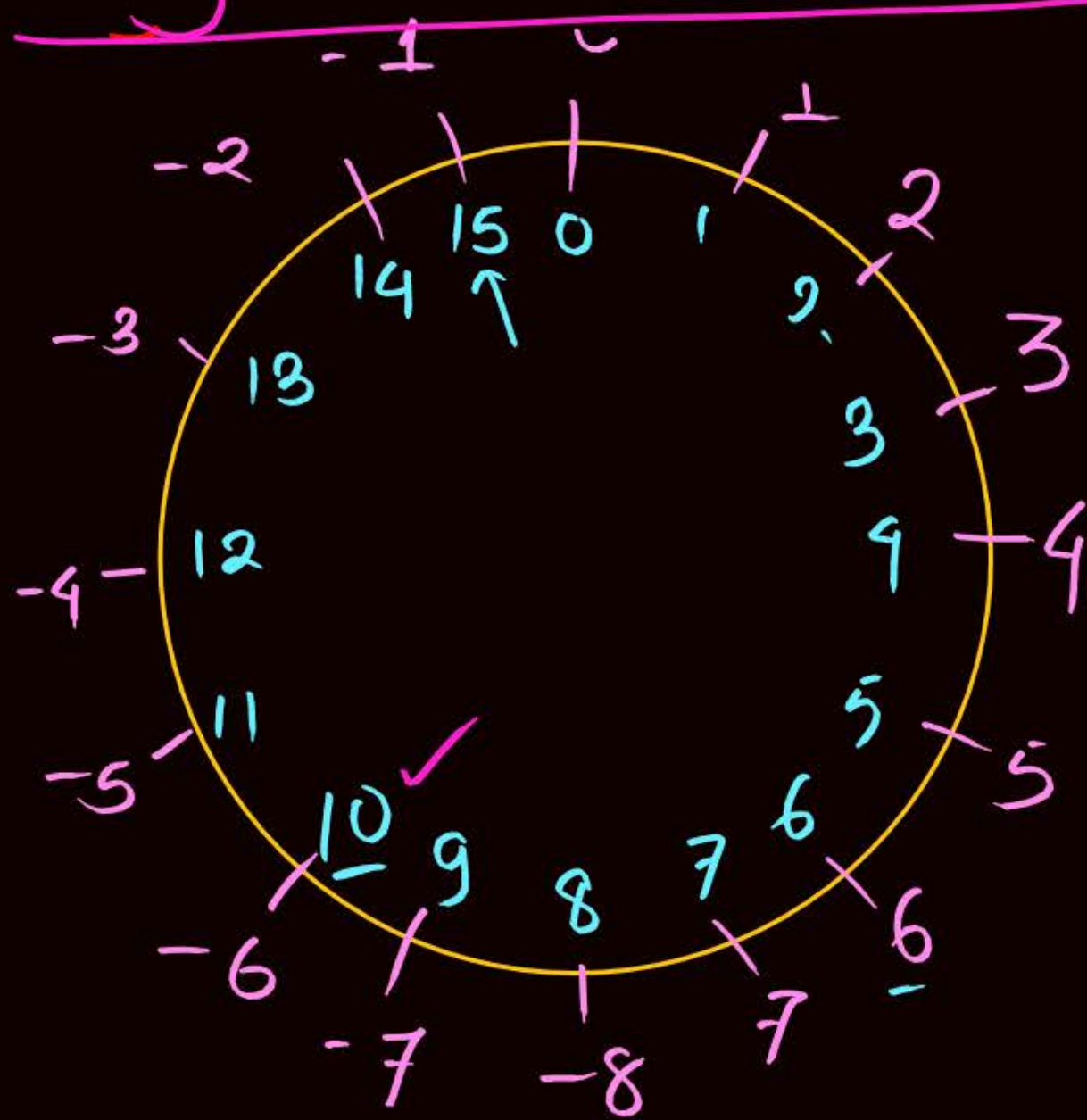
(I) $K = -1$; \leftarrow Signed

Unsigned value 15 $= 2^4 - 1 = 16 - 1 = 15$

(II) $K = -6$

Unsigned value: 10 $= 2^4 - 6 = 16 - 6 = 10$

$$(2^4 - |K|)$$



Negative No. overflow

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

27536

Block
{

```
short int a = -38000;
printf("%hd", a);
```

}

}

-32768

$$(2^{16} - |K|) = 65536 - 38000$$

27536



topic : Question



```
#include <stdio.h>

int main() {

    short int a = 10;

    printf("%hd", a);
    printf("%hu", a);

    return 0;

}
```



topic : Question



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

```
int main() {
```

```
    short int a = -10;
```

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

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

```
    return 0;
```

```
}
```

-10

$$(2^{16} - |K|)$$

$$(65536 - 10) = 65526$$

For Negative
No. unsigned No



topic : Question



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

```
int main() {
```

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

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

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

```
    return 0;
```

```
}
```

