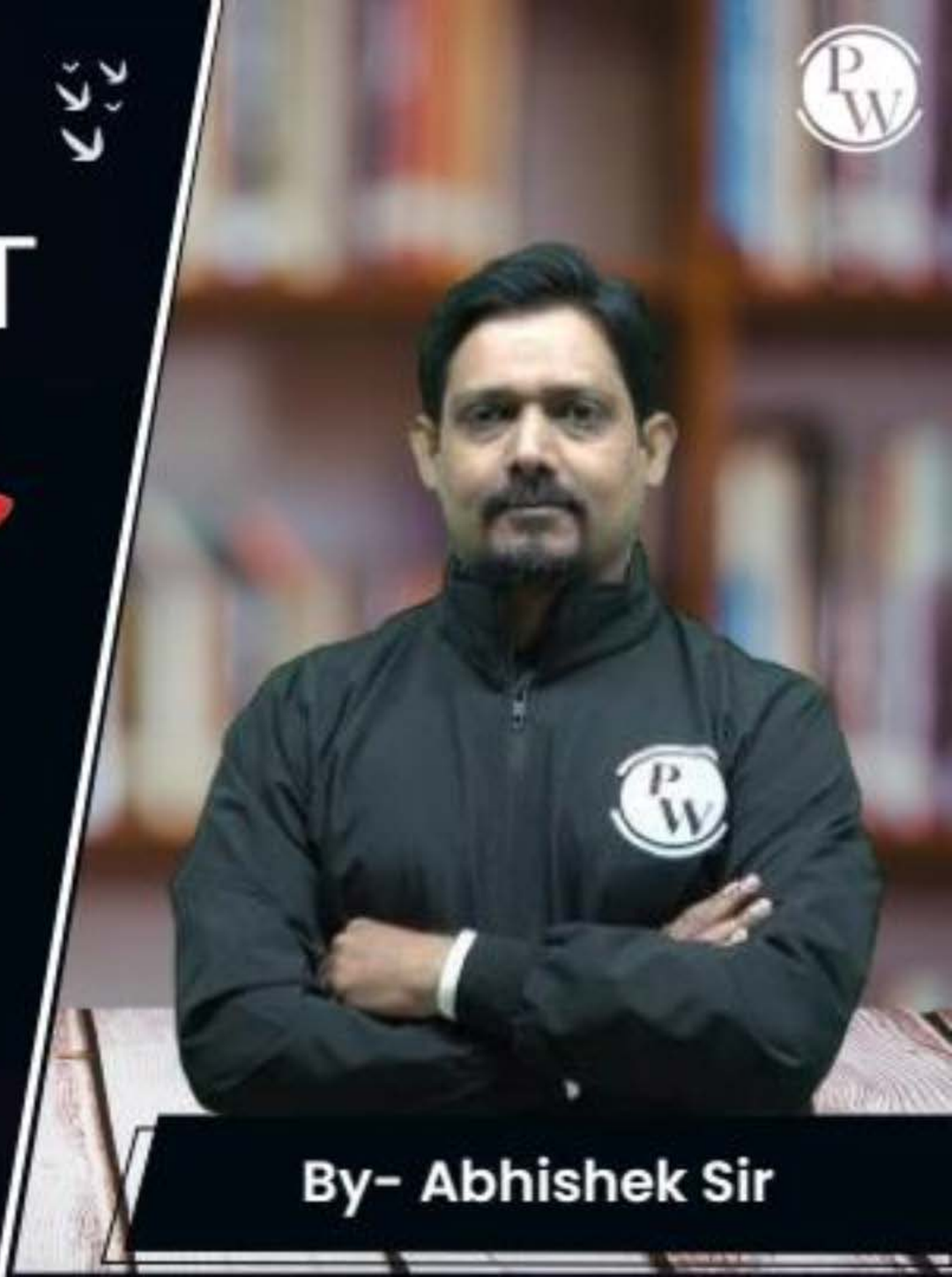
Computer Science & IT

C programming

Data Types & Operators

Lecture No. 02



By- Abhishek Sir

Recap of Previous Lecture

- Topic — fundamental data type
- Topic — Tokens –
- Topic — Constant Binary, octal, Hexadecimal
- Topic
- Topic

