

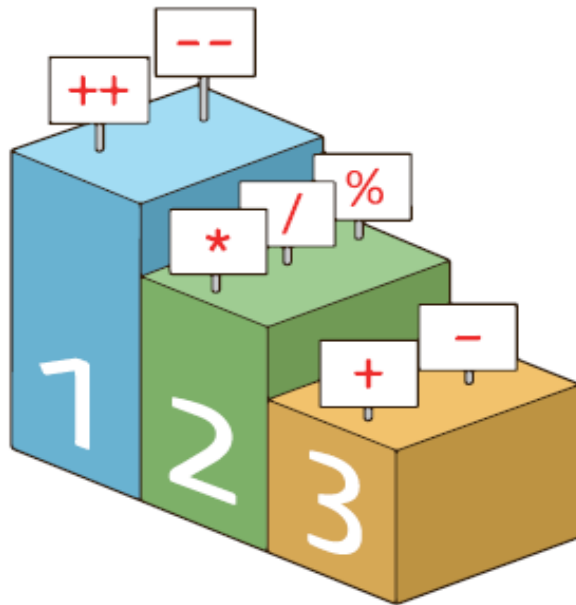
JAVA 프로그램 실습

연산자 우선순위

경북대학교
소프트웨어융합과
배희호

Operator Precedence

- 연산 수식 내에 두 종류 이상의 Operator가 포함되어 있는 경우 수식에서 Operator의 평가 순서



우선 순위는 어떤 연산자를
먼저 계산할지를 나타내는
순위입니다.



$$x + y * z$$

The diagram shows the expression $x + y * z$ with curly braces indicating evaluation order. A brace labeled '1' is under the $y * z$ part, and a larger brace labeled '2' is under the entire $x + y * z$ expression.

Operator Precedence

- 복합 연산 수식에서 어느 Operator가 먼저 계산 되는지를 결정하고자 할 때 사용하는 규칙
 - Precedence(우선순위)
 - Associativity(결합규칙)

$$2 + 3 * 4 * 5 \Leftrightarrow$$

$$2 + 3 + 4 + 5 \Leftrightarrow$$

Operator Precedence

순위	명칭		연산자	결합규칙
1	1차 수식		(), [], .	->
2	단항 연산자		!, -, ~, ++, --	<-
3	이항 연산자	승제	*, /, %	->
4		가감	+, -	
5		비트 이동	<<, >>, >>>	
6		대소	<, >, <=, >=, instanceof	
7		동등	==, !=	
8		비트 AND	&	
9		비트 XOR	^	
10		비트 OR		
11		논리 곱	&&	
12		논리 합		
13	조건 연산자 (삼항 연산자)		? :	<-
14	대입 연산자		=, +=, -=, *=, /=, %=, <<=, >>=, &=, :=, ^=	
15	coma 연산자		,	



Operator Precedence

- 수식에서 소괄호(())를 사용하여 연산 우선순위를 조정
- 일차 수식이 가장 우선함
- 단항 연산자가 2항 연산자보다 우선함
- 2항 연산자는 3항 연산자보다 우선함
- JAVA 언어의 유일한 3항 연산자는 조건 연산자
- 3항 연산자 다음은 대입 연산자
- 2항 연산자에서는
 - 사칙 연산자 (승제, 가감)
 - 관계 연산자 (비교, 등가)
 - 논리 연산자 (AND, OR)
- 우선순위가 가장 낮은 연산자는 콤마 연산자

- 연산자의 일반적인 연산 진행방향은 왼쪽에서 오른쪽(→)
- 단, 단항, 대입 연산자만 오른쪽에서 왼쪽(←)

Operator Precedence

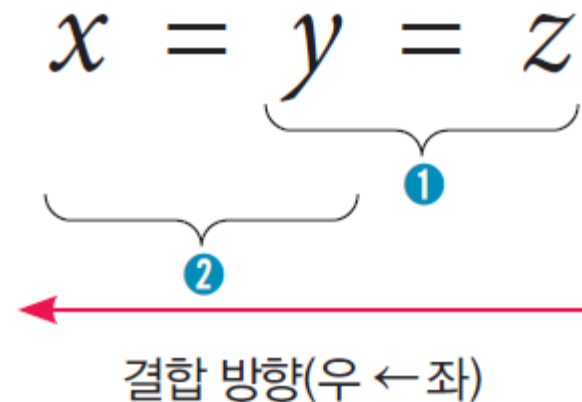
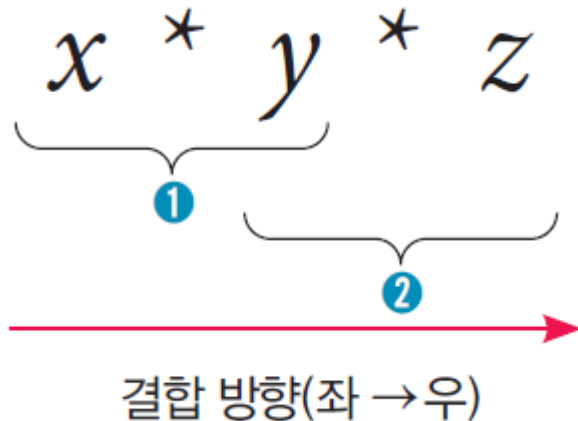
■ Operator의 주요 우선 순위

대입 < 조건 < 논리 < 비트 < 관계 < 이동 < 산술 < 단항

주요 단어	의미
괄호	괄호가 가장 우선순위가 높음. 따라서 연산의 우선 순위가 기억이 나지 않는다면 괄호를 사용해서 연산 우선권을 강제로 지정하는 것이 좋음
단항	모든 단항 연산자는 어느 이항 연산자보다 먼저 계산
이항	대입 연산자를 제외한 모든 이항 연산자는 삼항 연산자인 조건 연산자(? :)보다 먼저 계산
대입	콤마 연산자를 제외하고는 대입 연산자를 제일 나중에 계산
산술	산술 연산자의 우선순위가 관계 연산자나 논리 연산자보다 높음

