Technical Training

Mission 2024

C-TRAINING

DAY-02

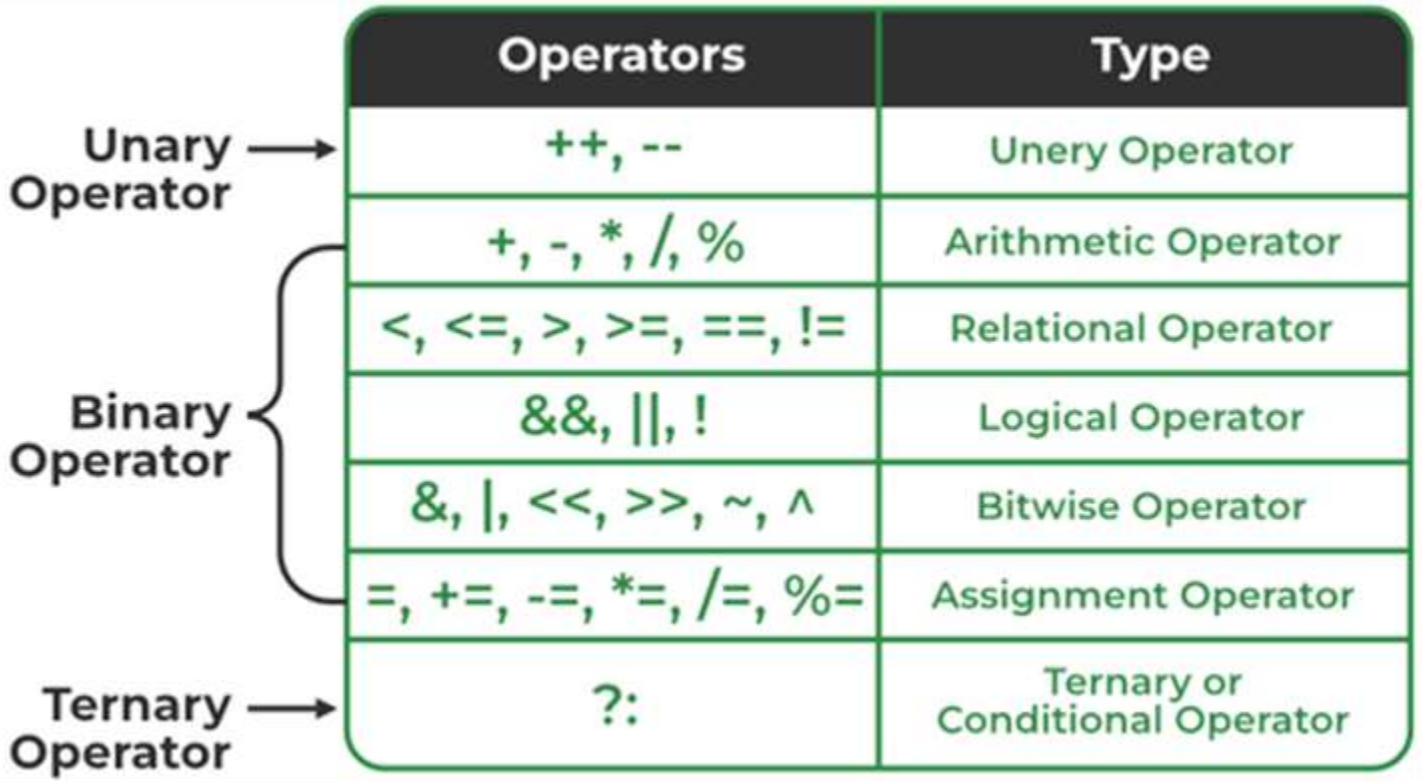
BATCH 2023-2024

DILIP BHARTI

OPERATORS

List of topics to be covered:

- Introduction to C Operators
- Unary Increment and Decrement Operator
- Arithmetic Operators
- Relational Operators
- Logical Operators
- Assignment Operators
- Bitwise Operators
- Conditional (Ternary) Operator
- Special Operators



Operators	Type
++, --	Unery Operator
+, -, *, /, %	Arithmetic Operator
<, <=, >, >=, ==, !=	Relational Operator
&&, ||, !	Logical Operator
&, |, <<, >>, ~, ^	Bitwise Operator
=, +=, -=, *=, /=, %=	Assignment Operator
?:	Ternary or Conditional Operator

C-TRAINING

ABES Engineering College

Introduction to C Operators:

Operators are symbols used to perform operations on variables and values. Emphasize the importance of operators in C for manipulating data.

