2008 Spring

Computer Engineering Programming 1

Lesson 6
- 제7장 반복문

Lecturer: JUNBEOM YOO
jbyoo@konkuk.ac.kr

※ 본 강의자료는 생능출판사의 “PPT 강의자료”를 기반으로 제작되었습니다.

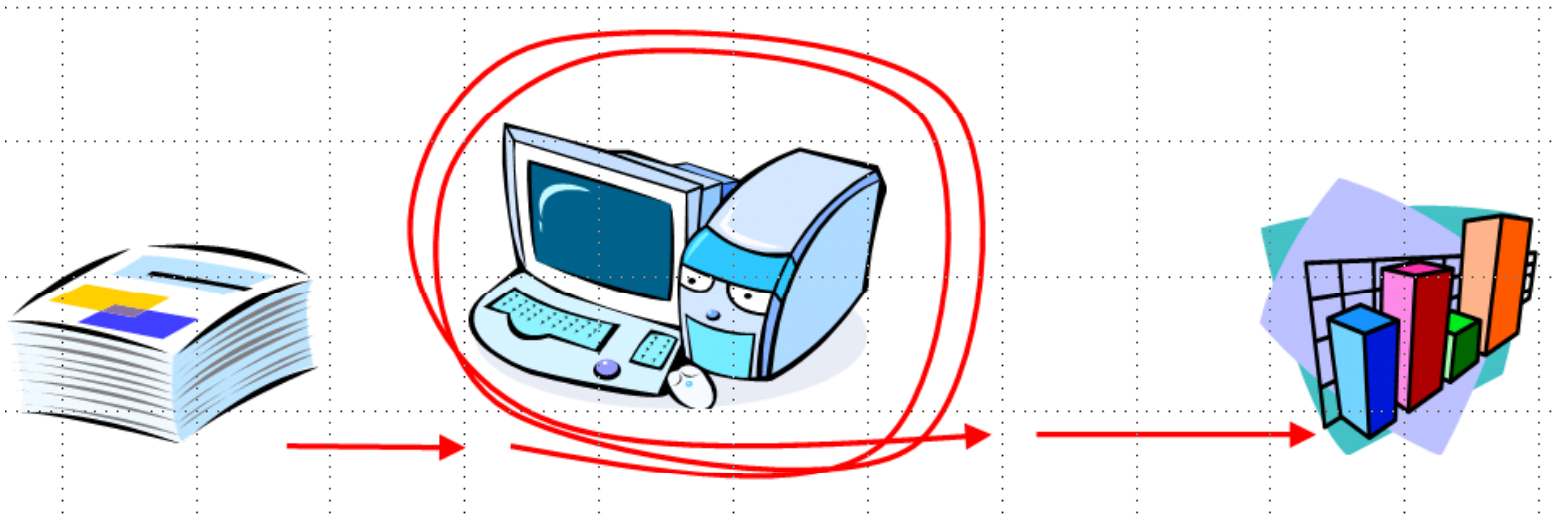
이번 장에서 학습할 내용

- 반복의 개념 이해
- while 반복문
- do-while 반복문
- for 반복문
- break와 continue문

반복문

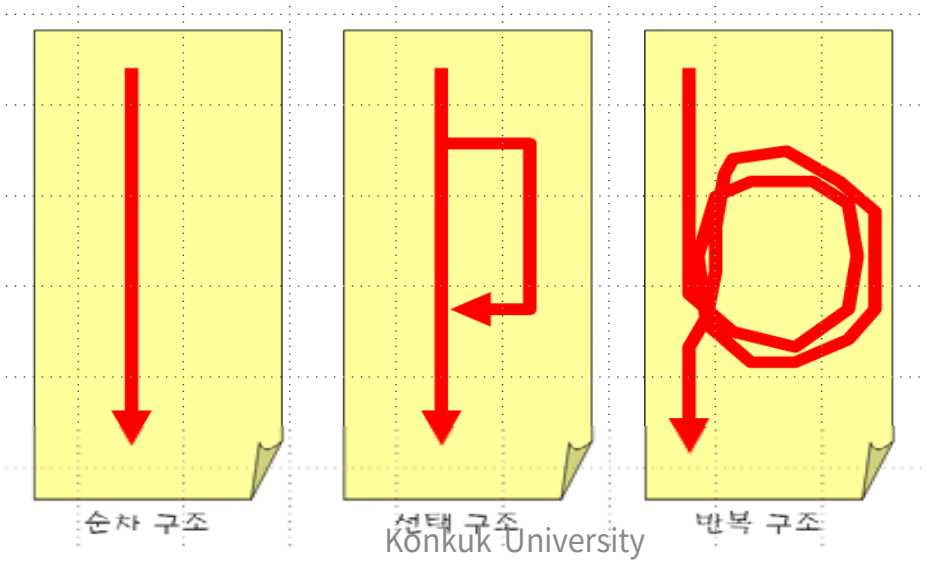
Q) 반복 구조는 왜 필요한가?

A) 같은 처리 과정을 되풀이하는 것이 필요하기 때문이다. 학생 30명의 평균 성적을 구하려면 같은 과정을 30번 반복하여야 한다.



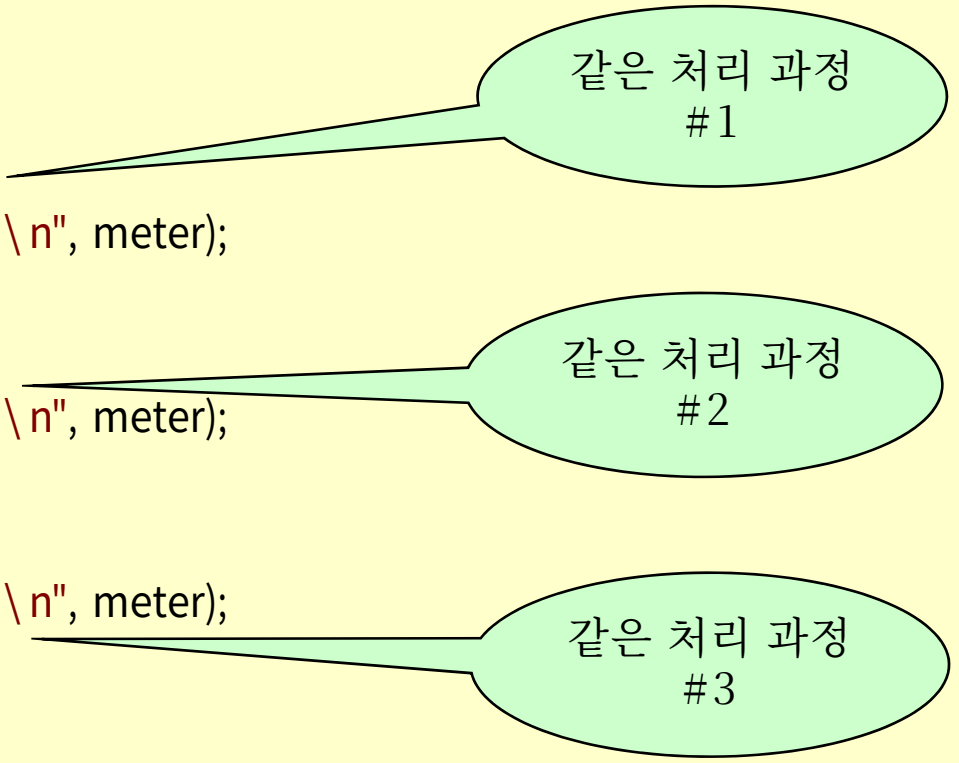
프로그램의 흐름을 제어하는 방법

- 순차 구조
  - 차례대로 실행
- 선택 구조
  - 조건을 검사하여 여러 개의 실행 경로 중에서 하나를 선택
- 반복 구조
  - 조건이 만족될 때까지 반복