Operator Associativity

- 동일한 Precedence의 연산이 있는 경우에 무엇을 먼저 수행하느냐에 대한 규칙



Operator Associativity

- Associativity(결합규칙)이란 동일한 Precedence의 Operator들간에 수식을 평가하는 방향을 결정하는 규칙
 - Left Associativity(좌측 결합규칙)
 - Right Associativity(우측 결합규칙)
- 한 수식에서 어떤 피 연산자가 먼저 평가되는지는 System에 따라서 다름
- 함수에 있어서 인수의 평가 순서는 System에 의존적
- 일반적으로 수식의 왼쪽에서 오른쪽으로 수행
- 모든 단항 연산자와 삼항 연산자는 오른쪽에서 왼쪽으로 수행
- 모든 대입 연산자는 오른쪽에서 왼쪽으로 수행

Operator Associativity

- 만약 같은 Precedence를 갖는 Operator들이 여러 개가 있으면 어떤 것을 먼저 수행하여야 하는가?

```
result = x % y * z;
```

③ ① ②

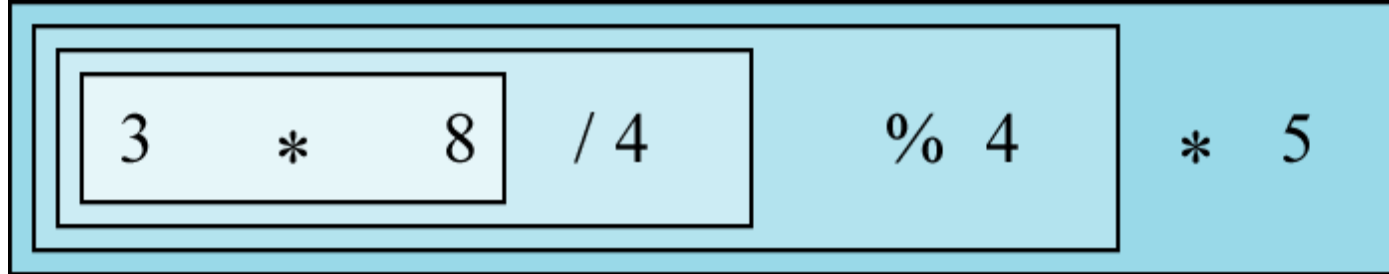
- 대부분의 Operator가 왼쪽에서 오른쪽으로 수행되지만 몇 개의 연산자는 오른쪽에서 왼쪽으로 연산이 진행 (대입 연산자가 대표적)

```
x = y = w = z ;
```

③ ② ①

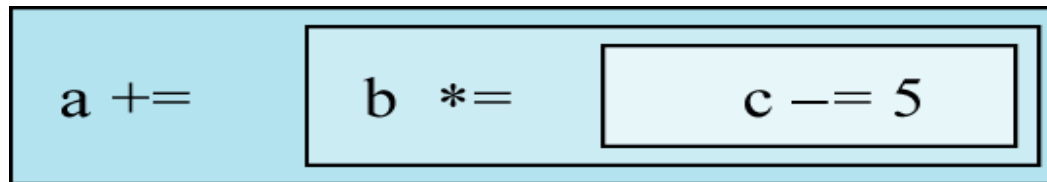
Left Associativity

- 대부분의 Operator에 적용되는 규칙
 - 사칙 연산자 (+, -, *, /, %)
 - 관계 연산자 (>, >=, <, <=, ==, !=)
 - 논리 연산자 (!, &&, ||)
 - 비트 연산자 (^, &, |, >>, <<)



Right Associativity

- 대입 연산자에 적용되는 규칙
 - 대입 연산자 (=)
 - 산술 대입 연산자 (+=, -=, *=, /=, %=)
- 예) $a = b = c = d = 0;$



Precedence 예제 1

- 어떤 Operator를 먼저 계산할 것인가?

```
result = x * y % z - a / b
```

5 1 2 4 3

- 괄호 안은 먼저 계산됨

```
m = ( x + y + z ) / 3
```

4 1 2 3

- Associativity

$3 * 4 * 5 \rightarrow ((3 * 4) * 5)$

$x = y = 3 \rightarrow (x = (y = 3))$

Precedence 예제 2

■ 다음 연산 수식에서 피 연산자들이 평가되는 순서는?

