

컴퓨터 및 프로그래밍입문 3주차

한상곤(sangkon@pusan.ac.kr)

3. 제어문

3.1 순차문

3.2 ~~if~~ 조건문

3.3 ~~if-else~~ 조건문

3.4 if-elif-else 문

3.5 ~~for~~ 반복문

3.6 중첩 for 루프

3.7 while 반복문

- 연습문제

2. 학습목표

- 3.1 순차적 실행과 조건문, 반복문과 같은 제어문의 차이를 이해하고 활용할 수 있다.
- 3.2 ~~조건문의 개념과 사용법에 대해 이해한다.~~
- 3.3 ~~조건식에 대해 알아보고 복잡한 조건식을 구성할 수 있다.~~
- 3.4 `if` 문과 `if-else` 문 등 다양한 조건문에 대해 이해한다.
- 3.5 블록과 들여쓰기에 대해 이해하고 중첩된 블록을 구성할 수 있다.
- 3.6 `for` 반복문에 대해 자세히 이해한다.
- 3.7 ~~`for in` 구문과 리스트에 대해 이해하고 활용할 수 있다.~~
- 3.8 이중 `for` 루프에 대해 알아보고 활용한다.
- 3.9 `while` 반복문을 정의하고 사용하는 방법을 이해한다.
- 3.10 ~~`for` 문을 이용하여 작성한 반복문을 `while` 문으로 변경할 수 있다.~~
- 3.11 `break` 와 `continue` 를 이용하여 반복문을 제어할 수 있다.

3.1 순차문 *sequential statements*

먼저 나타나는 코드가 먼저 실행되는 구조

```
num = 100
print('num = ', num)      # 100이 출력됨
num = num + 100
print('num = ', num)      # num에 100이 더해져 200이 출력됨
num = num + 100
print('num = ', num)      # num에 다시 100이 더해져 300이 출력됨
```