Slide

Topics to be Covered

- Topic — Arithmetic expression
- Topic — ++, -- unary operator
- Topic
- Topic
- Topic

Slide

Topic : Operators & Arithmetic Expresion

+ , − , * , / , % ⇒ BODMAS

order of evaluation of operator

C is having its own precedence & associativity Rule

Slide

Topic : Arithmetic Expression Rule

(I) Arithmetic operation between int and int result will be int

$10/3 = 3$

(II) Arithmetic operation between int and float, is float

(III) Arithmetic operation between float and float is float

$\frac{10.0}{3} = 3.33$ $\frac{5.0}{2.0} = 2.5$

Slide

Topic : Arithmetic Operator

+, -, *, /, %

Quotient Remainder

dividend

divisor

Slide

Topic : Arithmetic Operator

Dividend
Divisor
Quotient
Remainder

$\frac{10}{3}$ Q 3
R: 1

$10 = 3 \times 3 + 1$

Dividend = Quotient * Divisor+ Remainder

Slide

```
printf("%d", 6/12);
printf("%d", 6%12);
```

$\frac{6}{12}$

Q : 0

R : 6

Topic : Arithmetic Operator

$-\frac{7}{3}$ Q-2 R-1

7 / 3,

Q: 2

R: 1

-7/3,

Q: -2

R: -1

7/-3,

Q: -2

R: 1

-7/-3

Q: 2

R: -1

% (I) dividend is the sign of Remainder

/ (II) if dividend & divisor same sign then Quotient +ve

different sign Negative

Slide

$$\frac{17}{-2} \qquad Q \; -8 \qquad R: \; 1$$

repeated division

X / Y repeated Subtraction

```
while( r>=y) {
    r = r-y;
    q = q+1;
}
```

Condition x, y, r, q

$X == Y * q + r$

(A) $(q == r)$ && $r = 0$ X

(B) $(x > 0)$ && $r = x$ && $y > 0$

(C) $(q == 0)$ && $r == x$ && $y > 0$

(D) $(q == 0)$ && $(y > 0)$ X

7 X / 2 Y	Q / R	7-2=5 q=1	3-2=1 q=3
r=X q=0		5-2=3 q=2	1>=2 q=3 r=1

repeated division

x / y repeated subtraction

```
while( r >= y) {
    r = r - y;
    q = q + 1;
}
```

Condition x, y, r, q

$X == y * q + r$

10 x
3 y

10 − Remainder

10 − 3 = 7 q = 1

7 − 3 = 4 q = 2

4 − 3 = 1 q = 3

Remainder >= y

Topic : Arithmetic Operator

	7/3	7/-3	-7/3	-7/-3
Dividend	7	7	-7	-7
Divisor	3	-3	3	-3
Quotient	2	-2	-2	2
Remainder	1	1	-1	-1

Program

```
#include <stdio.h>
int main() {
        printf("%d\n",7/3);
        printf("%d\n",7%3);
        printf("%d\n",7/-3);
        printf("%d\n",7%-3);
        printf("%d\n",-7/3);
        printf("%d\n",-7%3);
        printf("%d\n",-7/-3);
        printf("%d\n",-7%-3);
        return 0;
}
```

Precedence Rule

precedence Rule decide order of evaluation of operator

Slide

Precedence & Associativity Rule

-5+6 ⇒ 1

-(5+6) = -11

-5+6

bracket	()		Highest
Unary minus	-		
Multiplicative	*, / , %	Left to Right	
Additive	+ -	Left to Right	
Equal	=	Right to Left	Lowest

unary operator precedence higher

L to R

$5 \times 6 - / 3$

$30/3$

$= 10$

int a = 10, b = 20, c = 30;

a = b = c,

a = (b = c)

Right to left

a = 30

b = 30

c = 30

Precedence & Associativity Rule

* / %	Multiplication/division/modulus	left-to-right
+ -	Addition/subtraction	left-to-right
<< >>	Bitwise shift left, Bitwise shift right	left-to-right
< <= > >=	Relational less than/less than or equal to Relational greater than/greater than or equal to	left-to-right
== !=	Relational is equal to/is not equal to	left-to-right
&	Bitwise AND	left-to-right
^	Bitwise exclusive OR	left-to-right
|	Bitwise inclusive OR	left-to-right
&&	Logical AND	left-to-right
||	Logical OR	left-to-right

