



주문식교육의 산실
영진전문대학교

구조적 프로그래밍 구성 요소 (연산자)

일본IT과

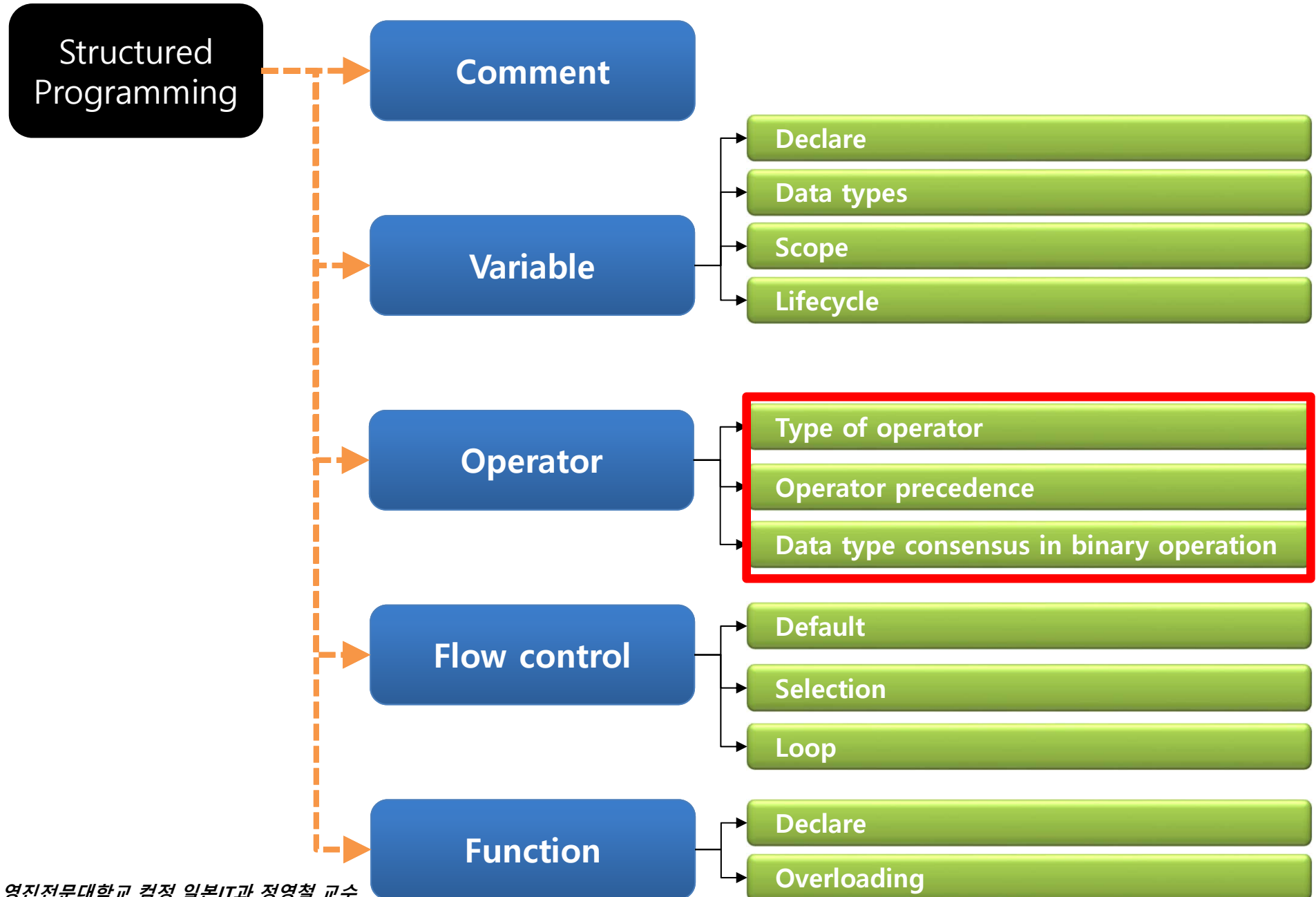
정영철 교수



영진전문대학교 컴퓨터정보계열

SCHOOL OF COMPUTER INFORMATION

구조적 프로그래밍 구성 요소



연산 (Operation) 이란?