① $a + b + c + d + e$

② $a + b * c - d / e$

③ $a / (b + c) - d \% e$

④ $a / (b * (c + (d - e)))$

$$a + b + c + d + e$$

1 2 3 4

$$a + b * c - d / e$$

3 1 4 2

$$a / (b + c) - d \% e$$

2 1 4 3

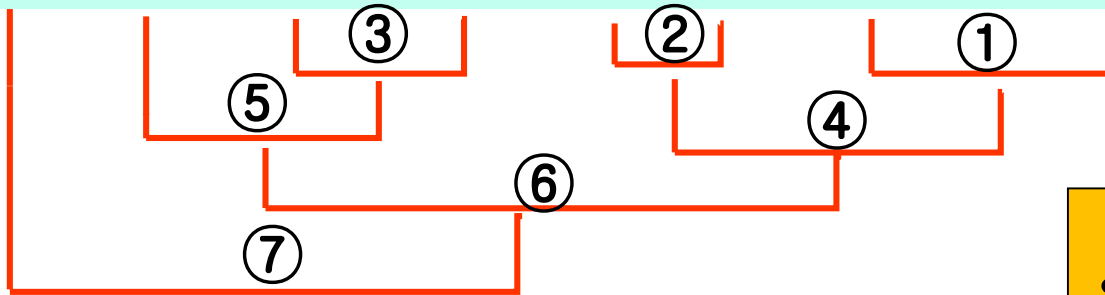
$$a / (b * (c + (d - e)))$$

4 3 2 1

Precedence 예제 3

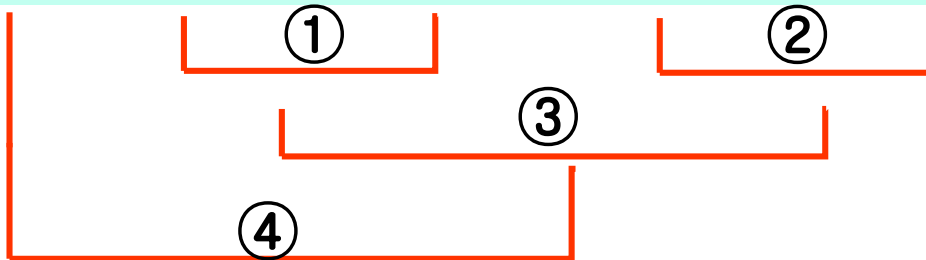
- 우선순위가 같은 사칙 연산자들끼리는 어떤 순서로 처리될까요?

$a = b - c * d - ++e * (f *= 3);$



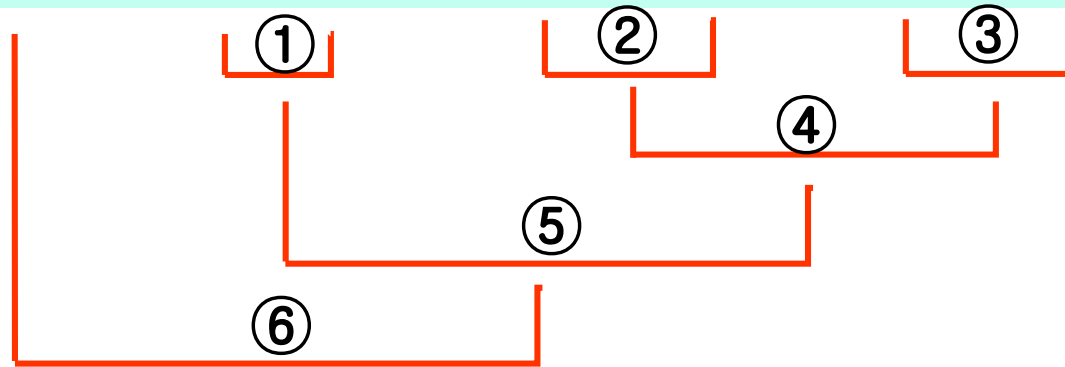
이런 연산 수식에서는
왼쪽 연산자부터 순서대로
처리됨

$m = (k || n) \&\& (a \&\& b);$



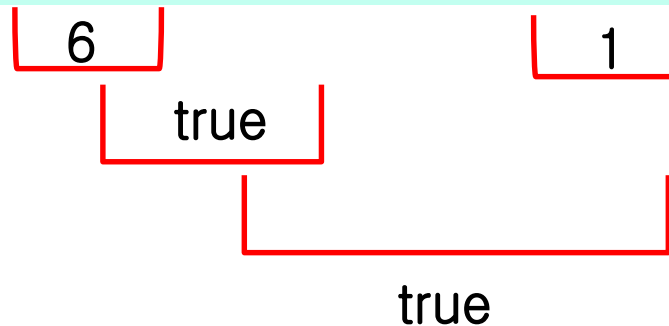
Precedence 예제 4

```
test = !flag || (y + z >= x - z);
```



b = 5일 때,

```
int result = 2 * 3 > b == 4 - 3;
```



Precedence 예제 5

■ 다음의 연산 수식의 연산 순서를 설명하여라

- ① $-x + 3$
- ② $x + 3 * y$
- ③ $x + 3 > y - 2$
- ④ $x > 3 \ \&\& \ x < 5$
- ⑤ `int result = x + y * 3;`

- ① 단항 > 이항이므로 $((-x) + 3)$
- ② 곱셈, 나눗셈 > 덧셈, 뺄셈이므로 $(x + (3 * y))$
- ③ 산술 > 비교이고, 왼쪽부터 연산하므로 $((x + 3) > (y - 2))$
- ④ 비교 > 논리이고, 왼쪽부터 연산하므로 $((x > 3) \ \&\& \ (x < 5))$
- ⑤ 대입은 맨 끝이므로 `int (result = (x + (y * 3)));`