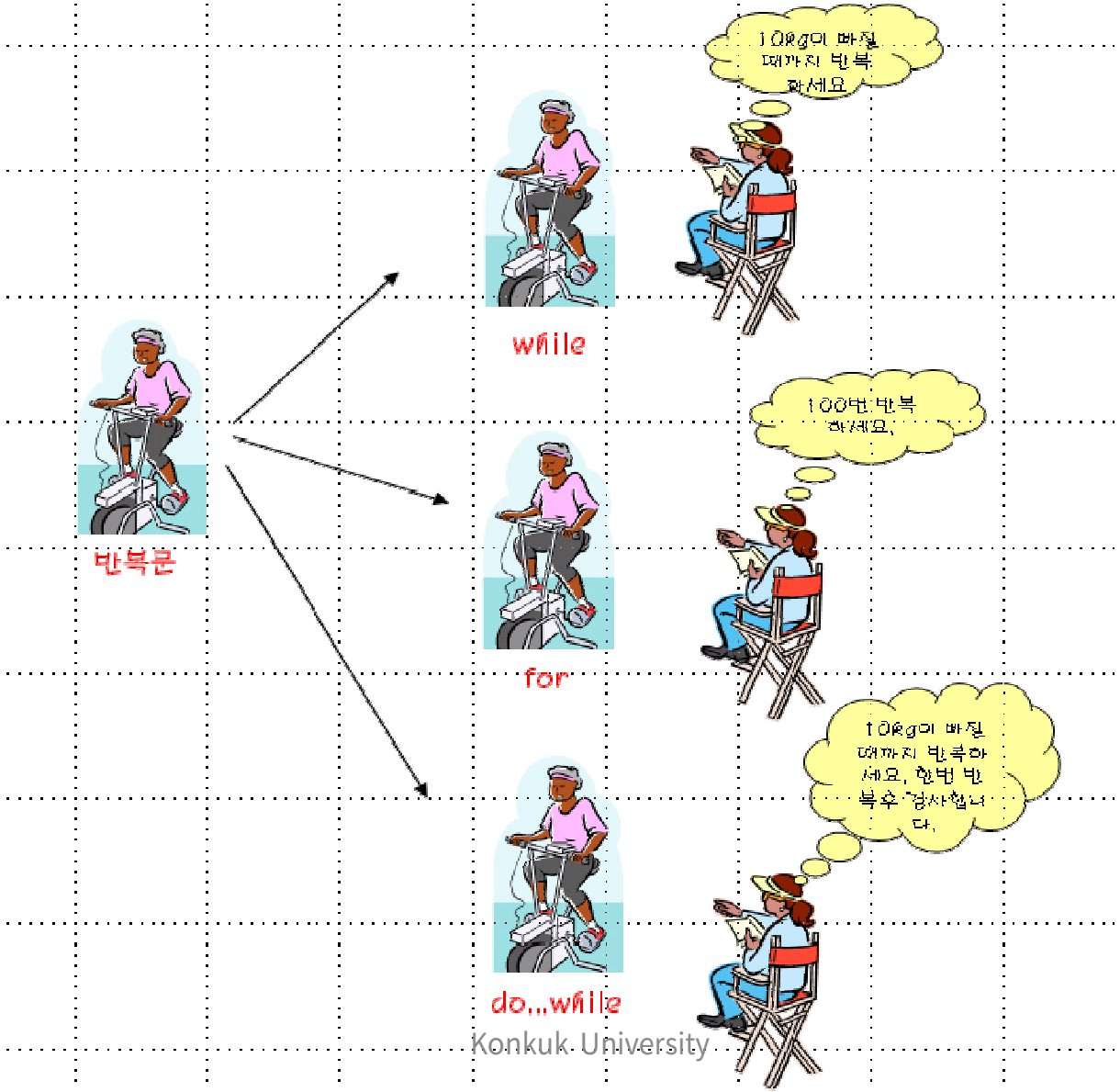


마일을 미터로 바꾸는 프로그램

```
#include <stdio.h>

int main(void)
{
        int meter;

        meter = 0 * 1609;
        printf("0 마일은 %d미터입니다\n", meter);

        meter = 1 * 1609;
        printf("1 마일은 %d미터입니다\n", meter);

        meter = 2 * 1609;
        printf("2 마일은 %d미터입니다\n", meter);
       return 0;
}
```