$\%$ ← Syntax

A. 20 & 65516

B. -20 & 65516

C. -20 & 20 ✓

D. 20 & -65516

$$\left(2^{16} - 20 \right) = \left(65536 - 20 \right) \\ 65516$$

if No is out of Range

Then first operation $K \% 2^n \leftarrow \underline{\text{No. of bits}}$

After that formula will be used



topic : Question



```
#include <stdio.h>

int main() {
    short int a = -48;
    printf("%hd", a);
    printf("%hu", a);
    return 0;
}
```

- A. -48 & 65488
- B. -48 & -65488
- C. 48 & 65488
- D. -48 & -65516



topic : Question



```
#include <stdio.h>

int main() {
    short int a = -48;
    printf("%hd", a);
    printf("%hu", a);
    return 0;
}
```

- A. -48 & 65488
- B. -48 & -65488
- C. 48 & 65488
- D. -48 & -65516



Homework



```
#include <stdio.h>

int main() {
    short int a = 32770;
    printf("%hd", a);
    printf("%hu", a);
    return 0;
}
```

$$\begin{array}{r} -2^{16-1} \\ -32768 \end{array} \quad \begin{array}{c} \text{---} \text{O} \text{---} \\ \text{---} \text{O} \text{---} \end{array} \quad \begin{array}{r} 2^{16-1} - 1 \\ 32767 \end{array}$$

A. -32766 & 32770

B. 32766 & 32770

C. 32770 & 32770

D. -32766 & 32776

$$k: 32770$$

$$- (2^{16} - 32770)$$

$$\begin{array}{r} 65536 \\ 32770 \\ \hline 32766 \end{array}$$



Homework



```
#include <stdio.h>

int main() {

    short int a = 32770;

    printf("%hd", a);
    printf("%hu", a);

    return 0;

}
```




Character



Character declaration

```
char ch;
```

Character Initialization

```
char ch = 'a';
```

```
printf("%c", ch);
```

Size of character: 1 Byte = 8 bits

'a' ← Single quote

↑ Character constant

format specifier

%c



ASCII value



for each character one Numeric value is assigned. This table Numeric value for each character present ASCII.

<u>0...9</u>	<u>48...57</u>
<u>A...Z</u>	<u>65...90</u>
a...z	<u>(97)...122</u>

A - 65

B - 66

C - 67

⋮

a - 97

b - 98

c - 99

⋮

0 - 48

1 - 49

⋮

8611 - Cyclic property



Sign Character

```
#include <stdio.h>
#include <stdio.h>
int main() {
    char ch1 = 'a';
    char ch2 = 'z';
    printf("%c\n", ch1);
    printf("%d\n", ch1);
    printf("%c\n", ch2);
    printf("%d", ch2);

    return 0;
}
```



String =
Constant

GATE

ASCII
value



Sign Character



```
#include <stdio.h>
#include <stdio.h>
int main() {
    char ch1 = 'A';
    char ch2 = 'Z';
    printf("%c\n", ch1);
    printf("%d\n", ch1);
    printf("%c\n", ch2);
    printf("%d", ch2);

    return 0;
}
```




Sign Character



```
#include <stdio.h>
#include <stdio.h>
int main(){
    char ch1 = '0';
    char ch2 = '9';
    printf("%c\n", ch1);
    printf("%d\n", ch1);
    printf("%c\n", ch2);
    printf("%d", ch2);
    return 0;
}
```

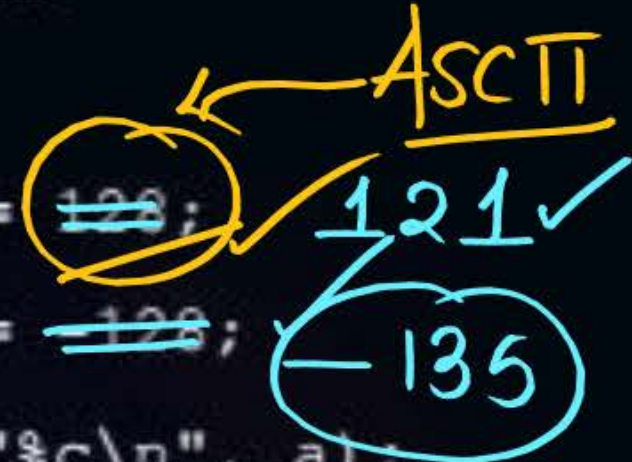


Sign Character

Extended ASCII table

Homework

```
#include <stdio.h>
int main(){
    char a = 128;
    char c = -128;
    printf("%c\n", a);
    printf("%d\n", a);
    printf("%c\n", c);
    printf("%d", c);
    return 0;
}
```



Sign

$$0 - 2^{8-1} - 1$$

ASCII table





2 mins Summary



Topic

positive overflow after modulus — $(2^n - k)$

Topic

Negative overflow, after modulus: $(2^n - |k|)$

Topic

Signed No. unsigned value, after % $(2^n - |k|)$

Topic

char representation. ASCII value

Topic

THANK - YOU