Precedence 예제 6

■ 다음 문장의 연산 우선 순위를 표시하여라

① $y = x + y - z ;$

② $y = -x ;$

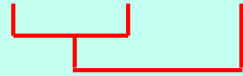
③ $y = -x++ ;$

④ $y = -++x ;$

⑤ $y = -x + z ;$

Precedence 예제 6

$y = x + y - z ;$



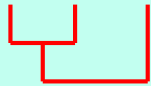
// 좌측 결합

$y = -x ;$



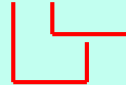
// 우측 결합

$y = -x++ ;$



// x의 값에 단항 - 연산을 적용한 후
y에 배정하고 x를 증가

$y = -++x ;$



// x를 증가한 후 x의 값에 단항
- 연산을 적용한 후 y에 배정

$y = -x + z ;$



// x의 값에 단항 - 연산한 후 z를 더하
고 그 결과를 y에 배정

Precedence 예제 7

- 다음 연산 수식의 연산 순서를 설명하여라

$x \ll 2 + 1 \rightarrow (x \ll (2 + 1))$

비트 이동 연산자(<<, >>, >>>)는 사칙 연산자보다 우선순위가 낮음

$x < -1 \ || \ x > 3 \ \&\& \ x < 5$

$\rightarrow ((x < -1) \ || \ ((x > 3) \ \&\& \ (x < 5)))$

||, |(OR)는 &&, &(AND)보다 우선순위가 낮음

Precedence 예제 8

- 다음 수식의 연산자 우선 순위를 괄호를 사용하여 표시하여 보자

문제1	$-e \% c - a / k * 3 + 5$
해답	$((((-e) \% c) - ((a / k) * 3)) + 5)$
문제2	$i = j = k = -10$
해답	
문제3	$k \% = a = a + b / 2$
해답	
문제4	$++c / d++ * --e$
해답	
문제5	$p \% 5 == 0 ? i : p + 1$
해답	

Precedence 예제 9

- 다음 수식의 연산자 우선 순위를 괄호를 사용하여 표시하여 보자

문제6	$a + b + d == -2 * i$
해답	
문제7	$w / !!z$
해답	
문제8	$a b \&\& k - 5$
해답	
문제9	$x != j \&\& j + 1 == !m + 4$
해답	
문제10	$j == 2 n == 4 i == 3$
해답	

Precedence 예제 10

- 다음 수식의 연산자 우선 순위를 괄호를 사용하여 표시하여 보자

문제11	$a + c \leq !d + b$
해답	
문제12	$!(a \leq b + 3)$
해답	
문제13	$x \leq \text{sizeof}(\text{int}) + 2 / 3 + !3$
해답	
문제14	$'a' + \text{sizeof}('a') < c$
해답	
문제15	$--a + ---b$
해답	

Precedence 예제 11

- 다음 수식의 연산자 우선 순위를 괄호를 사용하여 표시하여 보자

문제11	$-x + 3$
해답	
문제12	$x + 3 * y$
해답	
문제13	$x + 3 > y - 2$
해답	
문제14	$x > 3 \ \&\& \ x < 5$
해답	
문제15	<code>int result = x + y * 3;</code>
해답	

Precedence 예제 12

■ 다음 연산 수식의 연산의 순서를 적으시오

$$(1) x = y = 3 / 5 * 2 \% 6;$$

$$(2) y = a * x * x + b * x + c;$$

Precedence 예제 13

- 다음 연산 수식의 연산 결과를 구하여라

$$\text{exp} = 1 - -2 * 3 + 4 * 5 / 6;$$

The diagram illustrates the evaluation order of the expression $\text{exp} = 1 - -2 * 3 + 4 * 5 / 6;$ based on operator precedence. The operations are performed as follows:

- First, $-2 * 3$ is calculated, resulting in -6 .
- Next, $4 * 5$ is calculated, resulting in 20 .
- Then, $20 / 6$ is calculated, resulting in 3 .
- Finally, the expression is simplified to $1 - -6 + 3$, which is calculated as $1 + 6 + 3 = 10$.

산술 연산자의 계산 순서는 수학과 동일

Precedence 예제 14

- 다음 연산 수식의 연산 결과를 구하여라

int a = 1, b = 2, c = 3, d = 4;		
수 식	동일한 수식	결 과
a * b / c		
a * b % c + 1		
++a * b - c--		
7 - --b * ++d		

Precedence 예제 15

- 다음 연산 수식의 연산 결과를 구하여라

int i = 1, j = 2, k = 3, m = 4;		
수 식	동일한 수식	결 과
i += j + k		
j *= k = m + 5		

Precedence 예제 16

- 다음 연산 수식의 연산 순서를 설명하여라

```
int i = 2;  
int j = 2;  
int a = 2;  
int b = 2;  
int a += i * j + ++b;
```

순위	연산자	연산내용	값의 변화
1	++	++b	b = 3
2	*	i * j	4
3	+	1의 결과 + 2의 결과	7
4	+=	a += 3의 결과	9

Precedence 예제 17

- 다음 연산 수식의 연산 과정의 데이터 형을 설명하여라

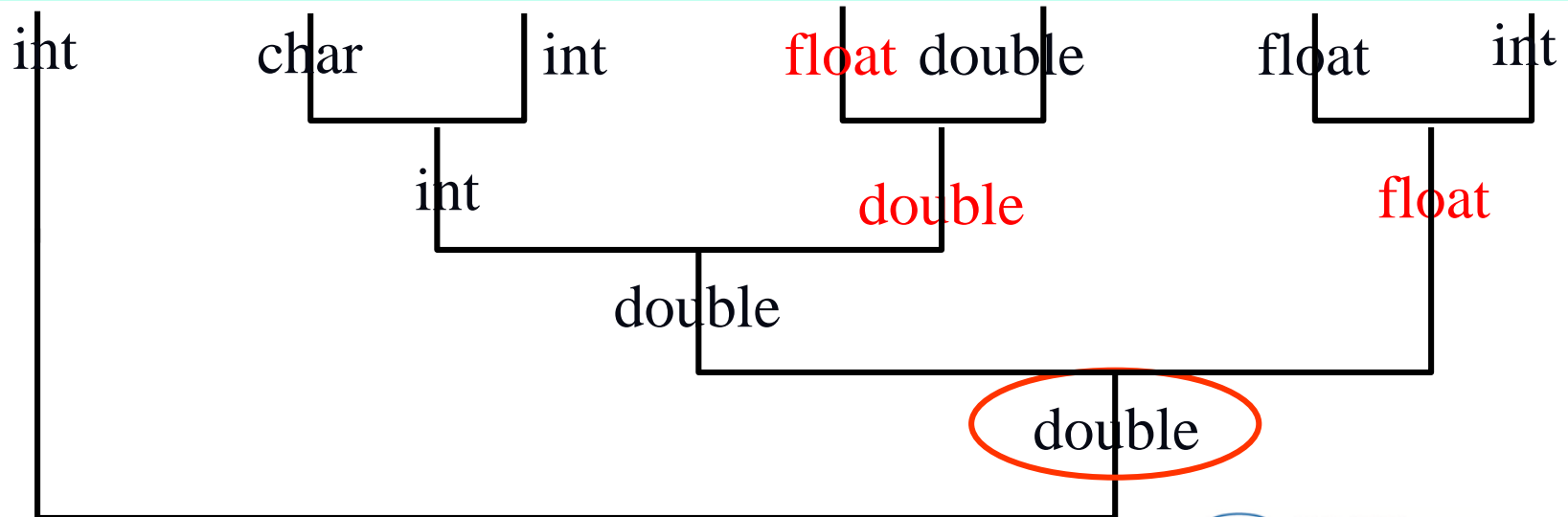
```
char    ch;
```

```
int     i;
```

```
float   f;
```

```
double d, int result;
```

```
result = ( ch  /  i )  +  ( f  *  d )  -  ( f  +  i );
```