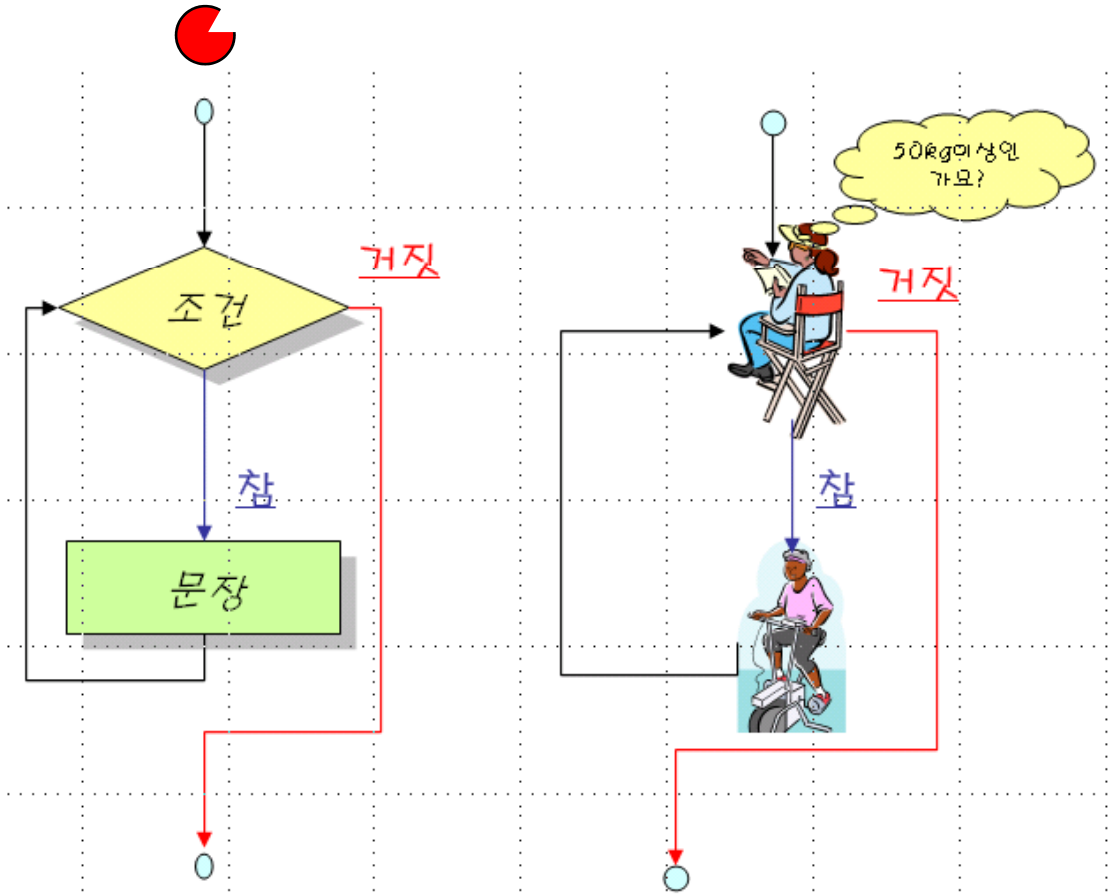
반복문의 종류


반복문
while
for
do...while
100kg이 빠질 때까지 반복하세요
100번 반복하세요.
100kg이 빠질 때까지 반복하세요. 한번 반복후 검사합니다.