3.1 제어문 *control statements*

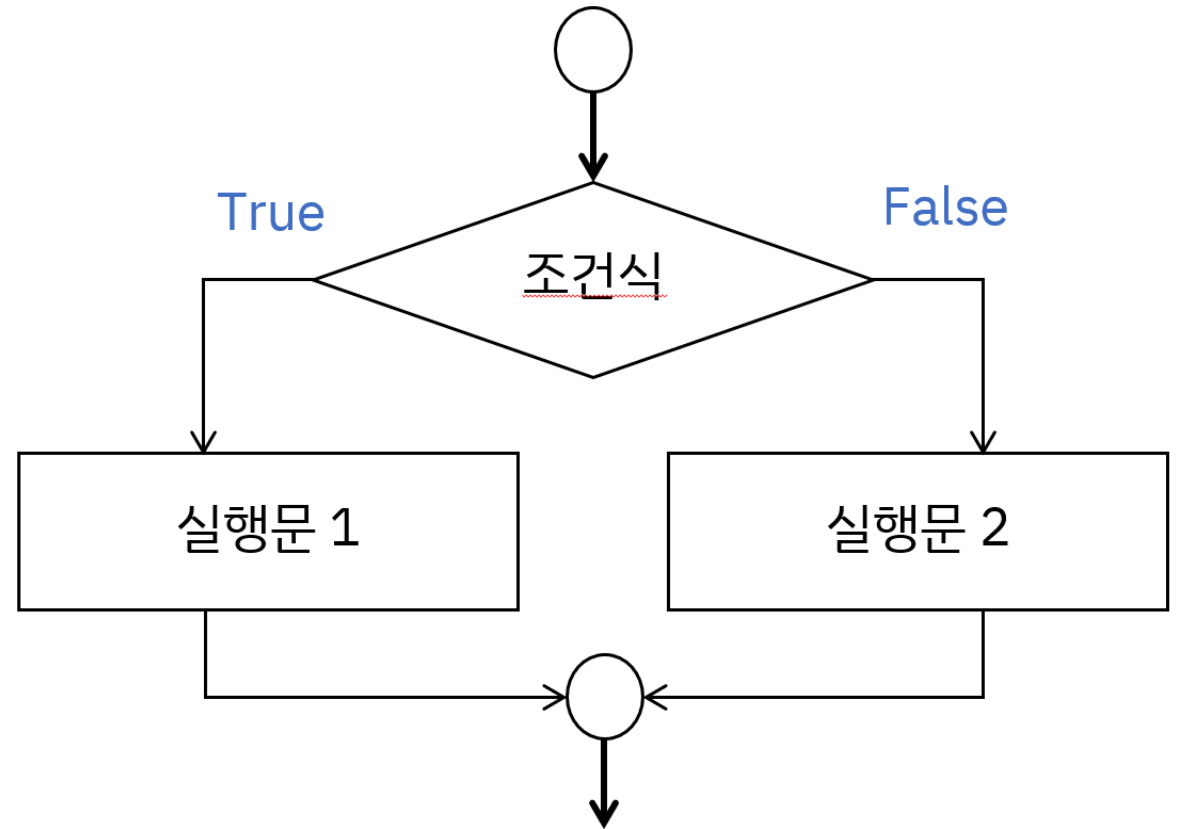
프로그램의 흐름 *flow* 을 제어하는 역할

- 조건문 *conditional, statements*
 - if 문, if-else 문
- 반복문
 - for 문, while 문
- 반복문의 흐름 변경
 - break, continue

3.4 if-elif-else 문

조건문

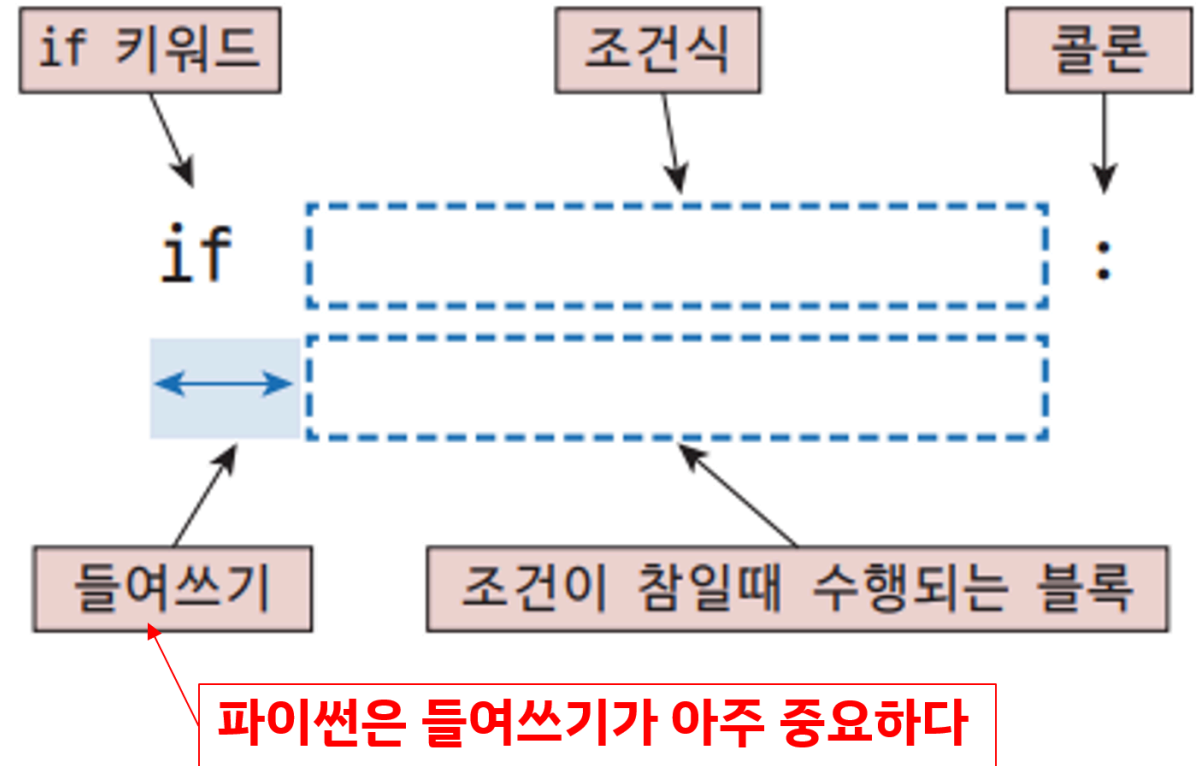
- 실행을 달리하는 여러 개의 실행문이 있음
- 특정한 조건에 따라서 실행됨
- 조건식은 True 혹은 False를 반환



