

Chapter 3

연산자



01

Question

두 개의 정수를 입력받아 다음과 같이 출력하는 프로그램을 작성하시오.

입력 예 두 개의 수를 입력하시오. 7 5

출력 예

$$\begin{aligned} 7 + 5 &= 12 \\ 7 - 5 &= 2 \\ 7 * 5 &= 35 \\ 7 / 5 &= 1 \\ 7 \% 5 &= 2 \end{aligned}$$

Chapter 3

연산자

소스

```
#include <stdio.h>

int main()
{
    int a, b;
    int plus, minus, multi, div, rest;

    printf("두 개의 수를 입력하십시오. ");
    scanf("%d %d", &a, &b);

    plus = a + b;
    minus = a - b;
    multi = a * b;
    div = a / b;
    rest = a % b;

    printf("%d + %d = %d \n", a, b, plus);
    printf("%d - %d = %d \n", a, b, minus);
    printf("%d * %d = %d \n", a, b, multi);
```

Chapter 3

연산자

```
printf("%d / %d = %d \n", a, b, div);  
printf("%d %% %d = %d \n", a, b, rest);  
  
return 0;  
}
```

설명

- 대입연산자

이미 배운 것처럼 변수에 값을 대입하는 연산자를 "대입 연산자"라고 한다.

'='의 왼쪽에 오른쪽의 내용을 대입하는 것이다.

" $a = b + c$ "라고 하면 $b + c$ 의 결과를 a 에 저장하는 것이다.

따라서, 왼쪽에는 수식이나 상수를 사용할 수 없으며 반드시 한 개의 변수만을 사용할 수 있다.

오른쪽에는 왼쪽 변수와 같은 자료형의 변수, 상수, 수식이 모두 사용 가능하다.

- 산술연산자

계산식으로 나타내어 결과 값을 돌려주는 연산자를 "산술연산자"라고 하며 다음과 같은 종류가 있다.

연산자	기능	사용법	의미
+	덧셈	$a = b + c$	b와 c를 더한 결과를 a에 대입한다.
-	뺄셈	$a = b - c$	b에서 c를 뺀 결과를 a에 대입한다.
*	곱셈	$a = b * c$	b와 c를 곱한 결과를 a에 대입한다.
/	몫	$a = b / c$	b를 c로 나눈 몫을 a에 대입한다.
%	나머지	$a = b \% c$	b를 c로 나눈 나머지를 a에 대입한다.

- * 정수의 연산에서 '/' 은 소수 이하를 모두 버린 몫만을 돌려주며 실수의 경우에는 소수점 이하를 모두 계산하여 돌려준다.
- * '%' 연산자는 정수의 연산에서만 사용된다.
- * '/'나 '%' 연산에서 제수(c)가 0이 되는 경우에는 에러가 발생하므로 사용시 주의해야 한다.
- * 정수와 실수가 혼합된 연산은 실수의 연산으로 처리된다.

- `printf(" %% ")`
'%'는 바로 뒤의 문자(열)과 합쳐져서 자료의 형을 나타내기 위한 문자로 사용된다.
만약 '%' 문자를 화면에 출력하려면 "%%"로 해주어야 된다.

자가진단 1

세 개의 정수를 입력 받아서 합계와 평균을 출력하는 프로그램을 작성하시오 (단, 평균은 소수 이하를 버리고 정수부분만 출력한다.)

입력 예 10 25 33

출력 예 sum : 68

avg : 22

02

Question

5개의 정수를 입력 받아 각각의 수에 $+ 3$, $- 3$, $\times 3$, $/ 3$, $\% 3$ 의 연산을 실행하여 그 값을 저장한 후 차례대로 출력하는 프로그램을 작성하시오.

입력 예 5개의 수를 입력하시오. 10 5 7 9 10

출력 예 13 2 21 3 1

소스1

```
#include <stdio.h>

int main()
{
    int a, b, c, d, e;

    printf("5개의 수를 입력하시오. ");
    scanf("%d %d %d %d %d", &a, &b, &c, &d, &e);

    a = a + 3;
    b = b - 3;
    c = c * 3;
    d = d / 3;
    e = e % 3;

    printf("%d %d %d %d %d \n", a, b, c, d, e);

    return 0;
}
```

설명

- $a = a + 3;$

만약 수학적식이라면 위의 식은 성립되지 않는다. 프로그램에서는 위 문장의 의미는 “a 변수의 값에 3을 더하여 a변수에 대입(저장)하라.”는 의미가 된다. 즉, 변수 a의 값이 3증가하는 것이다.