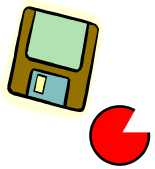
while 문

- 주어진 조건이 만족되는 동안 문장들을 반복 실행한다.

```
while( 조건식 )
    문장;
```



예제



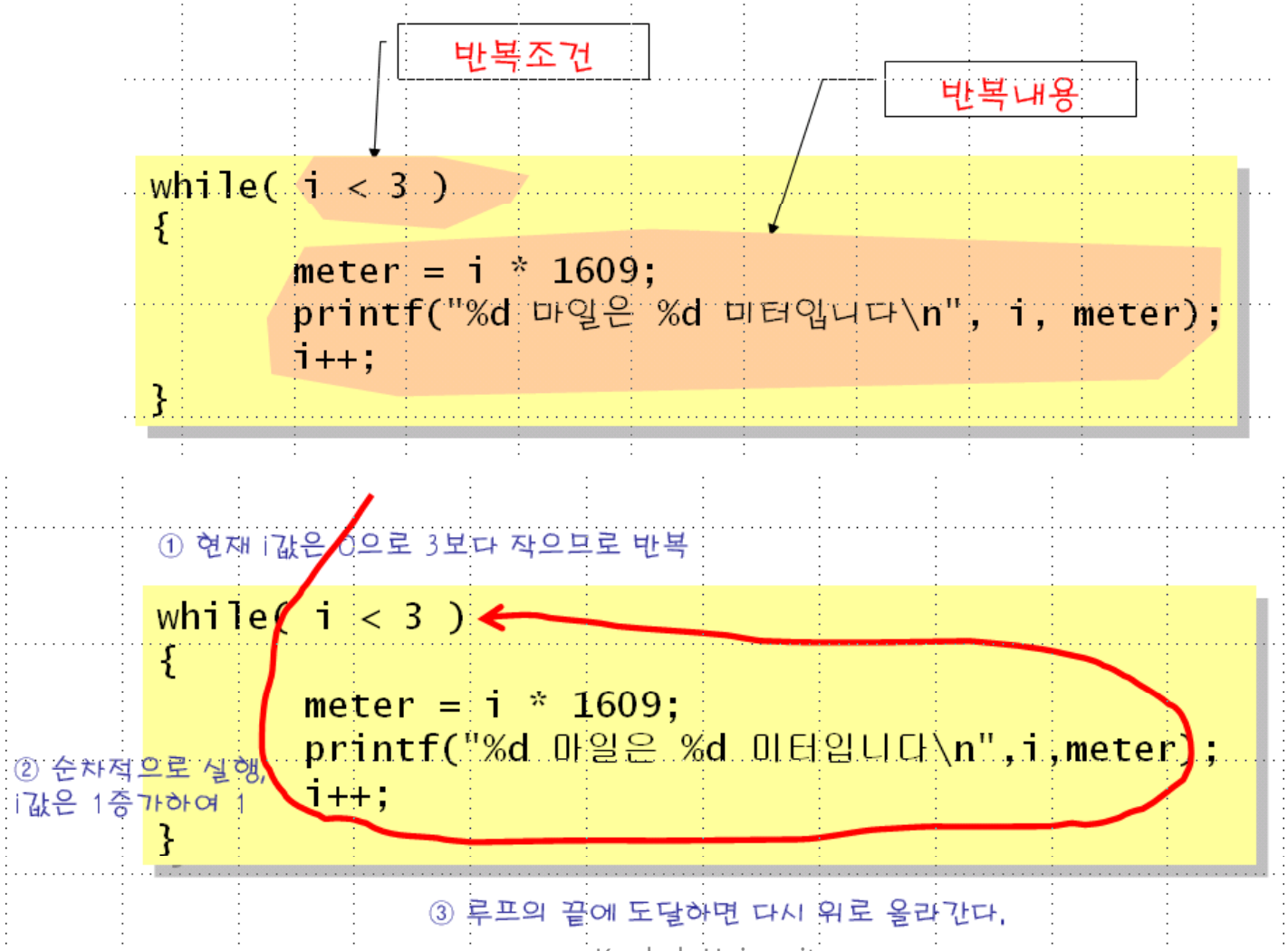
0

i

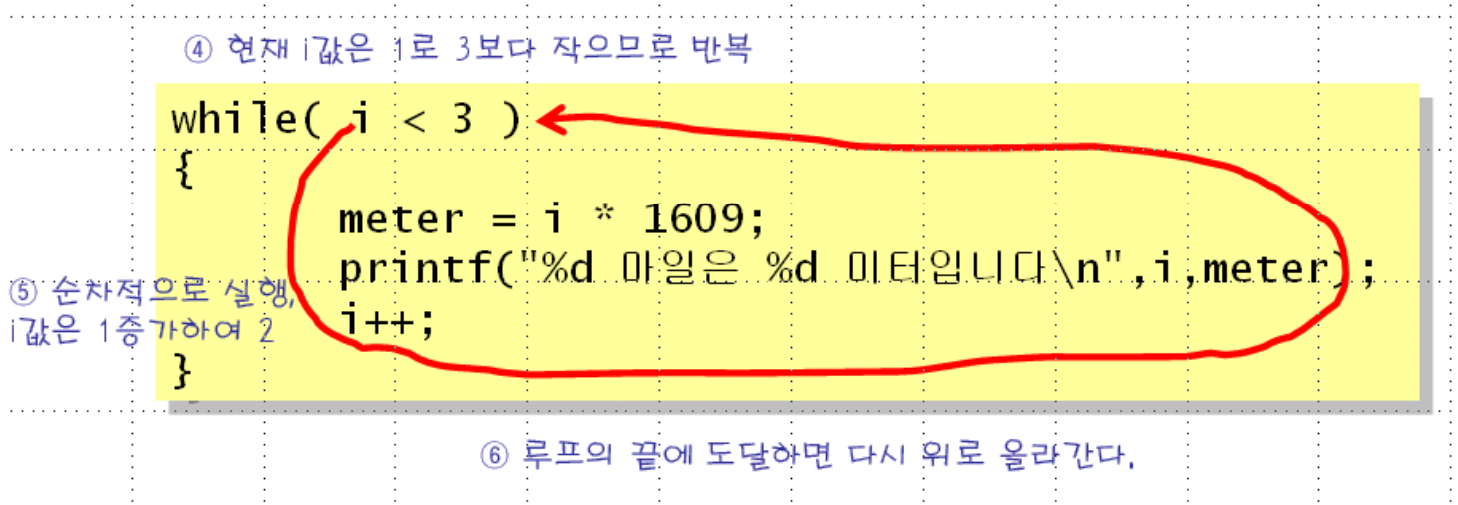
```
#include <stdio.h>

int main(void)
{
        int meter;
        int i = 0;

        while(i < 3)
        {
                meter = i * 1609;
                printf("%d 마일은 %d 미터입니다\n", i, meter);
                i++;
        }
        return 0;
}
```

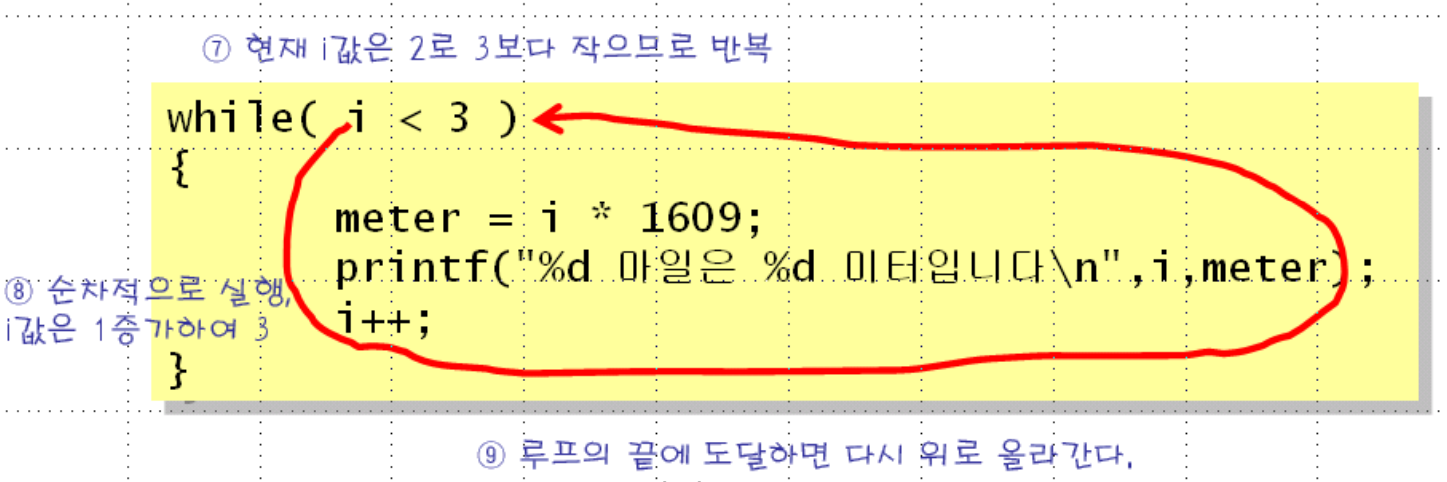
```
0 마일은 0 미터입니다
1 마일은 1609 미터입니다
2 마일은 3218 미터입니다
```

반복문의 실행 과정

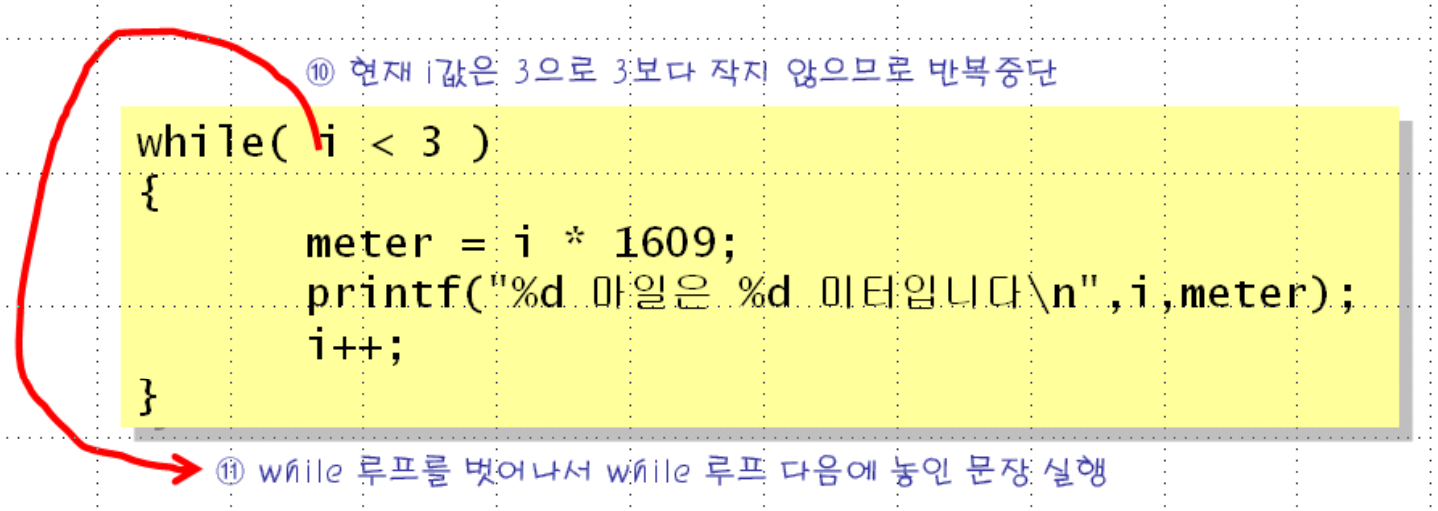

반복조건
반복내용
while(i < 3)
{
meter = i * 1609;
printf("%d 마일은 %d 미터입니다\n", i, meter);
i++;
}
① 현재 i값은 0으로 3보다 작으므로 반복
while(i < 3)
{
meter = i * 1609;
printf("%d 마일은 %d 미터입니다\n",i,meter);
i++;
}
② 순차적으로 실행, i값은 1증가하여 1
③ 루프의 끝에 도달하면 다시 위로 올라간다.