Precedence & Associativity Rule

? :	Ternary conditional	right-to-left
=	Assignment	right-to-left
+= -=	Addition/subtraction assignment	
*= /=	Multiplication/division assignment	
%= &=	Modulus/bitwise AND assignment	
^= |=	Bitwise exclusive/inclusive OR assignment	
<<= >>=	Bitwise shift left/right assignment	
,	Comma (separate expressions)	left-to-right

Lowest

1 postfix ++, --, [] (), . →

2 prefix → ++, --, &, !, ~, *, sizeof, +, -

3 Arith

Topic:Question

```
#include <stdio.h>
int main(void){
float x;
x = 7*2.0/2+10/3;
printf("%f", x);
return 0;
}
```

The value of x is ___

A. 10

B. 10.0

C. 10.33

D. 11.0

7 * 20/2 + 10/3

14.0/2 + 3

7.0 + 3 = 10.0

Slide

Topic:Question

```
#include<stdio.h>
   int main(){
   int x;
   x= -2 + 11 - 7 * 9 % 6 / 12;
   printf("%d",x);
   return 0 ;
   }
```

The value of x is

A. 6
B. 7
C. 8
D. 9

-2+11-7*9%6/12

-2+11-63%6/12

-2+11-3/12

-2+11-0 = 9

3 Q.0
12 R.63

63 Q 10
6 R:3

Slide

Topic:Question

```
#include<stdio.h>
   int main(){
   int x;
   x=  2 * 3/4+4/4 +8-2+5/8;
   printf("%d",x);
   return 0 ;
   }
```

The value of x is

____

(A) 6
(b) 7
(c) 8
(d) 9

Slide

Topic:Question

```
#include<stdio.h>
   int main(){
   int x;
   x=  3/2*4+3/8+3;
   printf("%d",x);
   return 0 ;
   }
```

The value of x is

____

(a) 6
(b) 7
(c) 8
(d) 9

Slide

Topic:Question

-14 Q:-1 | 2 Q:0
8 R:-6 | -8 R:2

```
#include<stdio.h>
   int main(){
   int x;
   x=  -14%8+2%-8;
   printf("%d",x);
   return 0 ;
   }
```

-14%8+2%-8

-6 + 2 = -4

The value of x is

____

(a) -6
(b) 7
(c) -4
(d) 9

Slide

#Q. Consider a c program

```
#include<stdio.h>
int g(int i, int j){
  return ((2*(i/4)) + (4/ j - 3)) % (i*i - j - 10);
}
```

Which of following is FALSE?

(A) g(20, 9) will return 7 ✓ true

(B) g(16, 9) will return 5 ✓ true

(C) g(9, 9) will return 5 ✗

(D) g(7, 9) will return -1 ✓ true

$(2*4) + (-3) \% (256 - 9 - 10)$

$8 - 3 = 5$

$2*2 + (-3) \% \quad = 1$

$(2 + -3) \% 30$

$= -1$

Topic: ++ and --

L-value is Address of a variable

* ++ , -- operator is modifier operator
* It updates value of a variable
* int a = 5