Precedence 예제 18

- 다음 Program의 실행 결과를 설명하여라

```
public static void main(String args[]) {  
    int a = 2, b = 3, c = 5;  
    double d = 4.0, e = 0.5, f;  
  
    f = a + b == c ? d : e;  
    System.out.println(f);  
}
```

4.0

$f = ((a + b) == c) ? d : e;$

Precedence 예제 19

- 다음 Program의 실행 결과를 설명하여라

```
public static void main(String[] args) {  
    int y;  
  
    y = 10 + 5 * 2;  
    System.out.println("y = 10 + 5 * 2 = " + y );  
    y = (10 + 5) * 2;  
    System.out.println("y = (10 + 5) * 2 = " + y );  
}
```

$y = 10 + 5 * 2 = 20$
 $y = (10 + 5) * 2 = 30$

Precedence 예제 20

- 다음 Program의 실행 결과를 설명하여라

```
public static void main(String[] args) {  
    Scanner keyboard = new Scanner(System.in);  
    int num;  
  
    System.out.print("정수 입력 : ");  
    num = keyboard.nextInt();  
    String str = (num % 2 == 0) ? "짝수 " : "홀수 ";  
  
    System.out.print("정수형 변수 num의 값은 " + num + " 이며 ");  
    System.out.println(str + "입니다");  
}
```

정수 입력 : 5

정수형 변수 num의 값은 5 이며 홀수 입니다

Precedence 예제 21

■ 다음 Program의 실행 결과를 설명하여라 ?

```
public static void main(String[] args) {  
    char ch = 'A';  
    short value = 1;  
    int test;  
    long sample;  
  
    test = ch + value;  
    sample = test + value;  
  
    System.out.printf("A = %d\n", (int) ch);  
    System.out.println(test);  
    System.out.println(sample);  
}
```

A = 65
66
67



Precedence 예제 22

■ 다음 Program의 실행 결과를 설명하여라 ?

```
public static void main(String[] args) {  
    int a, b, i, j;  
  
    a = b = i = j = 2;  
    System.out.println(" a = " + a + ", b = " + b + ", i = " + i + ", j = " + j);  
    System.out.println();  
  
    a += i * j + ++b;  
    System.out.println(" a += i * j + ++b 를 계산 후 : ");  
    System.out.println(" a = " + a);  
    System.out.println(" b = " + b);  
    System.out.println();  
  
    a = b = i = j = 3;  
    System.out.println(" a = " + a + ", b = " + b + ", i = " + i + ", j = " + j);  
    System.out.println();  
}
```

Precedence 예제 22

```
a += i * j + b-- >> 1;  
System.out.println(" a += i * j + b-- >> 1를 계산 후 :");  
System.out.println(" a = " + a);  
System.out.println(" b = " + b);  
}
```

a = 2, b = 2, i = 2, j = 2

a += i * j + ++b 를 계산 후 :

a = 9

b = 3

a = 3, b = 3, i = 3, j = 3

a += i * j + b-- >> 1를 계산 후 :

a = 9

b = 2



Futuristic Innovator

京福大學校
KYUNGBOK UNIVERSITY

Precedence 예제 23

■ 다음 Program의 실행 결과를 설명하여라 ?

```
public static void main(String[] args) {  
    int test = 12;  
    double d = 34.5444444444;  
  
    double test1 = test + d;  
    float test2 = (float) (test + d);  
    System.out.println("12 + 34.5 = " + (12 + 34.5444444444));  
    System.out.println("12 + 34.5 = " + test1);  
    System.out.println("12 + 34.5 = " + test2);  
    System.out.printf("%d + %.1f = %.15f", test, d, test1);  
}
```

