

1. 산술 연산자

1. 기본 산술 연산자

연산자	설명	사용 예	사용 예에 대한 설명
=	대입 연산자	$a = 3$	정수 3을 a에 대입한다.
+	더하기	$a = 5 + 3$	정수 5와 3을 더한 값을 a에 대입한다.
-	빼기	$a = 5 - 3$	정수 5와 3을 뺀 값을 a에 대입한다.
*	곱하기	$a = 5 * 3$	정수 5와 3을 곱한 값을 a에 대입한다.
/	나누기	$a = 5 / 3$	정수 5를 3으로 나눈 값을 a에 대입한다.
%	나머지 값	$a = 5 \% 3$	정수 5를 3으로 나눈 뒤 나머지 값을 a에 대입한다.

1. 산술 연산자

3. 대입 연산자와 증감 연산자

- C에서는 대입 연산자 = 외에도 +=, -=, *=, /=, %=를 사용할 수 있음
- 값을 1씩 증가 시키는 역할을 하는 ++ 연산자와 1씩 감소시키는 역할을 하는 -- 연산자도 있음

표 4-2 대입 연산자와 증감 연산자

연산자	명칭	사용 예	설명
+=	대입 연산자	a += 3	a = a + 3과 동일하다.
-=	대입 연산자	a -= 3	a = a - 3과 동일하다.
*=	대입 연산자	a *= 3	a = a * 3과 동일하다.
/=	대입 연산자	a /= 3	a = a / 3과 동일하다.
%=	대입 연산자	a %= 3	a = a % 3과 동일하다.
++	증가 연산자	a++ 또는 ++a	a += 1 또는 a = a + 1과 동일하다.
--	감소 연산자	a-- 또는 --a	a -= 1 또는 a = a - 1과 동일하다.

2. 관계 연산자

- 관계 연산자(또는 비교 연산자)는 어떤 것이 큰지, 작은지, 같은지를 비교하는 것으로 그 결과는 참(true)이나 거짓(false) 중 하나
- 조건문(if)이나 반복문(for, while)에서 사용하며 단독으로 사용하는 경우는 별로 없음
- 일반적으로 참은 1로, 거짓은 0으로 표시

$$a < b = \begin{cases} \text{참: 1} \\ \text{거짓: 0} \end{cases}$$

그림 4-4 관계 연산자의 기본 개념

표 4-3 관계 연산자

연산자	의미	설명
==	같다.	두 값이 동일하면 참이다.
!=	같지 않다.	두 값이 다르면 참이다.
>	크다.	왼쪽이 크면 참이다.
<	작다.	왼쪽이 작으면 참이다.
>=	크거나 같다.	왼쪽이 크거나 같으면 참이다.
<=	작거나 같다.	왼쪽이 작거나 같으면 참이다.

3. 논리 연산자

- 논리 연산자는 주로 여러 가지 조건을 복합적으로 사용하며 &&(그리고), ||(또는), !(부정)가 쓰임
- 예를 들어 a라는 값이 100과 200 사이에 들어 있어야 한다면 'a는 100보다 크다. 그리고 a는 200보다 작다.'라고 표현할 수 있음

```
(a > 100 ) && (a < 200)
```

- 참이 되려면 (a > 100)도 참이 되어야 하고 (a < 200)도 참이 되어야 함
- 즉 a가 100과 200 사이에 있어야만 두 조건 모두 참이 됨

표 4-4 논리 연산자

연산자	의미		사용 예	설명
&&	~ 이고	그리고(AND)	(a > 100) && (a < 200)	둘 다 참이어야 참이다.
	~ 이거나	또는(OR)	(a > 100) (a < 200)	둘 중 하나만 참이어도 참이다.
!	~ 아니다	부정(NOT)	!(a == 100)	참이면 거짓 거짓이면 참이다.

4. 비트 연산자

- 비트 연산자는 정수나 문자 등을 2진수로 변환한 후 각 자리의 비트끼리 연산을 수행

표 4-5 비트 연산자

연산자	명칭	설명
&	비트 논리곱(AND)	둘 다 1이면 1이다.
	비트 논리합(OR)	둘 중 하나만 1이면 1이다.
^	비트 배타적 논리합(XOR)	둘이 같으면 0, 둘이 다르면 1이다.
~	비트 부정	1은 0으로, 0은 1로 변경한다.
<<	비트 왼쪽 시프트(이동)	비트를 왼쪽으로 시프트(이동)한다.
>>	비트 오른쪽 시프트(이동)	비트를 오른쪽으로 시프트(이동)한다.