소스2

```
#include <stdio.h>

int main()
{
    int a, b, c, d, e;

    printf("5개의 수를 입력하시오.");
    scanf("%d %d %d %d %d", &a, &b, &c, &d, &e);

    a += 3;
    b -= 3;
    c *= 3;
    d /= 3;
    e %= 3;

    printf("%d %d %d %d %d \n", a, b, c, d, e);

    return 0;
}
```

설명

- `a += 3;`

“왼쪽의 변수(a)에 3을 더하여 다시 왼쪽의 변수에 대입(저장)하라.”는 의미로써 “`a = a + 3`”의 식과 같은 표현이다. 이와 같이 대입 연산자와 산술 연산자를 함께 사용하여 값을 변화시키는 연산자를 복합 대입연산자라고 한다.

연산자	사용법	의미
<code>+=</code>	<code>a += b</code>	“ <code>a + b</code> ”의 결과를 <code>a</code> 에 대입, “ <code>a = a + b</code> ”와 같다.
<code>-=</code>	<code>a -= b</code>	“ <code>a - b</code> ”의 결과를 <code>a</code> 에 대입, “ <code>a = a - b</code> ”와 같다.
<code>*=</code>	<code>a *= b</code>	“ <code>a * b</code> ”의 결과를 <code>a</code> 에 대입, “ <code>a = a * b</code> ”와 같다.
<code>/=</code>	<code>a /= b</code>	“ <code>a / b</code> ”의 결과를 <code>a</code> 에 대입, “ <code>a = a / b</code> ”와 같다.
<code>%=</code>	<code>a %= b</code>	“ <code>a % b</code> ”의 결과를 <code>a</code> 에 대입, “ <code>a = a % b</code> ”와 같다.

자가진단 2

정수 2개를 입력받아서 첫 번째 수에는 100을 증가시켜 저장하고, 두 번째 수는 10으로 나눈 나머지를 저장한 후, 두 수를 차례로 출력하는 프로그램을 작성하시오.

입력 예 20 35

출력 예 120 5

03

Question

정수형 변수 a와 b를 만들고 각각 10으로 초기화한 후 다음과 같이 출력하는 프로그램을 작성하시오

출력 예 최초값 a = 10, b = 10

a++ = 10, ++b = 11

실행후 a = 11, b = 11

a-- = 11, --b = 10

실행후 a = 10, b = 10

Chapter 3

연산자

소스

```
#include <stdio.h>

int main()
{
    int a = 10, b = 10;

    printf("최초값 a = %d, b = %d \n\n", a, b);

    printf("a++ = %d, ++b = %d \n", a++, ++b);
    printf("실행 후 a = %d, b = %d \n\n", a, b);

    printf("a-- = %d, --b = %d \n", a--, --b);
    printf("실행 후 a = %d, b = %d \n", a, b);

    return 0;
}
```

설명

- `int a = 10, b = 10;`
정수형 변수 `a`와 `b`를 선언하고 각각 10으로 초기화 한다. (변수를 선언하고 즉시 값을 대입하는 것을 "초기화한다."고 한다.)
- `printf("%d %d", a++, ++b);`
"`a++`"와 "`++a`"는 둘 다 `a`의 값을 1 증가시킨다.
즉 "`a = a + 1`" 또는 "`a += 1`"과 같다.
"`a++`"는 먼저 `a`의 값을 출력한 후 값을 증가시키고
"`++b`"는 `b`의 값을 먼저 증가시킨 후 출력한다.
따라서 위의 식에서 출력되는 값은 10과 11이 되지만,
출력한 후 `a`, `b`의 값은 모두 11이 된다.

설명

"a++"와 같이 연산자가 뒤에 붙은 것을 "후치 증가 연산자"라고 하고
"+b"와 같이 연산자가 앞에 붙은 것을 "전치 증가 연산자"라고 한다.
"a--"와 "--b"의 경우도 같은 방법으로 사용되며
각각 "후치 감소 연산자", "전치 감소 연산자"라고 한다.
결과적으로 위의 문장은 다음 세 문장을 한 개의 문장으로 표현한 것과 같다.

```
b = b + 1;  
printf("%d %d", a, b);  
a = a + 1
```

연산자	이름	기능
a++	후치 증가 연산자	작업을 끝내고 a의 값을 1증가시킴
++a	전치 증가 연산자	a의 값을 1 증가시킨 후 작업을 실시
a--	후치 감소 연산자	작업을 끝내고 a의 값을 1감소시킴
--a	전치 감소 연산자	a의 값을 1 감소시킨 후 작업을 실시

- 전치(++a)와 후치(a++) 증가 연산자의 사용 예

예1)

```
a = 1;  
b = ++a; //a가 1증가하고 b에 값을 저장  
printf("%d", b); // 2 출력  
printf("%d", a); // 2 출력
```

예2)

```
a = 1;  
b = a++; //b에 a의 값을 저장하고 a가 1증가  
printf("%d", b); // 1 출력  
printf("%d", a); // 2 출력
```

자가진단 3

한 개의 정수를 입력 받아서 후치증가 연산자를 사용하여 출력한 후, 전치 증가 연산자를 사용하여 출력하는 프로그램을 작성하시오

입력 예 5

출력 예 5

7

04

Question

정수 변수 a와 b를 입력받아서 a는 전치 증가 연산자를 사용하고, b는 후치 감소 연산자를 사용하여 두 수의 합을 c에 저장한 후 각각을 출력하는 프로그램을 작성하시오.