12 + 34.5 = 46.5444444444

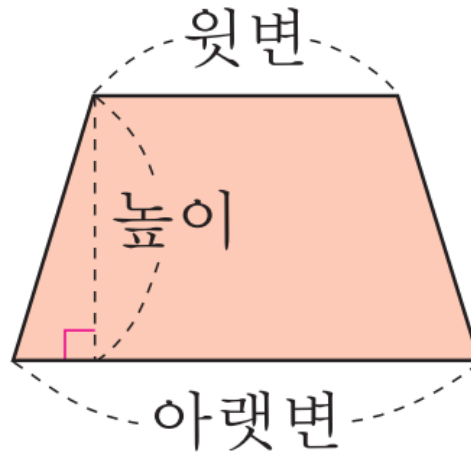
12 + 34.5 = 46.5444444444

12 + 34.5 = 46.544445

12 + 34.5 = 46.5444444444000000

도입 예제

- 사다리꼴의 넓이를 구하는 Program을 작성하여라

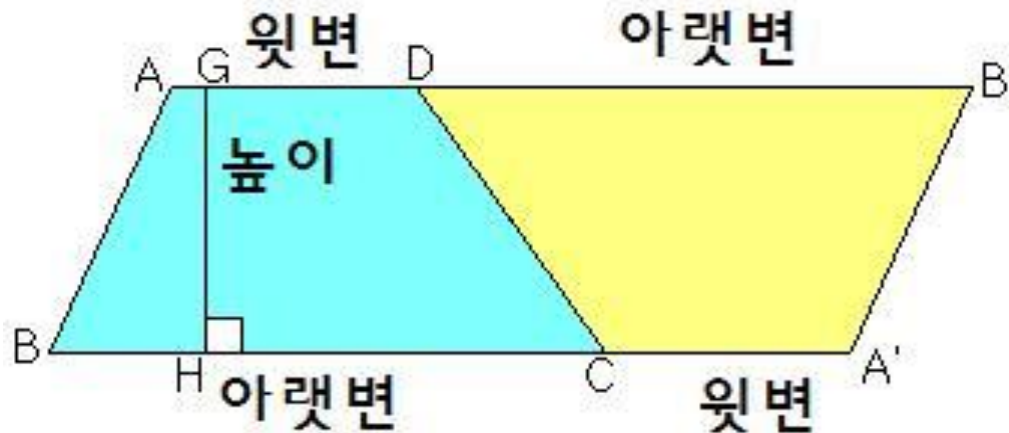


$$\begin{aligned} &(\text{사다리꼴의 넓이}) \\ &= \{(\text{윗변}) + (\text{아랫변})\} \times (\text{높이}) \div 2 \end{aligned}$$

도입 예제

■ 문제 분석

- 사다리꼴 하나를 180도 돌려서 옆에 붙이면, 다음과 같이 평행 사변형 모양이 됨



- 평행 사변형의 넓이 = (윗변 + 아랫변) × 높이
- 사다리꼴은 그 넓이의 절반
 - 사다리꼴의 넓이 = $\frac{1}{2} \times (\text{윗변} + \text{아랫변}) \times \text{높이}$

도입 예제

■ 문제 분석

- Data는 무엇이고, Information는 무엇인가?

■ 입력

■ 사다리꼴 3변

- 윗변(upper) – int
- 밑변(bottom) – int
- 높이(height) – int

사다리꼴 윗변 :	245
사다리꼴 밑변 :	678
사다리꼴 높이 :	456
사다리꼴 면적 :	210,444.00

■ 출력

- 사다리꼴 면적(area) – float

■ 계산 방법

- 사다리꼴 넓이 = (윗변 + 밑변) X 높이 / 2

도입 예제

```
public static void main(String[] args) {  
    Scanner keyboard = new Scanner(System.in);  
    int upper, bottom, height;  
    float area;  
  
    System.out.print("사다리꼴의 윗변 입력 : ");  
    upper = keyboard.nextInt();  
    System.out.print("사다리꼴의 밑변 입력 : ");  
    bottom = keyboard.nextInt();  
    System.out.print("사다리꼴의 높이 입력 : ");  
    height = keyboard.nextInt();  
  
    area = ((float) upper + bottom) * height / 2;  
    System.out.printf("사다리꼴 윗변 : %,10d", upper);  
    System.out.printf("사다리꼴 밑변 : %,10d", bottom);  
    System.out.printf("사다리꼴 높이 : %,10d", height);  
    System.out.printf("사다리꼴 면적 : %,10.2f", area);  
}
```


Heart

- 사람은 평균 분당 65회 심장이 박동한다
- 사람의 평균 수명은 78년이다.
- 그러면 사람은 평생 몇 회의 심장 박동을 하는지 구해보자.



Heart

- 문제 분석

- Data는 무엇이고, Information는 무엇인가?

- 입력

- 심장 박동(heart) – int (분당 65회)

- 수명(life) – int (78 년)

- 출력

- 심장 박동수(count) – long

- 1년이 몇 분(minute) – int

- 계산 방법

- $\text{minute} = 1\text{년}이\ 몇\ 분인지\ 구함$

- $\text{count} = \text{heart} * \text{minute} * \text{life}$

- 약 26억 번 심장 박동함

Heart

```
public static void main(String[] args) {  
    final int HEART = 65;  
    final int LIFE = 78;  
    int minute;  
    long count;  
  
    minute = 365 * 24 * 60;  
    count = HEART * LIFE + minute;  
  
    System.out.printf("1년은 %,d 분이다.\n", minute);  
    System.out.printf("나는 평생 %,d 년을 산다.\n", LIFE);  
    System.out.printf("심장은 1분에 %,d번 작동합니다.\n", HEART);  
    System.out.printf("나는 평생 %,d 번의 심장이 박동한다.\n", count);  
}
```

1년은 525,600 분이다.
나는 평생 78 년을 산다.
심장은 1분에 65번 작동합니다.
나는 평생 -1,630,175,296 번의 심장이 박동한다.

결과가 이상하지 않나요?
왜 이상한 결과가 나오는지 설명하고, 수정하여라.

