

K G 아 이 티 뱅 크

C 언 어

C L A N G U A G E

반복문 2. for

## 반복문( for )

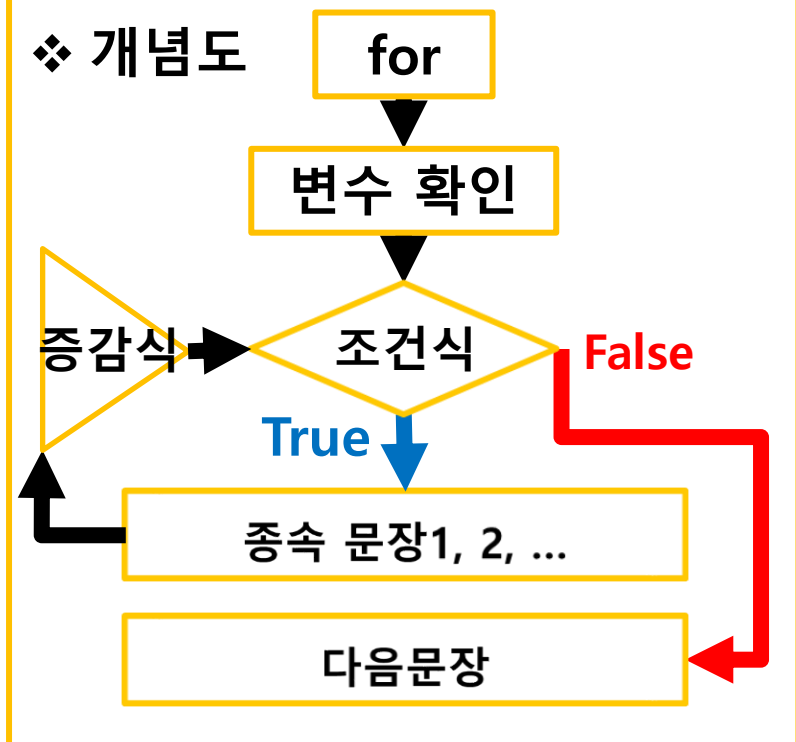
### ❖ 프로그램의 흐름을 제어하는 제어문 중의 세번째

- 규칙성 있게 반복되는 내용의 특정 횟수 실행을 처리
- 변수, 조건식, 증감식을 전부 미리 준비하여 사용

#### ❖ 코드

```
// 최초 구성시 미리 설정  
// 변수를 선언할 수 있음  
for (변수준비;조건식;증감식) {  
    종속문장1;  
    종속문장1;  
    종속문장1;  
}  
다음문장;
```

#### ❖ 개념도



## 반복문( for )

### ❖ for 반복문은 반복횟수가 정해져 있는 반복문

#### ➤ 기본적인 운용은 while과 동일하게 사용이 가능

① while : 변수를 가공하여 결과를 만들 때 주로 사용

② for : 변수는 제어 또는 종속문 보조용으로 사용

#### ➤ for문은 최초 구성단계부터 반복할 횟수를 미리 설정

① for제어문에 한눈에 들어오도록 작성하는 것이 좋음

#### ❖ 코드1. break를 통한 중단

// 반복문은 break로 중단이 가능

```
int num=1;
for (int i=1;i<=10;i++) {
    num*=i;
    if (num>=100) break;
}
printf("num : %d", num);
```

#### ❖ 코드2. continue로 넘기기

# while과 달리 증감식은 별도처리

```
int num=1;
for (int i=1;i<=10;i++) {
    if (i%2==0) continue;
    num*=i;
}
printf("num : %d", num);
```

## 반복문( for )

< 파일이름 : forEX1.c >

❖ 실습예제1. 아래의 조건을 달성하는 코드를 작성하세요.

### 조건

지정한 횟수만큼 아래의 내용을 출력하세요.

“C Language is SIMPLE”

### 결과( 입력 횟수 3로 가정 )

--값의 입력--

횟수 입력 :

--반복 출력--

“C Language is SIMPLE”

“C Language is SIMPLE”

“C Language is SIMPLE”

## 반복문( for )

< 파일이름 : forEX2.c >

❖ 실습예제2. 아래의 조건을 달성하는 코드를 작성하세요.

### 조건

1부터 20까지의 수 중 홀수만 출력합니다.  
아래의 결과를 참고하여 코드를 작성합니다.

### 결과

홀수만 출력하는 반복문

1회차: 1

2회차: 3

3회차: 5

4회차: 7

5회차: 9

...

10회차: 19

## 반복문( for )

< 파일이름 : forEX3.c >

❖ 실습예제3. 아래의 조건을 달성하는 코드를 작성하세요.

### 조건

출력할 단을 입력을 받아 구구단을 출력하세요.

### 결과( 6을 입력했을 경우)

-- 값의 입력 --

출력할 단 입력 >> 6

-- 반복 출력 --

구구단 6단

6 x 1 = 6

.....