반복문의 실행 과정


④ 현재 i값은 1로 3보다 작으므로 반복
while(i < 3)
{
meter = i * 1609;
printf("%d 마일은 %d 미터입니다\n",i,meter);
i++;
}
⑤ 순차적으로 실행, i값은 1증가하여 2
⑥ 루프의 끝에 도달하면 다시 위로 올라간다.


⑦ 현재 i값은 2로 3보다 작으므로 반복
while(i < 3)
{
meter = i * 1609;
printf("%d 마일은 %d 미터입니다\n",i,meter);
i++;
}
⑧ 순차적으로 실행, i값은 1증가하여 3
⑨ 루프의 끝에 도달하면 다시 위로 올라간다.

반복문의 실행 과정



반복횟수	i의 값	i< 3	반복여부
첫번째 반복(시작위치)	0	참(1)	반복
두번째 반복(시작위치)	1	참(1)	반복
세번째 반복(시작위치)	2	참(1)	반복
네번째 반복(시작위치)	3	거짓(0)	중지

예제

```c
// while 문을 이용한 구구단 출력 프로그램
#include <stdio.h>

int main(void)
{
	int n;
	int i = 1;

	printf("구구단 중에서 출력하고 싶은 단을 입력하시오: ");
	scanf("%d", &n);

	while (i <= 9)
	{
		printf("%d*%d = %d \n", n, i, n*i);
		i++;
	}

	return 0;
}
```

```
구구단 중에서 출력하고 싶은 단을 입력하시오: 9
9*1 = 9
9*2 = 18
9*3 = 27
....
9*9 = 81
```

예제

```c
// while 문을 이용한 제곱값 출력 프로그램
#include <stdio.h>

int main(void)
{
	int n;

	printf("====================\n");
	printf("   n      n의 제곱 \n");
	printf("====================\n");

	n = 1;
	while (n <= 10)
	{
		printf("%5d    %5d\n", n, n*n);
		n++;
	}

	return 0;
}
```

```
====================
   n      n의 제곱
====================
    1      1
    2      4
    3      9
    4     16
    5     25
    6     36
    7     49
    8     64
    9     81
   10    100
```

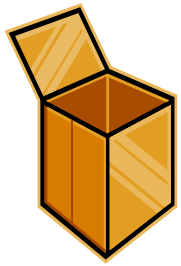
예제

- 1부터 n까지의 합을 계산하는 프로그램

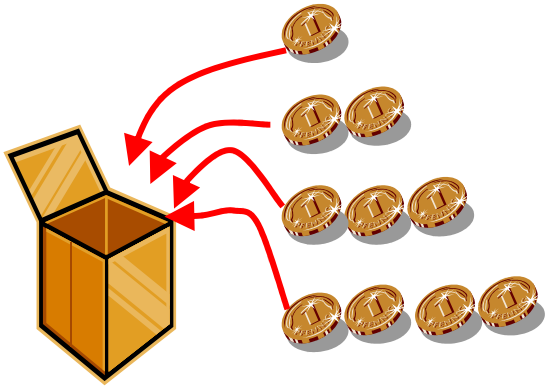
```
1 + 2 + 3 + ... + n
```

- n이 무엇이 될지 모르는 경우라면 다음과 같이 계산

①빈통을 준비한다.

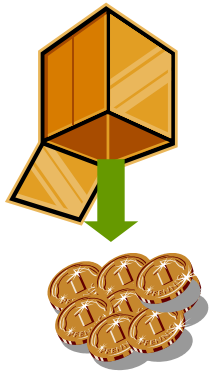


②통에 1부터 n까지를 넣는다.



. . .

③통에 들어 있는 동전의 개수를 출력한다.



예제

```c
#include <stdio.h>

int main(void)
{
	int i, n, sum;				// 변수 선언

	printf("정수를 입력하시오:");		// 입력 안내 메시지 출력
	scanf("%d", &n);			// 정수값 입력

	i = 1;					// 변수 초기화
	sum = 0;

	while(i <= n)
	{
		sum += i;			// sum = sum + i;와 같다.
		i++;				// i = i + 1과 같다.
	}

	printf("1부터 %d까지의 합은 %d입니다\n", n, sum);
	return 0;
}
```

```
정수를 입력하시오: 3
1부터 3까지의 합은 6입니다
```

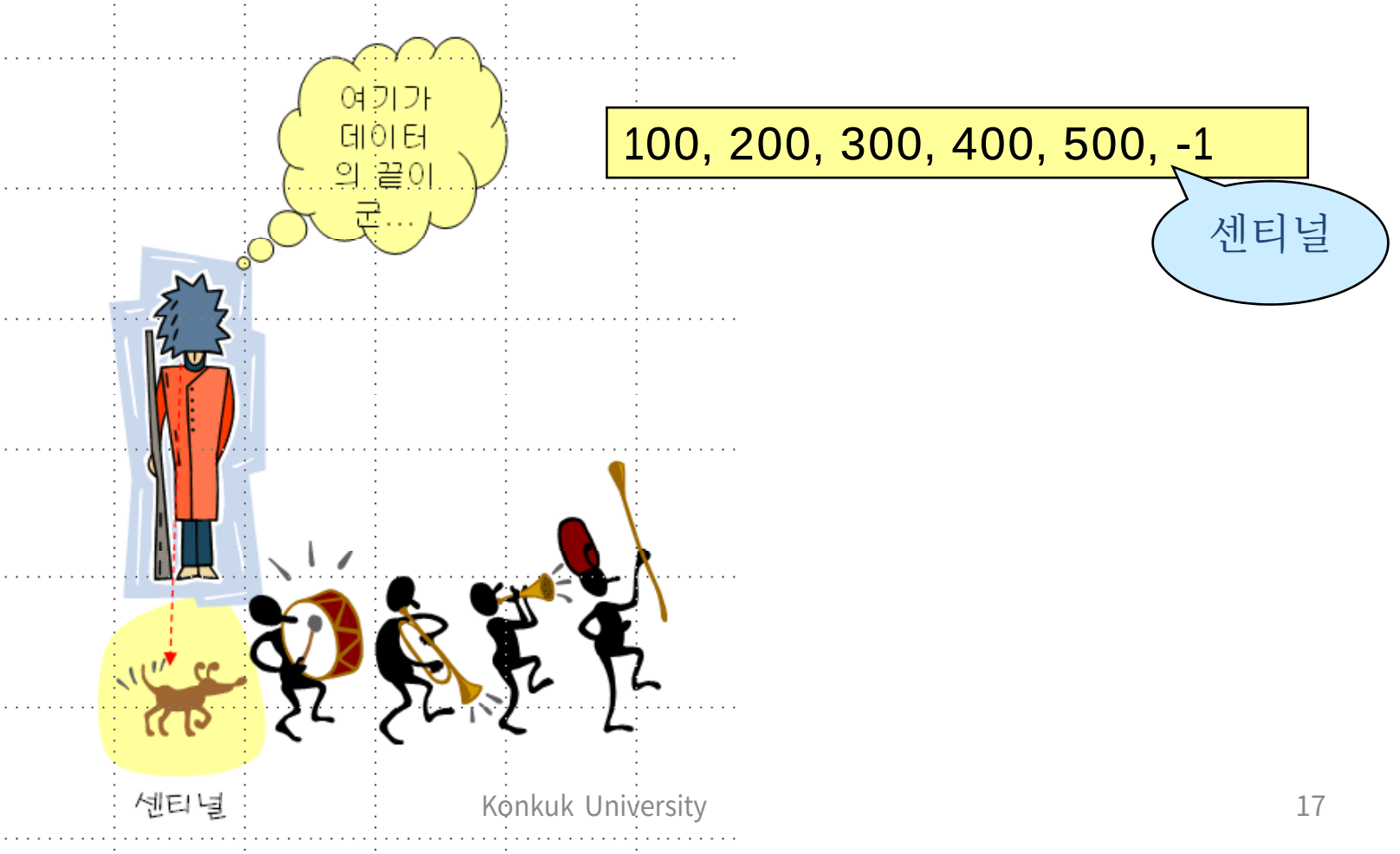
예제

```c
// while 문을 이용한 합계 프로그램
#include <stdio.h>

int main(void)
{
        int i, n, sum;

        i = 0;                          // 변수 초기화
        sum = 0;                        // 변수 초기화
        while (i < 5)
        {
                printf("값을 입력하시오: ");
                scanf("%d", &n);
                sum = sum + n;          // sum += n;과 같다.
                i++;
        }
        printf("합계는 %d입니다.\n", sum);

        return 0;
}
```