- 연산(Operation) 이란
 - 수(Number)나 식(Expression)을 일정한 규칙에 따라 계산 하는 것.
 - 예)

$$1 + 1 = ?$$

Arithmetic operator

$$3 < 4 = ?$$

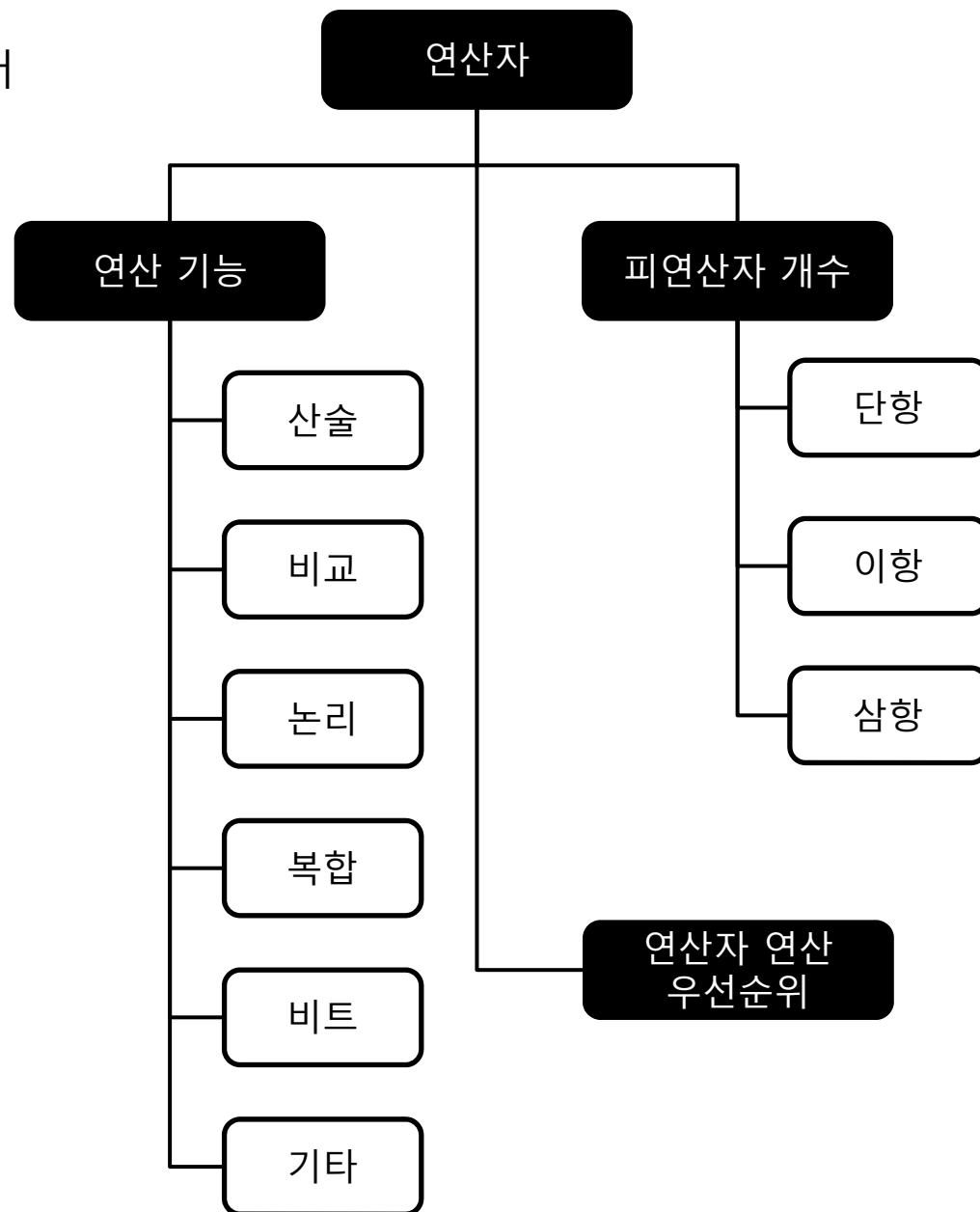
Comparison operator

$$2 > 3 \text{ or } 4 > 1 = ?$$

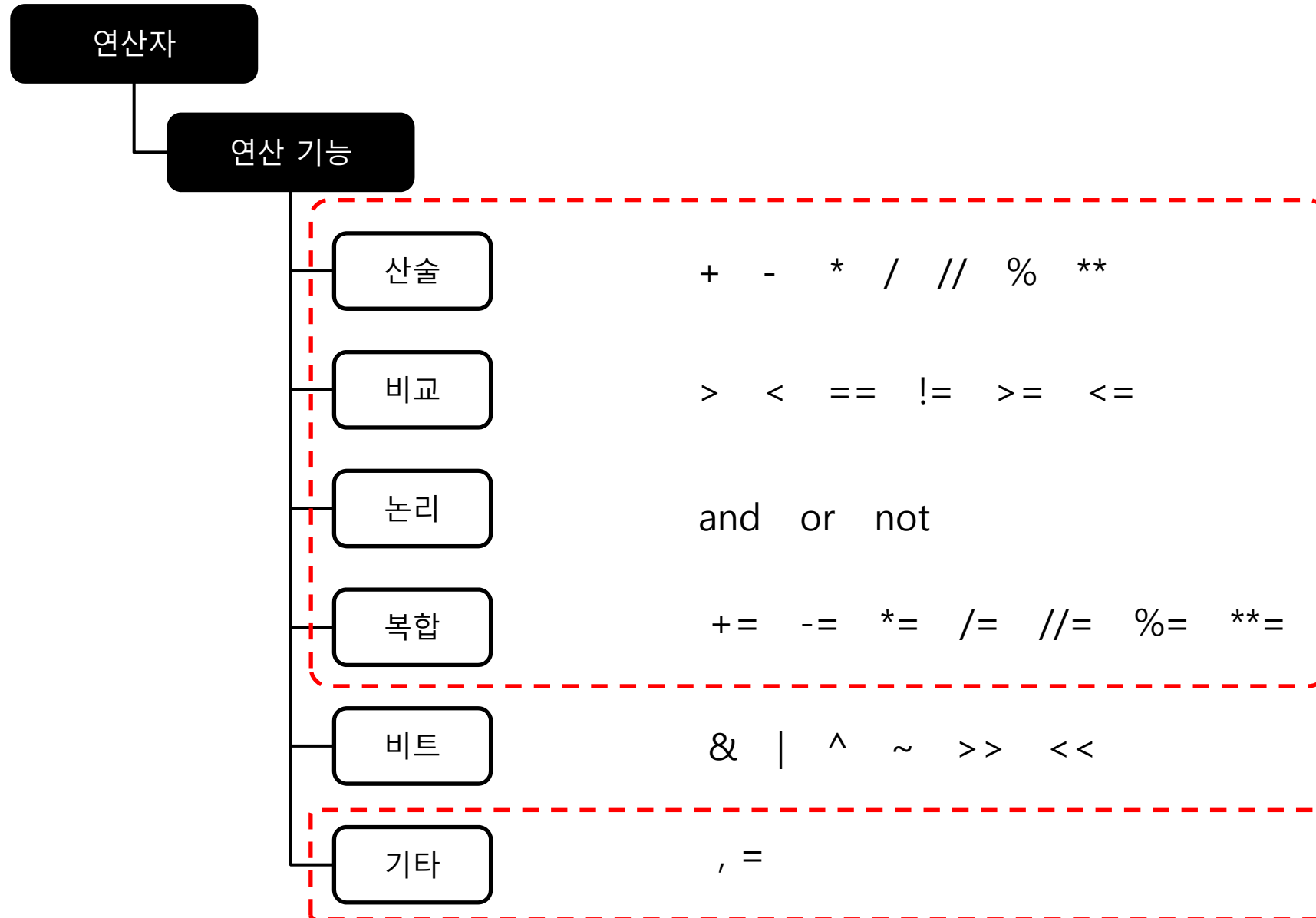
Logical operator

연산자 (Operator) 란?

- 연산자(Operator) 란
 - 연산 기능을 제공하는 명령어



연산자 종류 :



연산자 종류 : 산술연산자 (1)

```
1  # 산술 연산자 -> 수를 계산하는 연산자
2  # 기본 사칙연산과 나머지(%) 또는 몫 구하기(//), 지수(**)
3
4  result_1 = 1 + 3.5
5  result_2 = 2 - 5.5
6  result_3 = 3 * 7
7  result_4 = 5 / 2
8
9  # 출력 값 : 4.5   -3.5   21   2.5
10 print(result_1, result_2, result_3, result_4)
11
12 # 출력 값 : <class 'float'> <class 'float'> <class 'int'> <class 'float'>
13 print(type(result_1), type(result_2), type(result_3), type(result_4))
```