Heart

■ 오류 원인

- 연산 수식을 평가하는 과정에서 각 단계별 피 연산자의 데이터 형을 서로 보다 큰 데이터형으로 일치 시키고, 연산의 결과도 일치시킴
- 연산 순서에 따라서 데이터형 을 점검해보면 모든 데이터 형이 intg형이므로 중간 결과도 int 형의 범위를 넘지 못하므로 발생
- 캐스트 연산자를 사용할 때는 전체 수식에서 연산 순위가 가장 높은 연산자의 피 연산자에 하는 것이 효과적임

Heart

```
public static void main(String[] args) {  
    final int HEART = 65;  
    final int LIFE = 78;  
    int minute;  
    long count;  
  
    minute = 365 * 24 * 60;  
    count = (long) HEART * LIFE * minute;  
  
    System.out.printf("1년은 %,d 분이다.\n", minute);  
    System.out.printf("나는 평생 %,d 년을 산다.\n", LIFE);  
    System.out.printf("심장은 1분에 %,d번 작동합니다.\n", HEART);  
    System.out.printf("나는 평생 %,d 번의 심장이 박동한다.\n", count);  
}
```

1년은 525,600 분이다.

나는 평생 78 년을 산다.

심장은 1분에 65번 작동합니다.

나는 평생 2,664,792,000 번의 심장이 박동한다.

Gold

- 당신의 체중을 입력 받아 금값으로 계산하면 나는 얼마 일까요? (1,000원 미만은 버림)



Gold

■ 문제 분석

- Data는 무엇이고, Information는 무엇인가?

■ 입력

- 당신의 체중(weight) – float

- 금 1g의 값(gold) – int

■ 출력

- 나의 가치(price) – long

■ 계산 방법

- 1Kg = 1,000 g

- 금값(gold) : 1돈 (3.75 g) : 316,000원

- $\text{Price} = (\text{몸무게}(\text{weight}) : \text{gram으로 환산}) * \text{금값}(\text{gold})$

- 1,000원 미만은 버림 (나누기와 곱셈을 이용)

Gold

```
public static void main(String[] args) {  
    Scanner keyboard = new Scanner(System.in);  
    float weight;  
    int gold;  
    final float UNIT = 3.75f;  
    float gram;  
    int price;  
  
    System.out.print("당신의 몸무게 ? ");  
    weight = keyboard.nextFloat();  
    System.out.printf("금 1돈의 가격 : ");  
    gold = keyboard.nextInt();  
    gram = weight * 1000;  
    price = gram / UNIT * gold;  
    price /= 1000;  
    price *= 1000;  
}
```



Gold

```
System.out.printf("몸무게 : %.1f Kg(%.2f g)\n", weight, gram);  
System.out.printf("금 1돈(%.2f g)의 가격 : %,d 원\n", UNIT, gold);  
System.out.printf("몸무게 가격 = %,d 원\n", price);  
}
```

■ 오류 원인

Gold

```
public static void main(String[] args) {  
    Scanner keyboard = new Scanner(System.in);  
    float weight;  
    int gold;  
    final float UNIT = 3.75f;  
    float gram;  
    int price;
```

```
    System.out.print("당신의 몸무게 ? ");  
    weight = keyboard.nextFloat();  
    System.out.printf("금 1돈의 가격 : ");  
    gold = keyboard.nextInt();  
    gram = weight * 1000;  
    price = (int) (gram / UNIT * gold);  
    price /= 1000;  
    price *= 1000;
```

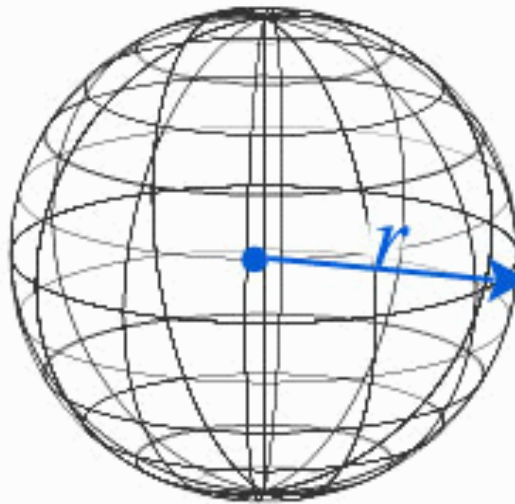
당신의 몸무게 ? 78.4
금 1돈의 가격 : 316000
몸무게 : 78.4 Kg(78,400.00 g)
금 1돈(3.75 g)의 가격 : 316,000 원
몸무게 가격 = 2,147,483,000 원



Korea

- 지구(Earth)의 표면적에서 대한민국이 얼마를 차지하고 있는가 ?
- 지구의 반지름 : 6,378,137m
- 대한민국 넓이 : 99,720km²

구



구의 겉넓이 O 와 부피 V

$$O = 4 \times \pi \times r^2$$

$$V = \frac{4}{3} \times \pi \times r^3$$

© doopedia.co.kr



Futuristic Innovator

京福大學校
KYUNGBOK UNIVERSITY

Korea

■ 문제 분석

- Data는 무엇이고, Information는 무엇인가?

■ 입력

- 지구의 반지름 : 6,378,137m

- 대한민국 넓이 : 99,720km²

■ 출력

- 대한민국이 지구 표면적의 몇 %를 점유 – float

■ 계산 방법

- 지구 표면적 구함

- 대한민국의 비율 계산

- 단위를 고려함

Korea

```
public static void main(String[] args) {  
    final long RADIUS = 6378137L;  
    final double PI = 3.141592;  
    final double AREA = 99720 * 1000;  
    double surface;  
    double korea;  
  
    System.out.println(" 대한민국의 표면적을 구해보자");  
    surface = 4 * PI * RADIUS * RADIUS;  
    korea = (AREA / surface) * 100;  
  
    System.out.printf(" 지구의 표면적 : %,2f \Wu33A1\Wn", surface);  
    System.out.printf(" 한국의 면적 : %,2f \Wu33A1\Wn", AREA);  
    System.out.printf(" 지구의 표면적중에 대한민국은 %.8f %%를  
                                                                차지하고 있다.\Wn ", korea);  
}
```

Korea

대한민국의 표면적을 구해보자

지구의 표면적 : 511,207,787,042,028.70 m²

한국의 면적 : 99,720,000.00 m²

지구의 표면적중에 대한민국은 0.00001951 %를 차지하고 있다.