a = a+1

++a or a++

Same

int b = 10

--b or b--; b = b-1

Same

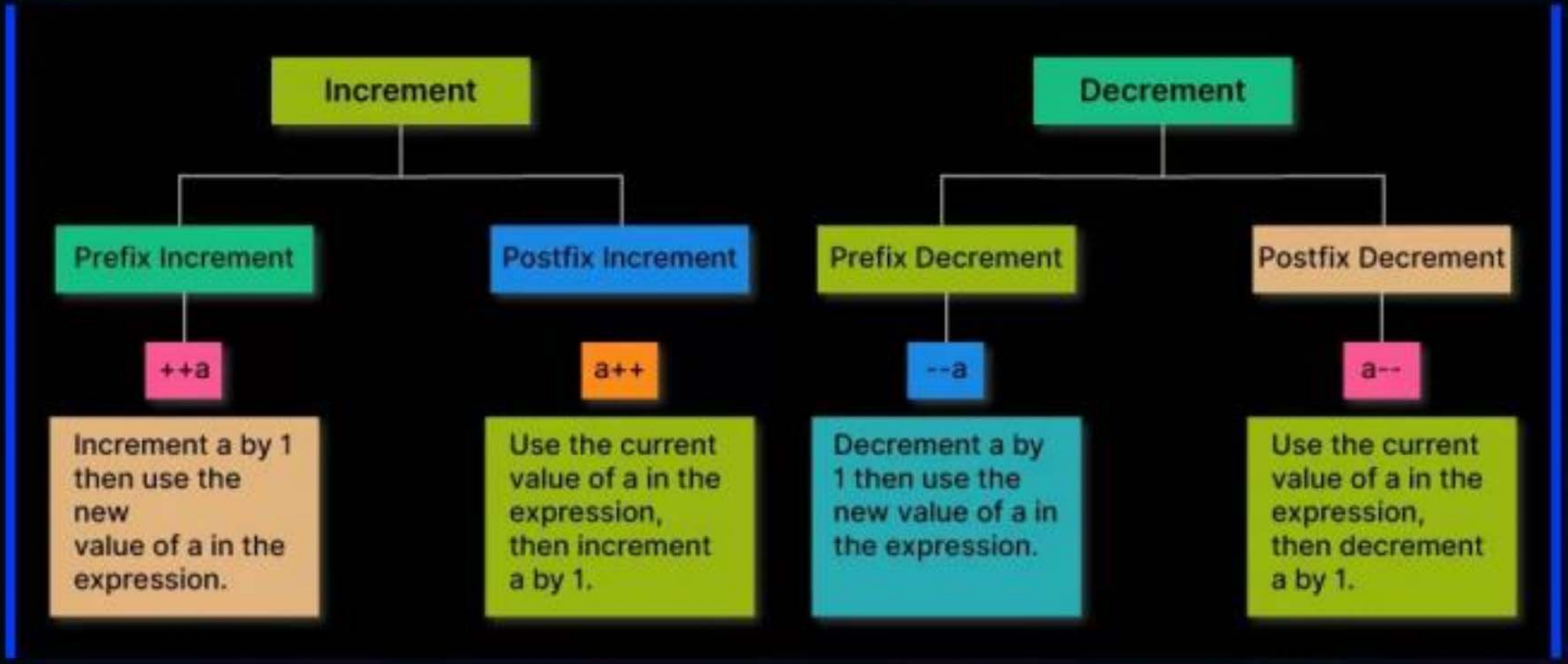
* ++ operator does not apply to expression, constant

(a+b)++ . 5++;

L-value required

Slide

Topic: Increment & decrement Operator



Slide

Topic: Question

```
#include <stdio.h>
      int main ()  {
       int x=5;
       int y;
       y = ++x;
      printf("%d %d", x, y);
     }
```

X = 6

Y = 6

prefix Increment ∴ Honest

first update then it will used.

expression

The output of the program

(A) 5,5

(B) 5,6

(C) 6,5

(D) 6,6

Slide

Topic: Question

```
#include <stdio.h>
      int main ()  {
       int x=5;
       int y;
       y = x++;
      printf("%d %d", x, y);
     }
```

The output of the program

(A) 5,5 (B) 5,6 (C) 6,5 (D) 6,6

Answer

Slide

first we use old value in expression

y = x++;

y = 5

after that value increment/decrement

x = 6

y++

[C]

Topic: ++ and --

```
#include <stdio.h>
int main(){
    int x = 5, y, b=10;
    y  = x++;
    y = ++x;
    y = x++*b;
    y = ++x*b;
    y  = x--;
    y = --x;
    y = x--*b;
    y = --x*b;

    printf("%d\n",x+y);
    return 0;
}
```

x

5	6	7	8	9	8	7	6	5

y = 9

y = 7

y = 7 * 10 = 70

y = 5 * 10 = 50

5 + 50 = 55

Slide

Topic: ++ and --

```
#include <stdio.h>
int main(){
    int x = 5, y, b=10;
    y  = x++;
    y = ++x;
    y = x++*b;
    y = ++x*b;
    y  = x--;
    y = --x;
    y = x--*b;
    y = --x*b;

    printf("%d\n",x+y);
    return 0;
}
```

x | 5 | 6 | 7 | 8 | 9

y = 5

y = ++x y = 7

y = 7 * 10 = 70

y = 9 * 10 = 90

Slide

Topic: Question

What is the output of the program?

```
#include <stdio.h>
      int main ()   {
        int x=5;
        int y;
        y = x++;
         printf("%d %d", x, y);
     }
```