연산자 종류 : 산술연산자 (2)

```
1  # 몫(/), 나머지(%) 연산자
2  # // : 나누기 연산 후 몫 값만 반환
3
4  result_1 = 3 // 2
5  result_2 = 3 / 2
6
7  # 출력 값 : 1    1.5
8  print(result_1, result_2)
9
10 # % : 나누기 연산 후 나머지 값 반환 -> Modulo 연산자
11 # 출력 값 : 0  1  2  0  1  2
12 for divisor in range(6):
13     print(divisor%3)
14
15 # 나머지 연산은 특정 패턴을 찾기위해 빈번하게 사용
16 # 예) 특정 반복문 내에서 3번째 반복마다 특정 명령어 실행
17 count = 1
18
19 for dan in range(2, 10):
20     for num in range(1, 10):
21         print(dan, " X ", num, " = ", (dan*num))
22
23         if count % 3 == 0:
24             print("=====")
25
26     count += 1
```

연산자 종류 : 산술연산자 (3)

```
1  # 지수 연산자 : **
2
3  # 2의 3승
4  result_1 = 2**3
5
6  # 결과 값 : 8
7  print(result_1)
8
9  # 결과 값 : 1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024
10 for value in range(11):
11     print(2**value)
```


연산자 종류 : 비교연산자 (1)

```
1  # 비교 연산자 : 좌항과 우항의 값을 비교
2  # 비교 연산자 종류 : >, >=, <, <=, ==, !=
3
4  print(3 > 4)           # false
5  print(2 > 2)           # false
6  print(2.0 >= 2)        # true
7  print(3 < 4)           # true
8  print(4 <= 4)          # true
9  print(1 == 1)          # true
10 print(1 != 2)           # true
11 print(True != False)   # true
12
13 print("a" > "c")        # false
```

연산자 종류 : 비교연산자 (2)

```
1 record = 70
2
3 # 주어진 점수에 대해 A ~ F 등급으로 출력 프로그램
4 # if ~ elif ~ else 문 활용
5
6 ~ if record >= 90:
7     print("A")
8 ~ elif record >= 80:
9     print("B")
10 ~ elif record >= 70:
11     print("C")
12 ~ elif record >= 60:
13     print("D")
14 ~ else:
15     print("F")
```

연산자 종류 : 논리연산자 (1)

```
1  # 논리 연산자 : 진리표를 활용한 연산 실시
2  # 논리 연산자 종류 : and or not
3
4  # and 연산자
5  #   - 이항 연산자
6  #   - 좌항과 우항의 값이 "참"일때만 참,
7  #   - 그 이외에는 모두 거짓
8  ~if 3 < 2 and 3 < 3:
9      |     print("T 1")
10
11 ~if 3 > 2 and 3 < 3:
12     |     print("T 2")
13
14 ~if 3 < 2 and 3 >= 3:
15     |     print("T 3")
16
17 ~if 3 > 2 and 3 >= 3:
18     |     print("T 4")
```

연산자 종류 : 논리연산자 (2)

```
1  # or 연산자
2  #   - 이항 연산자
3  #   - 좌항과 우항값 중 하나라도 "참"이면 참,
4  #   - 그 이외에는 모두 거짓
5
6  if 3 < 2 or 3 < 3:
7      print("T 1")
8
9  if 3 > 2 or 3 < 3:
10     print("T 2")
11
12 if 3 < 2 or 3 >= 3:
13     print("T 3")
14
15 if 3 > 2 or 3 >= 3:
16     print("T 4")
```

연산자 종류 : 논리연산자 (3)

```
1  # not 연산자
2  #   - 단항 연산자
3  #   - 피연자의 값을 반전 : 참 -> 거짓 또는 거짓 -> 참
4
5  if not(2 > 3):
6      print("T 1")
7  else:
8      print("F 1")
9
10 if not(2 < 3):
11     print("T 1")
12 else:
13     print("F 2")
```

연산자 종류 : 복합연산자

```
1  # 복합(Compound) 연산자
2  #   - 이항 연산자
3  #   - 지정된 연산 후 대입 실시
4
5  value = 1
6
7  value += 1          # value = value + 1
8
9  print(value)
10
11 value *= 3          # value = value * 3
12
13 print(value)
14
15 value /= 3          # value = value / 3
16
17 print(value)
18
19 value **=10         # value = value ** 10
20
21 print(value)
22
23 value //=1000        # value = value // 10
24
25 print(value)
```

실습 (1)

- 1~100까지 정수 중 짝수이면서 8의 배수인 수만 출력 하라.

출력 값

8

16

24

32

40

48

56

64

72

80

88

96

실습 (2)

- 키보드로부터 1~100 사이 성적을 입력 받아 아래 등급으로 변환 후 출력하는 프로그램을 작성하라

95이상 : A+

90이상 : A

85이상 : B+

80이상 : B

75이상 : C+

70이상 : C

65이상 : D+

60이상 : D

60미만 : F

Q/A

감사합니다



주문식교육의 산실
영진전문대학교