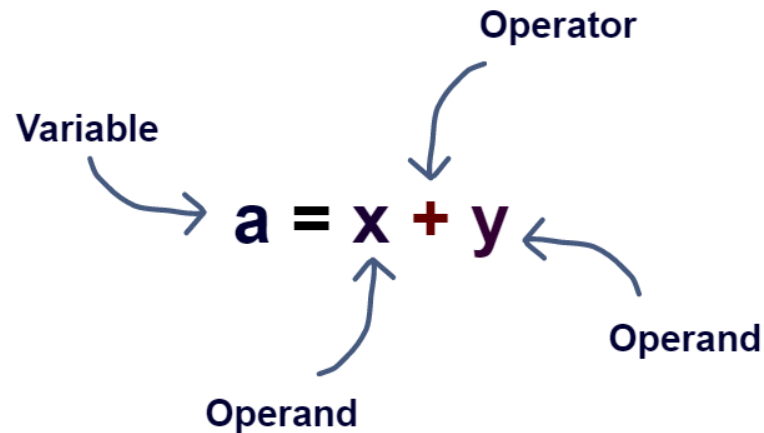


5- İFADELER VE İŞLEM OPERATÖRLERİ

İFADE (EXPRESSION)

- C'de bir ifade (expression), işlenenlerin (operand) ve işleçlerin (operator) bir birleşimidir - bir değişkende (variable) depolanan tek bir değeri hesaplar. Operatör, gerçekleştirilecek eylemi veya işlemi belirtir. İşlenenler, işlemi uyguladığımız öğelerdir.



İFADE TÜRLERİ

- **Aritmetik İfade:** Aritmetik operatörlerden (+ , - , * ve /) oluşur ve int, float veya double tipindeki değerleri hesaplar.
- **İlişkisel İfade:** Genellikle karşılaştırma operatörlerinden (> , < , >= , <= , == ve !=) oluşur ve yanıtı bool türünde, yani true (1) veya false (0) olarak hesaplar.
- **Mantıksal İfade:** Mantıksal operatörlerden (&&, || ve !) oluşur ve bool türündeki yanıtları hesaplamak için ilişkisel ifadeleri birleştirir.
- **Koşullu İfade:** Koşul sağlanırsa doğru, değilse yanlış olan koşullu ifadelerden oluşur.
- **Pointer İfade:** Bir ve işareti (&) operatöründen oluşabilir ve adres değerlerini döndürür.
- **Bitsel İfade:** Bitsel operatörlerden (>>, <<, ~, &, | ve ^) oluşur ve işlemleri bit düzeyinde gerçekleştirir.

İŞLEM ÖNCELİĞİ

Precedence	Operator	Description	Associativity
1	++ --	Suffix/postfix increment and decrement	Left-to-right
	()	Function call	
	[]	Array subscripting	
	.	Structure and union member access	
	->	Structure and union member access through pointer	
	(type){list}	Compound literal(C99)	
2	++ --	Prefix increment and decrement ^[note 1]	Right-to-left
	+ -	Unary plus and minus	
	! ~	Logical NOT and bitwise NOT	
	(type)	Cast	
	*	Indirection (dereference)	
	&	Address-of	
	sizeof	Size-of ^[note 2]	
	_Alignof	Alignment requirement(C11)	

3	* / %	Multiplication, division, and remainder	Left-to-right
4	+ -	Addition and subtraction	
5	<< >>	Bitwise left shift and right shift	
6	< <=	For relational operators < and ≤ respectively	
	> >=	For relational operators > and ≥ respectively	
7	== !=	For relational = and ≠ respectively	
8	&	Bitwise AND	
9	^	Bitwise XOR (exclusive or)	Right-to-left
10		Bitwise OR (inclusive or)	
11	&&	Logical AND	
12		Logical OR	
13	?:	Ternary conditional ^[note 3]	
14 ^[note 4]	=	Simple assignment	
	+= -=	Assignment by sum and difference	
	*= /= %=	Assignment by product, quotient, and remainder	
	<<= >>=	Assignment by bitwise left shift and right shift	
15	&= ^= =	Assignment by bitwise AND, XOR, and OR	
	,	Comma	Left-to-right