Cigarette

- 담배 1개피를 피우면 수명이 7분 단축 된다고 합니다. 평생 담배를 피우면 얼마의 시간이 수명 단축 될까요 ?
- 단 1년은 365일로 함



Cigarette

- 문제 분석

- Data는 무엇이고, Information는 무엇인가?

- 입력

- 담배 1개피 수명 단축 : 7분 - int
 - 하루에 몇 개피 피우는지 - int
 - 몇 년 동안 피우는지 - int

- 출력

- 단축된 수명 int

- 계산 방법

- 담배 1갑 = 20 개피
 - 1년, 하루 24시간, 1시간 60분

Cigarette

```
public static void main(String[] args) {  
    Scanner keyboard = new Scanner(System.in);  
    final int LIFE = 7;          /* 담배 1대의 단축 생명 7분 */  
    final int YEAR = 365;       /* 1년은 365일 */  
    int life;                   /* 담배 피운 기간 */  
    int tobacco;                /* 1일 담배 소비량 (개피) */  
    int piece;                  /* 소비한 담배 수량 */  
    long minute;                /* 담배로 인한 수명단축 (분) */  
    int day;                    /* 담배로 인한 수명단축 (날) */  
  
    System.out.print("담배를 몇년동안 피우셨나요 ? ");  
    life = keyboard.nextInt();  
    System.out.print("담배를 하루에 몇개피 피우시나요 ? ");  
    tobacco = keyboard.nextInt();  
}
```



Cigarette

```
piece = life * YEAR * tobacco;
minute = piece * LIFE;
day = (int) (minute / (60 * 24));
int year = day / YEAR;
day -= (year * YEAR);
minute -= ((year * YEAR + day) * 24 * 60);
int hour = (int) (minute / 60);
minute -= (hour * 60);
System.out.printf("담배를 하루에 %d개피씩 ", tobacco);
System.out.printf("%d년 동안 피우면 ", life);
System.out.printf("수명단축은 %d년 %d일 %d 시간 %d분 입니다.\n",
year, day, hour, minute);
}
```

담배를 몇년동안 피우셨나요 ? 30

담배를 하루에 몇개피 피우시나요 ? 15

담배를 하루에 15개피씩 30년 동안 피우면 수명단축은 2년 68일 10
시간 30분 입니다.

Cigarette[심화]

- 앞의 Program에 담배 1갑의 가격을 입력 받아 담배 피우는데 필요한 비용을 계산하여야보자.

Cigarette[심화]

■ 문제 분석

- Data는 무엇이고, Information는 무엇인가?

■ 입력

- 담배 1개피 수명 단축 : 7분 - int
- 하루에 몇 개피 피우는지 - int
- 몇 년 동안 피우는지 - int
- 1갑의 금액

■ 출력

- 단축된 수명 int
- 비용

■ 계산 방법

- 담배 1갑 = 20 개피, 한 갑씩 거래한다
- 1년, 하루 24시간, 1시간 60분

Cigarette[심화]

```
public static void main(String[] args) {
    Scanner keyboard = new Scanner(System.in);
    final int LIFE = 7;           /* 담배 1대의 단축 생명 7분 */
    final int YEAR = 365;         /* 1년은 365일 */
    final int CASE = 20;
    int life;                     /* 담배 피운 기간 */
    int tobacco;                 /* 1일 담배 소비량 (개피) */
    int piece;                   /* 소비한 담배 수량 */
    long minute;                 /* 담배로 인한 수명단축 (분) */
    int day;                     /* 담배로 인한 수명단축 (날) */
    int price;

    System.out.print("담배를 몇년동안 피우셨나요 ? ");
    life = keyboard.nextInt();
    System.out.print("담배를 하루에 몇개피 피우시나요 ? ");
    tobacco = keyboard.nextInt();
    System.out.print("담배 가격은 ? ");
    price = keyboard.nextInt();
}
```

Cigarette[심화]

```
piece = life * YEAR * tobacco;
int box = piece / CASE + (piece % CASE == 0 ? 0 : 1);
minute = piece * LIFE;
day = (int) (minute / (60 * 24));
int year = day / YEAR;
day -= (year * YEAR);
minute -= ((year * YEAR + day) * 24 * 60);
int hour = (int) (minute / 60);
minute -= (hour * 60);

System.out.printf("담배를 하루에 %d개피씩 ", tobacco);
System.out.printf("%d년 동안 피우면 %d개피를 피우는데", life, piece);
System.out.printf("수명단축은 %d년 %d일 %d 시간 %d분 입니다.\n",
                    year, day, hour, minute);
System.out.printf("평생 구매한 담배 값 : %,d 값(%,d개피)\n",
                    box, box*CASE);
System.out.printf("담배 비용은 %,d원 입니다.\n", box * price);
}
```

Cigarette[심화]

담배를 몇년동안 피우셨나요 ? 30

담배를 하루에 몇개피 피우시나요 ? 15

담배 가격은 ? 4500

담배를 하루에 15개피씩 30년 동안 피우면 164250개피를
피우는데수명단축은 2년 68일 10 시간 30분 입니다.

평생 구매한 담배 갑 : 8,213 갑(164,260개피)

담배 비용은 36,958,500원 입니다.