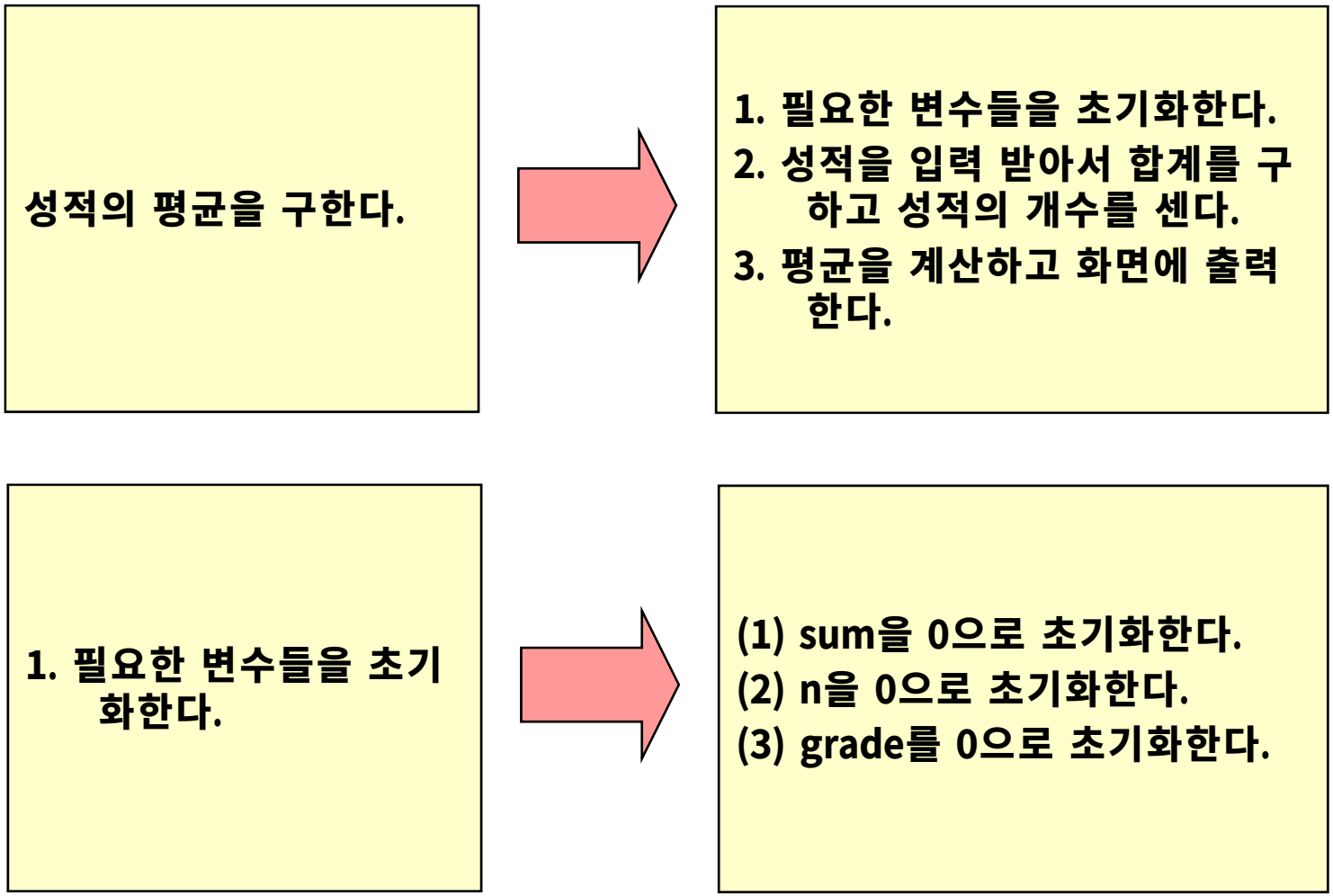
```
값을 입력하시오: 10
값을 입력하시오: 20
값을 입력하시오: 30
값을 입력하시오: 40
값을 입력하시오: 50
합계는 150입니다.
```

센티널(보초값의 이용)

- 센티널: 입력되는 데이터의 끝을 알리는 특수한 값

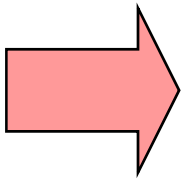


성적들의 평균을 구하는 문제



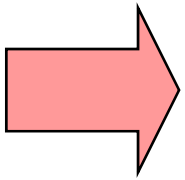
성적들의 평균을 구하는 문제

2. 성적을 입력받아서 합계를 구하고 성적의 개수를 센다.



while 성적이 0보다 작지 않으면
(1) 사용자로부터 성적을 읽어서 grade에 저장한다.
(2) sum에 이 점수를 누적한다.
(3) n을 하나 증가한다.

3. 평균을 계산하고 화면에 출력한다.



(1) sum을 n으로 나누어서 average에 저장한다.
(2) average를 화면에 출력한다.