입력 예 5 6

출력 예 a = 6, b = 5, c = 12

소스

```
#include <stdio.h>

int main()
{
    int a, b, c;

    scanf("%d %d", &a, &b);

    c = ++a + b--;
    printf("a = %d, b = %d, c = %d \n", a, b, c);

    return 0;
}
```

설명

- `c = ++a + b--;`

a가 먼저 1 증가한 후 a와 b의 합을 c에 저장하고 나서 b가 1이 감소한다.

중감연산자의 이해를 위해 위와 같이 작성은 했지만 별로 바람직한 방법은 아니다.

위의 결과를 여러 개의 식으로 나타내면 다음과 같다.

```
a = a + 1; (또는 a++;)
```

```
c = a + b;
```

```
b = b - 1; (또는 b--;)
```

자가진단 4

두 개의 정수를 입력받아 각각 후치 증가 연산자와 전치 감소 연산자를 사용하여 두 수의 곱을 새로운 변수에 저장한 후 각각의 값을 출력하는 프로그램을 작성하시오.

입력 예 10 20

출력 예 11 19 190

05

Question

3개의 정수 a, b, c를 입력받아서 a와 b, b와 c를 각각 비교하여 같으면 1, 같지 않으면 0을 출력하고, 다음에는 같지 않으면 1, 같으면 0을 출력하는 프로그램을 작성하시오

입력 예 10 20 20

출력 예 0 1 1 0

Chapter 3

연산자

소스

```
#include <stdio.h>

int main()
{
    int a, b, c;
    int result1, result2, result3, result4;

    scanf("%d %d %d", &a, &b, &c);

    result1 = (a == b);
    result2 = (b == c);
    result3 = (a != b);
    result4 = (b != c);

    printf("%d %d %d %d \n", result1, result2, result3, result4);

    return 0;
}
```

설명

- `result1 = (a == b);`
"a == b"는 "a와 b가 같은가?"라는 의미이며
참(True)이면 1, 거짓(False)이면 0을 돌려준다.
따라서 `result1`에는 a와 b가 같으면 1, 아니면 0이 대입된다.

a=10, b=20이면 `result1`에는 0이 대입된다.

- `result3 = (a != b);`
"a != b"는 "a와 b가 같지 않은가?"라는 의미이며
참(True)이면 1, 거짓(False)이면 0을 돌려준다.

`result3`에는 a와 b가 같지 않으면 1, 아니면 0이 대입된다.
입력의 예에서 a와 b는 같지 않으므로 `result3`에는 1이 대입된다.

자가진단 5

두 개의 정수를 입력받아서 두 정수의 값이 같으면 1, 아니면 0을 출력하고 두 번째 줄에는 같지 않으면 1, 아니면 0을 출력하는 프로그램을 작성하시오.

입력 예 5 5

출력 예 1

0

06

Question

3개의 정수 a , b , c 를 입력받아서 a 가 b 보다 큰지, b 가 c 보다 크거나 같은지, a 가 b 보다 작거나 같은지, b 가 c 보다 작은지를 비교하여 참이면 1, 거짓이면 0을 각각 출력하는 프로그램을 작성하시오

입력 예 1 2 2

출력 예 0 1 1 0

Chapter 3

연산자

소스

```
#include <stdio.h>

int main()
{
    int a, b, c;
    int result1, result2, result3, result4;

    scanf("%d %d %d", &a, &b, &c);

    result1 = (a > b);
    result2 = (b >= c);
    result3 = (a <= b);
    result4 = (b < c);

    printf("%d %d %d %d \n", result1, result2, result3, result4);

    return 0;
}
```

설명

- 관계연산자

변수나 숫자 등의 값을 비교하여 참인지 거짓인지를 판단하는 연산자를 관계연산자 또는 비교연산자라고 한다.

조건이 맞으면 "참(True)",
맞지 않으면 "거짓(False)"이라고 하고,
참이면 1, 거짓이면 0 값을 갖게 된다.

설명

- 관계 연산자의 종류는 다음과 같다.

