Operators Assignment

Bitwise Operations:

The bitwise operators are the operators used to perform the operations on the data at the bit-level. It consists of two digits, either 0 or 1.

Syntax: a&b, a|b , etc.

Example,

```
#include <stdio.h>
int main()
{
    // a = 5(00000101), b = 9(00001001)
    unsigned char a = 5, b = 9;

    // The result is 00000001
    printf("a = %d, b = %d\n", a, b);
    printf("a&b = %d\n", a & b);

    // The result is 00001101
    printf("a|b = %d\n", a | b);

    // The result is 00001100
    printf("a^b = %d\n", a ^ b);

    // The result is 11111010
    printf("~a = %d\n", a = ~a);

    // The result is 00010010
    printf("b<<1 = %d\n", b << 1);

    // The result is 00000100
    printf("b>>1 = %d\n", b >> 1);

    return 0;
}
```

Ternary Operator in C

The ternary operator is used to execute code based on the result of a binary condition.

It takes in a binary condition as input, which makes it similar to an 'if-else' control flow block. It also, however, returns a value, behaving similar to a function.

Syntax:

result = binaryCondition ? valueReturnedIfTrue : valueReturnedIfFalse;

Example,

```
 //Looking at the maximum example
int findMaximum(int a, int b){
    //if a > b, it returns a, if not it returns b
    return (a > b) ? a : b;
}
```

A2)

```
#include stdio.h
Int main(){
        int a=4,b=5,s,d,e,f,g;
        s=a+b;
        printf(“sum of %d and %d is %d”,a,b,s);
        d=a-b;
        printf(“difference of %d and %d is %d”,a,b,d);
        e=a*b;
        printf(“product of %d and %d is %d”,a,b,e);
        f=a/b;
        printf(“%d divided by %d is %d ”,a,b,f);
        g=a%b;
        printf(“modulus of %d and %d is %d”,a,b,g);
        return 0;    }
```