Output of the program is_____

(A) 5,5

(B) 5,6

(C) 6,5

(D) 6,6

Slide

Topic: Question

What is the output of the program?

```
#include <stdio.h>
      int main ()   {
        int x=5;
        int y;
        y = x++;
         printf("%d %d", x, y);
     }
```

Output of the program is_____

(A) 5,5

(B) 5,6

(C) 6,5

(D) 6,6

Slide

Topic: Assigment Operator

=

a+=b	=	a=a+b
a-=c		a=a-c
a*=10		a=a*10
a/=2		a=a/2
a%=5		a=a%5

Slide

Question

What will be output if you will execute following c code?

```
#include<stdio.h>
int main(){
int num,a=5;
num=-a--;
printf("%d  %d",num,a);
}
```

num = - a - - ; ← post

unary

top ① postfix

② = postfix

-5 4

num = -5

(a) 5 4

(b) -4 4

(c) -5 4

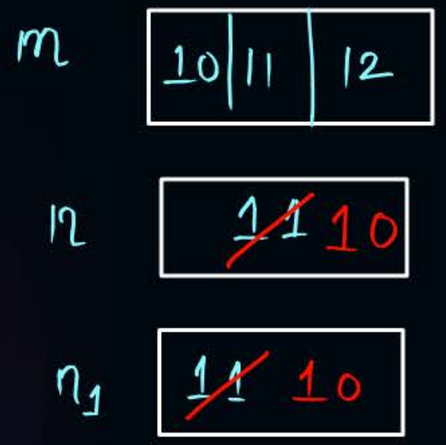
(d) -4 5

Slide

Topic: GATE 2017

Consider the following C program.

```
#include<stdio.h>
int main () {
    int m=10;
    int n, n1;
    n=++m;
    n1=m++;
    n--;
    --n1;
    n-=n1;
    printf("%d", n);
    return 0;
}
```



n = n - n1 = 10 - 10 = 0

0

The output of the program is 0

Slide

Question

```
#include <stdio.h>
int main()     {
        int num1 = 5;
        int num2 = 3;
        int num3 = 2;
        num1 = num2++;
        num2 = --num3;
        return 0;
    }
```

num1 5/3

num2 3/4/1

num3 2/1

num1 = num2

num2 = --num3

The valur of num1+num2+num3 3+1+1=5

Slide

++ & -- Operator

Consider the following C program.

```
##include<stdio.h>
int main () {
    int m=20, k=50;
    int n, n1;
    n=++m + ++k;
    n1=m-- + --k ;
    n--;
    --n1;
    n-=n1;
    printf("%d", n+k);
    return 0;
}
```

The output of the program is _____

Slide

Question

```
#include <stdio.h>
int main()     {
        int num1 = 5;
        int num2 = 3;
        int num3 = 2;
        num1 = num2++;
        num2 = --num3;
        return 0;
    }
```

The valur of num1+num2+num3_____. Ans 5

Slide

Question

```
#include <stdio.h>
int main()     {
        int num1 = 5;
        int num2 = 3;
        int num3 = 2;
        num1 = num2++ * --num3 / ++num1;
        printf( "%d %d %d", num1, num2, num3);
        return 0;
    }
```

num1 5 6 0

num2 3 4

num3 = 2 1

A. 0 4 1
B. 6 4 1
C. 6 3 2
D. 0 3 2

num1 = 3 * 1 / 6

= 3/6 = 0

Slide

Question

Consider the following C program.

```
#include<stdio.h>
int main () {
        int a=2,b=5,c;
        c=a+++b;
        printf("%d",a+b+c);
        return 0;
}
```

The output of the program is 3+5+7 = 15

c = a++ + b

c = 2 + 5 = 7

a = 3

c = a++ + b

a + ++b

+ - token

++ 2nd token

Slide

2 mins Summary

Topic Arithmetic expression

Topic Division , Quotient, Remainder

Topic ++, --, post

Topic - - ++ pre

Topic

Slide

Relation. Logical

THANK - YOU

Slide