3.4 if-elif-else 문

- (상황 1) - 콜론(:)앞에 나타나는 조건문 절에서 < 연산자를 이용해 나이(age)가 20세 미만인 경우에만 print('청소년 할인')이라는 코드를 실행

```
age = 18                # age가 20 미만의 값
if age < 20:            # age < 20 조건식의 결과는 True임
    print('청소년 할인')
```



3.4 if-elif-else 문

- (상황 2) - 조건문 절에서 `>=` 연산자를 이용해 걸음(walk_count)이 1,000 이상이 되면 `print('목표 달성')`이라는 코드를 실행

```
walk_count = 1200
if walk_count >= 1000:    # walk_count >= 1000 조건식의 결과는 True임
    print('목표 달성')
```

3.4 if-elif-else 문

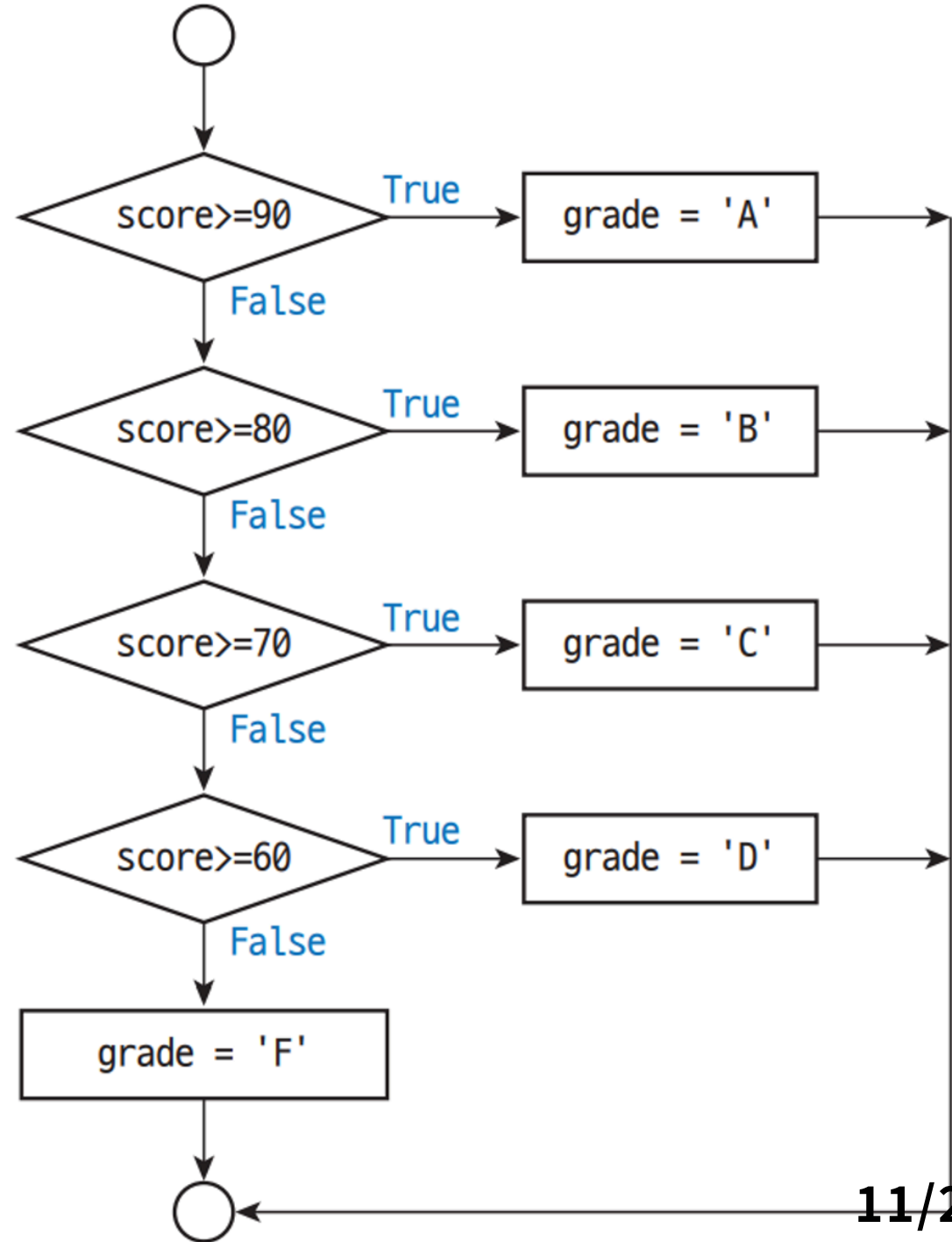
블록 *block*

- 어떤 조건을 만족하는 경우에 특정한 코드를 선택적으로 실행하는 구조
- 실행될 코드 덩어리
- 블록은 반드시 들여쓰기를 해야함
- `:` 이 나오면 다음에 들여쓰기 코드 블록이 나와야 하며 `else`, `elif`, `for`, `while`, `def`, `class` 등에서도 코드 블록이 사용됨

```
age = 18
if age < 20:
print('청소년 할인')      # 들여쓰기 없는 print()문
# IndentationError: expected an indented block
```

3.4 if-elif-else 문

```
score = int(input('점수를 입력하세요 : '))  
if score >= 90:  
    grade = 'A'  
elif score >= 80:  
    grade = 'B'  
elif score >= 70:  
    grade = 'C'  
elif score >= 60:  
    grade = 'D'  
else:  
    grade = 'F'  
print('당신의 등급은 :', grade)
```



3.6 중첩 for 루프

```
for i in range(5):  
    print('Welcome to everyone!!')
```

- range(5)는 range(0, 5)와 같이 주어진 시작 값에서 마지막 값 사이의 연속적인 정수들을 생성할 수도 있으며, range(0, 5, 2)와 같이 마지막에 증가치 값을 넣어 줄 수도 있음
