# Operators in C with Example Programs

## Arithmetic Operators

These operators perform basic mathematical operations like addition, subtraction, multiplication, etc.

* Operator: +

Example Program:

int a = 10, b = 20;  
int sum = a + b;  
printf("Sum: %d", sum);

* Operator: -

Example Program:

int a = 10, b = 20;  
int diff = b - a;  
printf("Difference: %d", diff);

* Operator: \*

Example Program:

int a = 10, b = 20;  
int product = a \* b;  
printf("Product: %d", product);

* Operator: /

Example Program:

int a = 20, b = 10;  
int quotient = a / b;  
printf("Quotient: %d", quotient);

* Operator: %

Example Program:

int a = 20, b = 3;  
int remainder = a % b;  
printf("Remainder: %d", remainder);

## Relational Operators

These operators are used to compare two values and return a boolean result.

* Operator: ==

Example Program:

int a = 10, b = 10;  
printf("Equal: %d", a == b);

* Operator: !=

Example Program:

int a = 10, b = 20;  
printf("Not Equal: %d", a != b);

* Operator: >

Example Program:

int a = 20, b = 10;  
printf("Greater: %d", a > b);

* Operator: <

Example Program:

int a = 10, b = 20;  
printf("Lesser: %d", a < b);

* Operator: >=

Example Program:

int a = 10, b = 10;  
printf("Greater or Equal: %d", a >= b);

* Operator: <=

Example Program:

int a = 10, b = 20;  
printf("Lesser or Equal: %d", a <= b);

## Logical Operators

These operators are used to perform logical operations.

* Operator: &&

Example Program:

int a = 1, b = 0;  
printf("Logical AND: %d", a && b);

* Operator: ||

Example Program:

int a = 1, b = 0;  
printf("Logical OR: %d", a || b);

* Operator: !

Example Program:

int a = 1;  
printf("Logical NOT: %d", !a);

## Bitwise Operators

These operators perform bit-level operations.

* Operator: &

Example Program:

int a = 5, b = 3;  
printf("Bitwise AND: %d", a & b);

* Operator: |

Example Program:

int a = 5, b = 3;  
printf("Bitwise OR: %d", a | b);

* Operator: ^

Example Program:

int a = 5, b = 3;  
printf("Bitwise XOR: %d", a ^ b);

* Operator: ~

Example Program:

int a = 5;  
printf("Bitwise Complement: %d", ~a);

* Operator: <<

Example Program:

int a = 5;  
printf("Left Shift: %d", a << 1);

* Operator: >>

Example Program:

int a = 5;  
printf("Right Shift: %d", a >> 1);

## Assignment Operators

These operators are used to assign values to variables.

* Operator: =

Example Program:

int a;  
a = 10;  
printf("Assigned Value: %d", a);

* Operator: +=

Example Program:

int a = 10;  
a += 5;  
printf("Add and Assign: %d", a);

* Operator: -=

Example Program:

int a = 10;  
a -= 5;  
printf("Subtract and Assign: %d", a);

* Operator: \*=

Example Program:

int a = 10;  
a \*= 2;  
printf("Multiply and Assign: %d", a);

* Operator: /=

Example Program:

int a = 10;  
a /= 2;  
printf("Divide and Assign: %d", a);

* Operator: %=

Example Program:

int a = 10;  
a %= 3;  
printf("Modulus and Assign: %d", a);

## Increment and Decrement Operators

These operators are used to increase or decrease a value by 1.

* Operator: ++

Example Program:

int a = 10;  
printf("Pre-increment: %d", ++a);  
printf("Post-increment: %d", a++);

* Operator: --

Example Program:

int a = 10;  
printf("Pre-decrement: %d", --a);  
printf("Post-decrement: %d", a--);

## Conditional (Ternary) Operator

A compact way to write an if-else condition.

* Operator: ? :

Example Program:

int a = 10, b = 20;  
int max = (a > b) ? a : b;  
printf("Maximum: %d", max);

If the marks obtained by a student in five different subjects are input through the keyboard, find out the aggregate marks and percentage marks obtained by the student. Assume that the maximum marks that can be obtained by a student in each subject is 100.

#include<stdio.h>

**int** main()

{

**int** m1,m2,m3,m4,m5,total;

**float** avg;

printf("\nEnter Five Marks : ");

scanf("%d%d%d%d%d",&m1,&m2,&m3,&m4,&m5);

total=m1+m2+m3+m4+m5;

avg=total/5.0;

printf("\nTotal : %d",total);

printf("\nAverage : %0.2f",avg);

**return** 0;

}

o/p

Enter Five Marks : 75

78

98

85

99

Total : 435

Average : 87.00