SESSION 2: OPERATORS & INPUT/OUTPUT

In this session, we will learn about different operators in C, along with standard input/output functions (printf and scanf). We will also see example codes demonstrating the use of these operators, and build a simple calculator program using arithmetic operators only.

Operators in C

1. Arithmetic Operators
+ : Addition
- : Subtraction
* : Multiplication
/ : Division
% : Modulus (remainder after division)

Example Code:

```c
#include <stdio.h>
int main() {
    int a = 10, b = 3;
    printf("Addition: %d\n", a + b);
    printf("Subtraction: %d\n", a - b);
    printf("Multiplication: %d\n", a * b);
    printf("Division: %d\n", a / b);
    printf("Modulus: %d\n", a % b);
    return 0;
}
```

2. Relational Operators
== : Equal to
!= : Not equal to
> : Greater than
< : Less than
>= : Greater than or equal to
<= : Less than or equal to

Example Code:

```c
#include <stdio.h>
int main() {
    int a = 10, b = 20;
    printf("a == b: %d\n", a == b);
    printf("a != b: %d\n", a != b);
```

```c
    printf("a > b: %d\n", a > b);
    printf("a < b: %d\n", a < b);
    printf("a >= b: %d\n", a >= b);
    printf("a <= b: %d\n", a <= b);
    return 0;
}
```

3. Logical Operators

&& : Logical AND
|| : Logical OR
! : Logical NOT

Example Code:

```c
#include <stdio.h>
int main() {
    int a = 1, b = 0;
    printf("a && b: %d\n", a && b);
    printf("a || b: %d\n", a || b);
    printf("!a: %d\n", !a);
    printf("!b: %d\n", !b);
    return 0;
}
```

4. Bitwise Operators

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

Example Code:

```c
#include <stdio.h>
int main() {
    int a = 5, b = 3;
    printf("a & b: %d\n", a & b);
    printf("a | b: %d\n", a | b);
    printf("a ^ b: %d\n", a ^ b);
    printf("~a: %d\n", ~a);
    printf("a << 1: %d\n", a << 1);
    printf("a >> 1: %d\n", a >> 1);
    return 0;
}
```

5. Assignment Operators

= : Simple assignment
+= : Add and assign
-= : Subtract and assign
*= : Multiply and assign
/= : Divide and assign
%= : Modulus and assign

Example Code:

```c
#include <stdio.h>
int main() {
    int a = 10;
    a += 5;
    printf("a after += 5: %d\n", a);
    a -= 3;
    printf("a after -= 3: %d\n", a);
    a *= 2;
    printf("a after *= 2: %d\n", a);
    a /= 4;
    printf("a after /= 4: %d\n", a);
    a %= 3;
    printf("a after %= 3: %d\n", a);
    return 0;
}
```

6. Ternary Operator

Syntax: condition ? expression1 : expression2

Example Code:

```c
#include <stdio.h>
int main() {
    int a = 10, b = 20;
    int max = (a > b) ? a : b;
    printf("The maximum is: %d\n", max);
    return 0;
}
```

Standard Input/Output Functions

1. printf() - Used to display output on the screen.
   Example: printf("Hello, World!");

2. scanf() - Used to take input from the user.
   Example: scanf("%d", &number);

Calculator Program (Arithmetic Operators Only)

```c
#include <stdio.h>
int main() {
    int a, b;
    printf("Enter first number: ");
    scanf("%d", &a);
    printf("Enter second number: ");
    scanf("%d", &b);

    printf("Addition: %d\n", a + b);
    printf("Subtraction: %d\n", a - b);
    printf("Multiplication: %d\n", a * b);
    if(b != 0)
        printf("Division: %d\n", a / b);
    else
        printf("Error: Division by zero!\n");
    printf("Modulus: %d\n", a % b);

    return 0;
}
```