- range(0, 5, 1)과 같이 호출할 경우 마지막의 1은 디폴트 간격(step) 값으로 1씩 더하면서 값을 변경하라는 의미

```
list(range(5))  
list(range(0, 5))  
list(range(0, 10, 2))  
list(range(-2, -10, -2))
```

3.6 중첩 for 루프

- 1~10까지 덧셈

```
s = 0
for i in range(1, 11):
    s = s + i
    print(f'i = {i}, s = {s}')
print('1에서 10까지의 합:', s)
```

- list 순회

```
numbers = [11, 22, 33, 44, 55, 66]
for n in numbers:
    print(n, end = ' ')
```

3.6 중첩 for 루프

```
for i in range(2, 10):           # 외부 for 루프
    for j in range(1, 10):       # 내부 for 루프
        print('{i}*{j} = {i*j:2d}', end = ' ')
    print()                     # 내부 루프 수행 후 줄바꿈을 함
```

3.7 while 반복문

```
n = 10
s = 0
i = 1
while i <= n:
    s = s + i
    i += 1
print(f'1부터 {n}까지의 합은 {s}')
```

3.7 break, continue

```
st = 'Programming'
for ch in st:
    if ch in ['a', 'e', 'i', 'o', 'u']:
        break
    print(ch)
print('The end')
```

```
st = 'Programming'
for ch in st:
    if ch in ['a', 'e', 'i', 'o', 'u']:
        continue
    print(ch)
print('The end')
```


~~과제1. 실습 환경을 구성하여, 캡처해서 제출하세요~~
(~~Exercises1. Configure, capture, and submit your lab experience).~~)

과제2. 아래 연습문제를 풀어서 제출하세요
(Exercises2. Complete and submit the exercise below).