Unary Increment and Decrement Operator:

Unary increment (++) and decrement (--) operators:

Unary Increment Operator (++x and x++):

Unary increment operator (++).
The difference between prefix (++x) and postfix (x++) usage.

Decrement Operator (--x and x--):

Unary decrement operator (--).
The difference between prefix (--x) and postfix (x--) usage.

C-TRAINING

ABES Engineering College

Arithmetic Instructions:

Example:
x=10/2*3%2+5;

Solution:

?

Arithmetic Operators:

- Addition (+)
- Subtraction (-)
- Multiplication (*)
- Division (/)
- Modulus (%)

	Operator Precedence
1	! Logical not (Highest)
2	( ) Parenthesis
3	*, /, %
4	+, -
5	>, >=, <, <=
6	==, !=
7	&& (AND)
8	|| (OR)
9	= (Lowest)

C-TRAINING

ABES Engineering College

Relational Operators:

- Equal to (==)
- Not equal to (!=)
- Greater than (>)
- Less than (<)
- Greater than or equal to (>=)
- Less than or equal to (<=)

Logical Operators:

Logical AND (&&)
Logical OR (||)
Logical NOT (!)

Assignment Operators:

Assignment (=)
Addition assignment (+=)
Subtraction assignment (-=)
Multiplication assignment (*=)
Division assignment (/=)
Modulus assignment (%=)

Bitwise Operators:

Bitwise AND (&)
Bitwise OR (|)
Bitwise XOR (^)
Bitwise NOT (~)
Left shift (<<)
Right shift (>>)

Conditional Operator

Explain the conditional operator (? :)

(Condition)? True Action : False Action

Special Operators

Sizeof()	Sizeof Operator to find out the size of data type, variable and constant value
,	Comma
*	Dereferencing Operator
&	Referencing Operator
[ ]	Subscript Operator
( )	Parenthesis
.	Dot
->	Arrow

C-TRAINING

ABES Engineering College

Operators:

Example: DAY02/HR04

```c
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int a, b;
    scanf("%d %d", &a, &b);
    printf("%d\n", a + b);
    printf("%d\n", abs(a - b));
    printf("%d\n", a * b);
    return 0;
}
```

Sample Input:

3

4

Sample Output:

C-TRAINING

ABES Engineering College

Character Set, Format Specified

Example: DAY02/HR04

```c
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int a, b;
    scanf("%d %d", &a, &b);
    printf("%d\n", a + b);
    printf("%d\n", abs(a - b));
    printf("%d\n", a * b);
    return 0;
}
```

Sample Input:

3

4

Sample Output:

7

1

12

C-TRAINING

ABES Engineering College

Operators:

Example: DAY02/HR05

```c
#include <stdio.h>
int main() {
    int n;
    scanf("%d", &n);
    printf("%d\n", n);
    printf("%d ", ++n);
    printf("%d\n", --n);
    return 0;
}
```

Sample Input:

5

Sample Output:

C-TRAINING

ABES Engineering College

Operators:

Example: DAY02/HR05

```c
#include <stdio.h>
int main() {
    int n;
    scanf("%d", &n);
    printf("%d\n", n);
    printf("%d ", ++n);
    printf("%d\n", --n);
    return 0;
}
```

Sample Input:

5

Sample Output:

5

6 5

C-TRAINING

ABES Engineering College

Operators:

Example: DAY02/HR05

```c
#include<stdio.h>
int main()
{
    int x = 2;
    (x & 1) ? printf("true") : printf("false");
    return 0;
}
```

Sample Output:

C-TRAINING

ABES Engineering College

Operators:

Example: DAY02/HR05

```c
#include<stdio.h>
int main()
{
    int x = 2;
    (x & 1) ? printf("true") : printf("false");
    return 0;
}
```

Sample Output:

false

THANK YOU