연산자	읽는법	사용법	의미
<code>==</code>	같다.	<code>a == b</code>	a와 b는 같다
<code>!=</code>	같지 않다.	<code>a != b</code>	a와 b는 같지 않다
<code><</code>	작다.(미만)	<code>a < b</code>	a가 b보다 작다
<code>></code>	크다.(초과)	<code>a > b</code>	a가 b보다 크다
<code><=</code>	작거나 같다.(이하)	<code>a <= b</code>	a가 b보다 작거나 같다
<code>>=</code>	크거나 같다.(이상)	<code>a >= b</code>	a가 b보다 크거나 같다

자가진단 6

두 개의 정수를 입력받아서 다음과 같이 4가지 관계연산자의 결과를 출력하시오.

입력 예 4 5

출력 예 4 > 5 --- 0

4 < 5 --- 1

4 >= 5 --- 0

4 <= 5 --- 1

07

Question

3개의 정수 a , b , c 를 선언하여 각각 0, 1, 2로 초기화한 후 a 와 b 가 모두 참인지, a 또는 b 가 참인지, b 와 c 가 모두 참인지, a 가 참이 아닌지를 확인하여 참이면 1, 거짓이면 0을 각각 출력하는 프로그램을 작성하시오.

출력 예

0 1 1 1

소스

```
#include <stdio.h>

int main()
{
    int a = 0, b = 1, c = 2;
    int result1 = (a && b);
    int result2 = (a || b);
    int result3 = (b && c);
    int result4 = !a;

    printf("%d %d %d %d \n", result1, result2, result3, result4);

    return 0;
}
```

설명

- 논리연산자

두 개 이상의 조건을 조합하여 논리적인 결과를 확인할 수 있는 연산자를 논리연산자라고 하며 다음과 같은 종류가 있다.

연산자	읽는법	사용법	의미
&&	그리고(and), 논리곱	$a \ \&\& \ b$	a도 참이고 b도 참이다. (모두 참이러야 참)
	또는(or), 논리합	$a \ \ b$	a가 참이거나 b가 참이다. (둘 중 어느 하나라도 참이면 참)
!	~가 아니다.(not)	$!a$	a가 참이 아니다. (a가 참이면 거짓, 거짓이면 참이 된다.)

Chapter 3

연산자

- `result1 = (a && b);`
둘 다 참이어야 하는데 `a`가 거짓(0)이므로 결과값은 거짓(0)이 된다.
- `result2 = (a || b);`
둘 중 하나인 `b`가 참(1)이므로 결과값은 참(1)이 된다.
'|' 표시는 키보드에서 "Shift + 원문자(W)"를 누르면 입력된다.



- `result3 = (b && c);`
b와 c가 모두 참이므로 결과값은 참(1)이다.
※ 0이 아닌 모든 값은 참이다.
- `result4 = !a;`
a는 원래 거짓(0)이므로 결과값은 참(1)이 된다.

자가진단 7

2개의 정수를 입력 받아서 논리곱과 논리합의 결과를 출력하는 프로그램을 작성하시오

입력 예 2 0

출력 예 0 1

08

Question

3개의 정수 a, b, c를 선언하여 각각 0, 1, 2로 초기화 한 후 아래 소스와 같은 다양한 논리 연산 결과를 출력하는 프로그램을 작성하시오.

출력 예 1 1 0 1

Chapter 3

연산자

소스

```
#include <stdio.h>

int main()
{
    int a = 0, b = 1, c = 2;
    int result1 = (a < b) && (b < c);
    int result2 = (a && b) || (b && c);
    int result3 = (a || !c);
    int result4 = (a != b) && (b != c);

    printf("%d %d %d %d \n", result1, result2, result3, result4);

    return 0;
}
```


설명

- `result1 = (a < b) && (b < c);`

논리연산자에는 변수뿐 아니라 상수, 수식, 관계식, 논리식 등이 모두 사용될 수 있다. `(a < b)` 가 참이고 `(b < c)` 도 참이므로 `(True && True)` 논리연산의 결과는 참이 된다.

- `result2 = (a && b) || (b && c);`

`(a && b)`는 `a`가 거짓이므로 거짓, `(b && c)`는 둘 다 참이므로 참, 따라서 `(False || True)` 논리연산의 결과는 참이 된다.

- `result3 = (a || !c);`

`a`는 거짓, `!c`는 거짓(`c`가 참이기 때문에)이므로 `(False || False)` 논리연산의 결과는 거짓이 된다.

- `result4 = (a != b) && (b != c);`

`a`와 `b`는 같지 않으므로 참, `b`와 `c`도 같지 않으므로 참이므로 `(True && True)` 논리연산의 결과는 참이 된다.

자가진단 8

3개의 정수를 입력 받아 첫 번째 수가 가장 크면 1, 아니면 0을 출력하고 세 개의 수가 모두 같으면 1, 아니면 0을 출력하는 프로그램을 작성하시오

입력 예 10 9 9

출력 예 1 0

Thank You!!!