1. 각 문제는 `week3_1.py`, ... `week3_8.py` 과 같이 별도의 파일에 작성하세요
(Write each question in a separate file, such as `week3_1.py`, ... `week3_8.py`, and so on.).
2. 파일은 압축(zip)해서 제출하세요(Please `zip` your files before submitting).

연습문제(a)

- 1 다음과 같이 사용자로부터 3개의 임의의 정수를 입력으로 받아서 가장 작은 수부터 큰수까지 나열하는 프로그램을 `if-else` 문을 사용하여 작성하세요.
- 2 사용자로부터 x, y 좌표를 가진 한 점을 입력으로 받아서 이 점이 1,2,3,4 분면의 어디에 속하는지 알려주는 프로그램을 작성하세요.
- 3 다음과 같이 사용자로부터 1에서 9사이의 숫자를 입력받아 입력받은 숫자의 절에 해당하는 구구단을 출력하는 프로그램을 `for` 문과 `while` 문을 사용하여 각각 작성하여라. 만일 1에서 9이외의 숫자가 입력되면
1에서 9까지의 수를 다시 입력하세요: 라는 안내문을 출력하여라.

```
1에서 9까지의 수를 입력하세요:  
1에서 9까지의 수를 다시 입력하세요:  
...
```

연습문제(b)

- 4 소수란 양의 자연수 중에서 1과 자기 자신이외의 약수를 가지지 않는 수를 말한다. 사용자로부터 양의 정수 n 을 입력받은 후 이 숫자가 소수인지 아닌지를 판별하는 프로그램을 작성하시오(힌트 : n 을 2부터 $n-1$ 까지의 정수로 모두 나누어 본다. 이 때 나누어 떨어지는 원소가 하나라도 있으면 소수가 아니다).

```
숫자를 입력하세요 : 5  
5는 소수입니다
```

```
숫자를 입력하세요 : 9  
9는 소수가 아닙니다
```

연습문제(c)

- **5** 깊이가 30 미터인 우물이 있다. 이 우물에 사는 달팽이는 하루 종일 기어서 올라가면 7 미터를 올라갈 수 있다. 그러나 밤이 되어 휴식을 취하는 동안 5 미터를 미끄러져 내려간다. 이 우물을 벗어나는데 며칠의 시간이 걸릴까? while 문을 이용하여 매일 저녁 달팽이의 위치를 다음과 같이 구하시오(주의 : 하루에 $7 - 5 = 2$ 미터를 올라갈 것이라고 단순 계산하면 안된다. 이 경우 15일 되는 날 탈출하는 것으로 계산되지만 실제로는 13일이면 된다).

```
day : 1    달팽이의 위치 : 7 미터
day : 2    달팽이의 위치 : 9 미터
...
day : 12   달팽이의 위치 : 29 미터
day : 13   달팽이의 위치 : 31 미터
```

우물을 탈출하는 데 걸린 날은 13일 입니다.

연습문제(d)

- **6** 뱀행렬 문제 : 사용자로부터 숫자 1보다 크고 10보다 작은 값 n 을 입력으로 받아서 다음과 같이 뱀의 몸통처럼 증가하는 이차원 배열을 출력하여라. 이 배열은 마치 뱀의 몸통처럼 수가 1씩 증가하는 형태의 배열이라서 뱀행렬이라는 이름을 지었다.

n 을 입력하시오 : 6

1	2	3	4	5	6
12	11	10	9	8	7
13	14	15	16	17	18
24	23	22	21	20	19
25	26	27	28	29	30
36	35	34	33	32	31

연습문제(e)

- 7 암스트롱 수란 세 자리의 정수 중에서 각 자리의 수를 세 제곱한 수의 합과 자신이 같은 수를 말하며 다음의 식으로 정의된다. $xyz = x^3 + y^3 + z^3$ 좀 더, 구체적인 예를 들면 153은 $1 + 125 + 27$ 의 합으로 구성될 수 있는데 이러한 수가 암스트롱 수이다. 세 자리 정수들 중에서 모든 암스트롱 수를 구하여 다음과 같이 출력하여라.