센티넬 예제 1/2

```
// while 문을 이용한 성적의 평균 구하기 프로그램
#include <stdio.h>

int main(void)
{
        int grade, n;
        float sum, average;

        // 필요한 변수들을 초기화한다.
        n = 0;
        sum = 0;
        grade = 0;

        printf("성적 입력을 종료하려면 음수를 입력하시오\n");
```

센티넬 예제 2/2

```
// 성적을 입력받아서 합계를 구하고 학생 수를 센다.
        while (grade >= 0)
        {
                printf("성적을 입력하시오: ");
                scanf("%d", &grade);


                sum += grade;
                n++;
        }

        sum = sum - grade;   // 마지막 데이터를 제거한다.
        n--;                 // 마지막 데이터를 제거한다.
        // 평균을 계산하고 화면에 출력한다.
        average = sum / n;
        printf("성적의 평균은 %f입니다.\n", average);

        return 0;
}
```

```
성적 입력을 종료하려면 음수를 입력하시오
성적을 입력하시오: 10
성적을 입력하시오: 20
성적을 입력하시오: 30
성적을 입력하시오: 40
성적을 입력하시오: 50
성적을 입력하시오: -1
성적의 평균은 30.000000입니다.
```

예제

- 두수의 최대 공약수 구하기
- 유클리드 알고리즘

① 두 수 가운데 큰 수를 x, 작은 수를 y라 한다.
② y가 0이면 공약수는 x와 같다.
③ r ← x % y
④ x ← y
⑤ y ← r
⑥ 단계 ②로 되돌아간다.

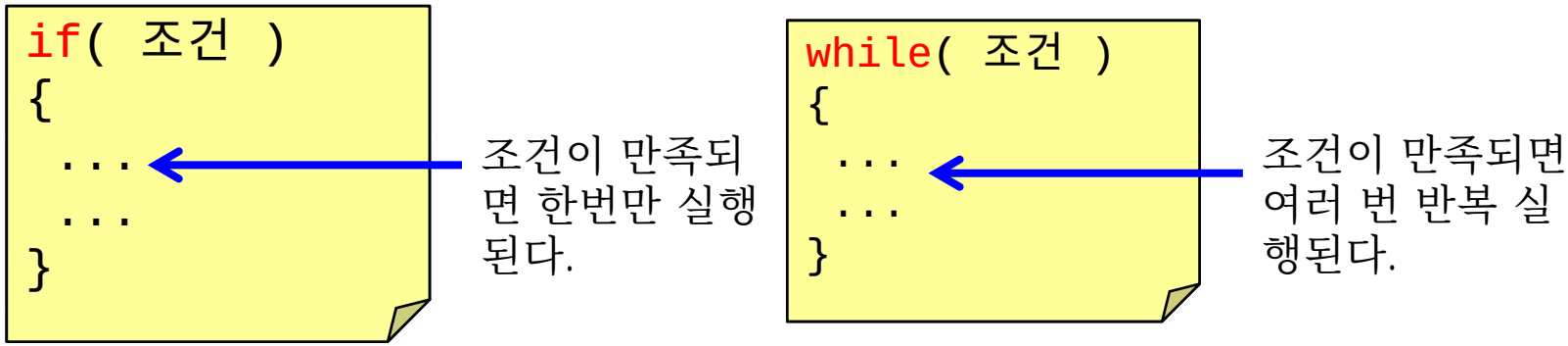
예제

```c
// while 문을 이용한 최대 공약수 구하기 프로그램
#include <stdio.h>

int main(void)
{
        int x, y, r;

        printf("두개의 정수를 입력하시오(큰수, 작은수): ");
        scanf("%d%d", &x, &y);

        while (y != 0)
        {
                r = x % y;
                x = y;
                y = r;
        }
        printf("최대 공약수는 %d입니다.\n", x);

        return 0;
}
```

```
두개의 정수를 입력하시오(큰 수, 작은 수): 12 8
최대 공약수는 4입니다.
```

if 문과 while 문의 비교

- if문은 while 문으로 변경할 수 있다.
- while 문도 if문으로 변경할 수 있다.



while 문에서 주의할 점

```
int i = 1;
while(i < 10)
{
        printf("반복중입니다\n");
        i--;
}
```

변수가 증가 아니라 감소

```
int i = 0;
while(i < 3)
        printf("반복중입니다\n");
        i++;
```

반복 루프에 포함되어 있지 않다.

```
int i = 0;
while(i < 3)  ;
{
        printf("반복중입니다\n");
        i++;
}
```

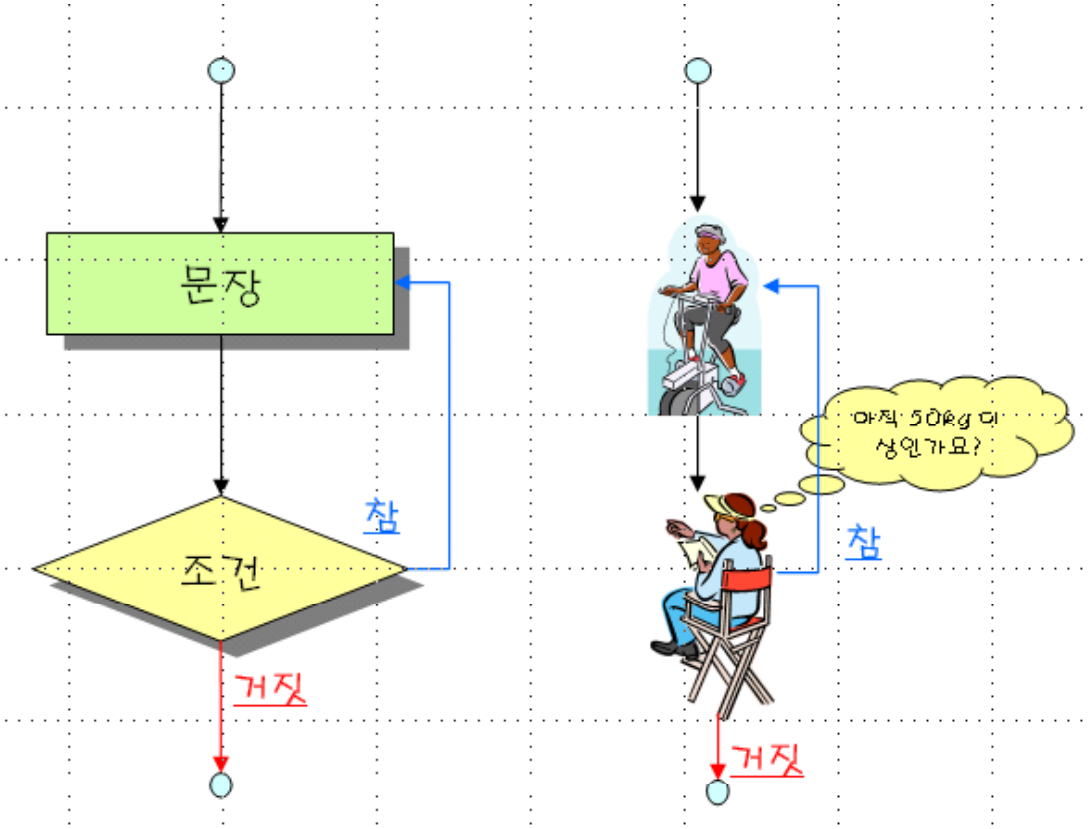
조건뒤에 ;이 있음

do...while문

- 반복 조건을 루프의 끝에서 검사

```
do
    문장
while(조건)
```

① 문장들이 실행된다.
② 조건식이 계산된다.
③ 결과가 참이면 ①로 돌아간다.
④ 결과가 거짓이면 종료된다.



