











학습 목표

6.5 중첩 반복이 무엇인지 이해한다.











중첩 반복(다중 반복)

- 반복문에서 반복 대상은 어떤 문장(예 : 조건문)이든 가능
- 반복문 안에 또 다른 반복문이 오는 경우를 **중첩 반복**이라 함
 - ✓ 중첩 반복은 기존 문법의 단순한 조합





예시) 구구단 출력을 위한 반복문

■ for 문 안에 for 문이 사용된 형태

```
for( i=1 ; i < 10 ; ++i )
{
   for( j=1 ; j < 10 ; ++j )
   {
     printf("%d x %d = %d\n", i, j, i*j );
   }
}</pre>
```

✓ 다른 종류의 반복문도 중첩 가능(예 : for 문 안에 while 문 사용)





구구단(하나의 단출력)

■ 하나의 단은 다음과 같이 반복문으로 표현 가능

```
printf("1 x 1 = %d\n", 1*1 ); ⇒ 1 x 1 = 1

printf("1 x 2 = %d\n", 1*2 ); ⇒ 1 x 2 = 2

... (생략)

printf("1 x 9 = %d\n", 1*9 ); ⇒ 1 x 9 = 9
```







구구단(9개의단출력)

■ 하나의 단 출력을 9번 반복

```
for( j=1 ; j < 10 ; ++j ) → 1단
    printf("1 x %d = %d\n", j, 1*j );

for( j=1 ; j < 10 ; ++j ) → 2단
    printf("2 x %d = %d\n", j, 2*j );

... (생략)

for( j=1 ; j < 10 ; ++j ) → 9단
    printf("9 x %d = %d\n", j, 9*j );
```

```
for( i=1 ; i < 10 ; ++i ) ⇒ i번째 단의

for( j=1 ; j < 10 ; ++j ) ⇒ j번째 곱

printf("%d x %d = %d\n", i, j, i*j );
```





[프로그램 6] 구구단(완성 프로그램)

```
int i, j;
for( i=1 ; i < 10 ; ++i ) { // i 단
  printf("== %d 단 ==\n", i);
  for( j=1 ; j < 10 ; ++j ) // j번째 곱
     printf("%d x %d = %d\n", i, j, i*j);
  printf("----\n");
```

실행 결과

```
== 1 단 ==
1 \times 1 = 1
   ···(생략)
1 \times 9 = 9
== 9 단 ==
9 \times 1 = 9
   ┉(생략)
9 \times 9 = 81
```

프로그램의 실행됨에 따라 제어 흐름과 변수가 어떻게 변하는지 따져보자.
 (교재 p.163의 수행과정 표 참조)





[예제 6.13] 양의 정수 N을 입력 받아, 높이가 N인 삼각형 출력하기



입력

출력

**





[예제 6.14] 양의 정수 N을 입력 받아, 높이가 N인 역삼각형 출력하기

실습 하기



입력

4

출력

**

**



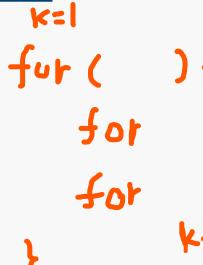


[예제 6.15] 양의 정수 N을 입력 받아, 높이가 N이고 오른쪽으로 치우친 역삼각형 출력하기





√ 별 앞에 적절히 공백 출력



```
입력
```







[예제 6.15a] [응용] 별 대신 수를 출력하도록 위 프로그램을 수정해보자

실습 하기

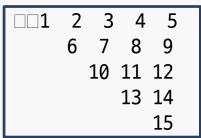


✓ 각 수는 3칸을 차지하도록 출력

입력

5

출력









[예제 6.16] 2~50 사이의 소수를 구하여 각 줄에 5개씩 출력하기

실습 하기



- ✓ 소수: 1과 자신 이외의 수로 나누어 떨어지지 않는 수
- ✓ 소수를 검사하는 간단한 방법
 - ✓ n을 2부터 (n-1)까지 나누어 보기

if((nt2%5==0); printf("\n");



학습 정리

- 중첩 반복(nested loop)은 하나의 반복문 내에 다른 반복문이 포함된 것을 의미
- 반복문의 중첩 횟수에 따라, 이중 반복, 삼중 반복이라고 부르기도 함