세 자리의 암스트롱 수 : 153 ...

연습문제(5)

- 8 거꾸로 정수는 121이나 3443와 같이 거꾸로 나열해도 그 값이 원래의 값과 같은 정수를 말한다. 사용자로부터 숫자 n 을 입력받아 이 수가 거꾸로 정수인지 아닌지를 판단하는 다음과 같은 프로그램을 작성하시오.

정수를 입력하시오 : 3443

3443은(는) 거꾸로 정수입니다.

정수를 입력하시오 : 324

324은(는) 거꾸로 정수가 아닙니다.

Exercises(a)

- **1** Write a program that takes three random integers as input from the user and lists them from smallest to largest using an `if-else` statement, as shown below.
- **2** Write a program that takes a point with x,y coordinates from a user as input and tells you where it falls in the 1,2,3,4 quadrant.
- **3** Write a program that accepts a number between 1 and 9 from the user and outputs a phrase corresponding to the clause of the number entered, using the following `for` and `while` statements, respectively. If a number other than 1 through 9 is entered, print the message `Please enter a number from 1 through 9 again: .`

Exercises(b)

- 4 A prime number is a positive natural number that does not have a weak factor other than 1 and itself. Write a program that takes a positive integer n from the user and determines whether the number is prime or not (hint: divide n by all the integers from 2 to $n - 1$. If any element of the division falls out, it is not prime).

```
Enter a number : 5
```

```
5 is a prime
```

```
Enter a number : 9
```

```
5 is a prime
```

Exercises(c)

- **5** There is a well that is 30 meters deep. A snail that lives in this well can climb 7 meters by crawling all day long. However, at night, while resting, it slides down 5 meters. How many days will it take to escape the well? Using a while statement, find the snail's position each evening as follows (Note: Do not simply calculate that it will climb $7 - 5 = 2$ meters per day. In this case, it will escape on the 15th day, when in fact it will only take 13 days).

```
day : 1    Snail's position : 7 Meters
day : 2    Snail's position : 9 Meters
...
day : 12   Snail's position : 29 Meters
day : 13   Snail's position : 31 Meters
```

The number of days it took to escape the well is 13.

Exercises(d)

- **6** Serpentine matrix problem: Take as input a value n from the user that is greater than 1 and less than 10 and output a two-dimensional array that grows like the body of a snake, as shown below. This array is called a serpentine matrix because it is shaped like the body of a snake, with numbers increasing by 1.

```
Enter n : 6
```

```
 1   2   3   4   5   6
12  11  10   9   8   7
13  14  15  16  17  18
24  23  22  21  20  19
25  26  27  28  29  30
36  35  34  33  32  31
```

Exercises(4)

- **7** An Armstrong number is the number of three-digit integers whose digits are equal to the sum of the numbers in each digit squared, defined as $xyz = x^3 + y^3 + z^3$. More specifically, for example, 153 can be composed of the sum of $1 + 125 + 27$, which is an Armstrong number. Find all the Armstrong numbers among the three-digit integers and print them out as follows

```
Three-digit Armstrong number : 153 ...
```

Exercises(5)

- 8 An inverted integer is an integer whose value is the same as its original value when listed upside down, such as 121 or 3443. Write the following program that takes a number n from the user and determines whether it is an inverted integer or not.

```
Please enter an integer : 3443
```

```
3443 is an inverted integer.
```

```
Please enter an integer : 324
```

```
324 is not an inverted integer.
```