6 x 9 = 54

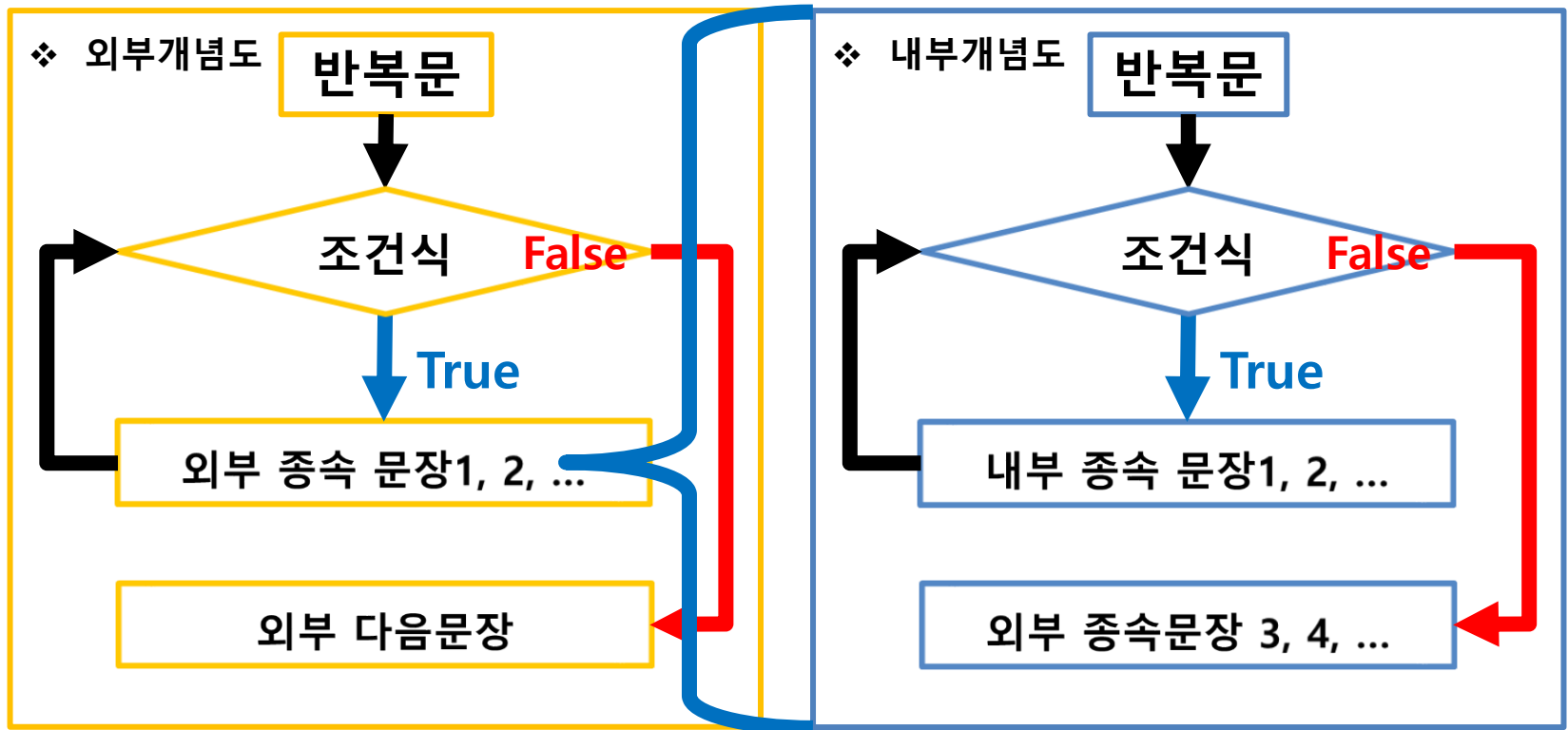
다중 반복문



## 다중 반복문

### ❖ 반복문의 종속문장으로 반복문이 들어간 코드

- while과 for 둘 다 가능하나, 일반적으로 for문으로 구성
- 내부의 반복문의 반복이 다 끝나야 다음 코드를 진행
  - 동시반복, 병렬반복이 아닌 별개의 반복으로 처리됨



## 다중 반복문

❖ 다중 반복문은 종속문의 반복되는 내용을 반복시키는 것

1) 코드의 간략화가 가능하며, 좀 더 가독성이 높아짐

- 비슷한 내용의 반복은 검토할 때 피로를 유발
- 반복되는 비슷한 내용은 사소한 실수를 유발하게 됨

2) while은 같은 변수를 이용하게 되니 주의해야 함

- for : 지역변수로 선언된 변수를 이용하여 처리
- while : 지역변수가 아닐 경우 값이 누적되니 바뀌어야 함

### ❖ 코드1. 단일반복문

```
for (int i=1;i<=3;i++) {  
    printf("%d ",i+1);  
    printf("%d ",i+2);  
    printf("%d ",i+3);  
    printf("%d ",i+4);  
    printf("%d ",i+5);  
    printf("\n");  
}
```

### ❖ 코드2. 이중반복문

```
for (int i=1;i<=3;i++) {  
    for (int j=1;j<=5;j++) {  
        printf("%d ",i+j);  
    }  
    printf("\n");  
}
```

## 다중 반복문

< 파일이름 : dualEX1.c >

❖ 실습예제1. 아래의 조건을 달성하는 코드를 작성하세요.

**조건**

구구단을 출력해보세요.

**결과**

1 x 1 = 1

...

3 x 6 = 18

...

9 x 9 = 81

## 다중 반복문

< 파일이름 : dualEX2.c >

❖ 실습예제2. 아래의 조건을 달성하는 코드를 작성하세요.

### 조건

1부터 25까지의 숫자를 아래와 같이 출력합니다.  
임의로 구성합니다.

### 결과

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

## 다중 반복문

< 파일이름 : dualEX3.c >

❖ 실습예제3. 아래의 조건을 달성하는 코드를 작성하세요.

### 조건

크기와 문자를 입력을 받아 큐브를 만드세요.

### 결과( 크기 3, 문자열 □으로 가정)

- 입력 -  
크기 입력 :  
문자 지정 :  
- 출력 -  
□□□  
□□□  
□□□