예제

```c
// do..while 문을 이용한 메뉴
#include <stdio.h>

int main(void)
{
	int i = 0;
	do
	{
		printf("1---새로만들기\n");
		printf("2---파일열기\n");
		printf("3---파일닫기\n");
		printf("하나를 선택하시요.\n");
		scanf("%d", &i);
	} while(i < 1 || i > 3);

	printf("선택된 메뉴=%d\n",i);
	return 0;
}
```

```
1---새로만들기
2---파일열기
3---파일닫기
하나를 선택하시요.
1
선택된 메뉴=1
```

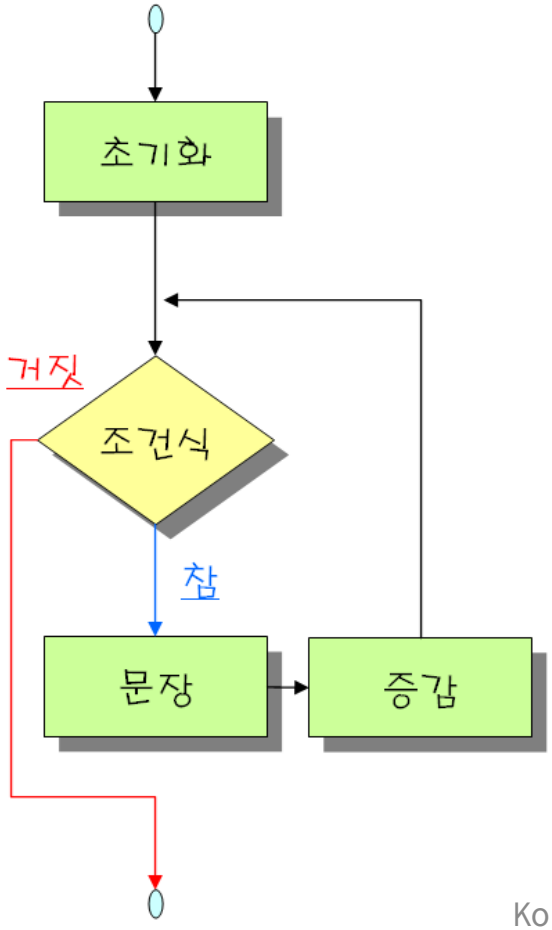
글자 게임 예제

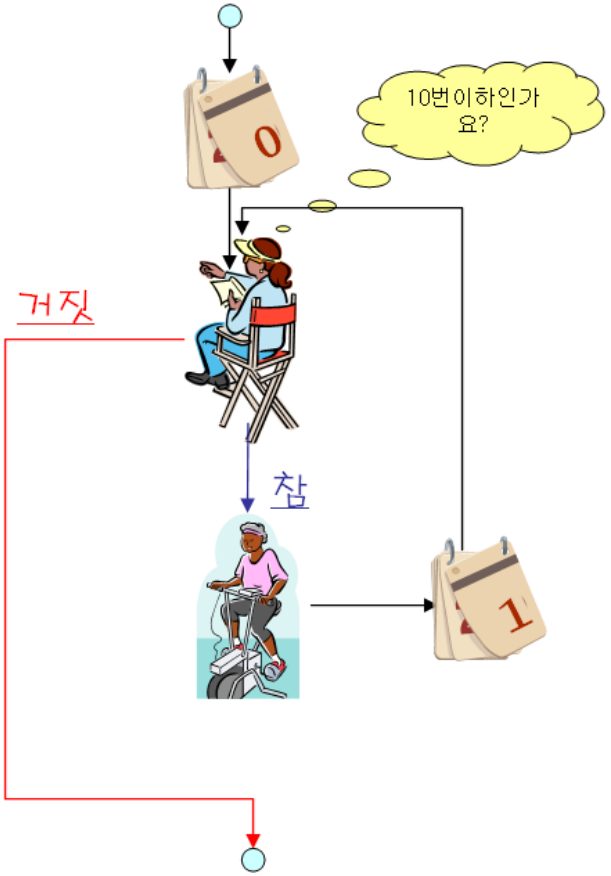
```c
// 반복을 이용한 글자 게임 프로그램
#include <stdio.h>

int main(void)
{
        char answer = 'm'
        char guess;
        int tries = 0;

        do
        {
                printf("정답을 추측하여 보시오: ");
                scanf(" %c", &guess);
                tries++;

                if( guess > answer )
                        printf("제시한 문자의 아스키 코드 값이 높습니다.\n");
                if( guess < answer )
                        printf("제시한 문자의 아스키 코드값이 낮습니다.\n");
        }
        while(guess != answer);

        printf("축하합니다. 시도횟수=%d \n", tries);
        return 0;
}
```

```
정답을 추측하여 보시오: a
제시한 문자의 아스키 코드값이 낮습니다.
정답을 추측하여 보시오: s
제시한 문자의 아스키 코드값이 높습니다.
정답을 추측하여 보시오: b
제시한 문자의 아스키 코드값이 낮습니다.
정답을 추측하여 보시오: z
제시한 문자의 아스키 코드값이 높습니다.
정답을 추측하여 보시오: m
축하합니다. 시도횟수=5
```

for 루프

- 정해진 횟수만큼 반복하는 구조

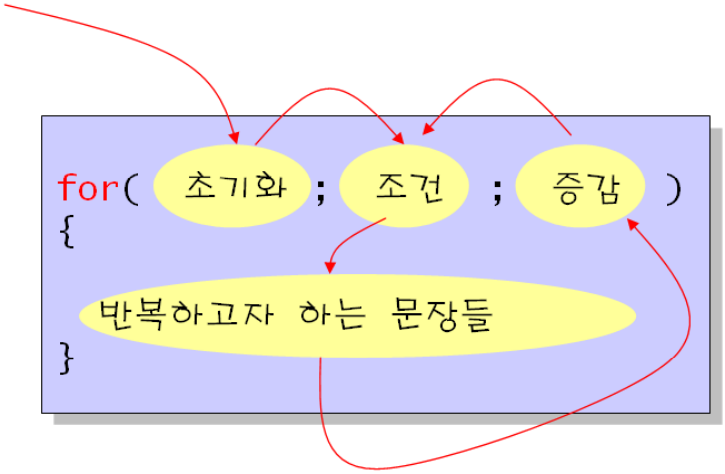




for 문의 구조

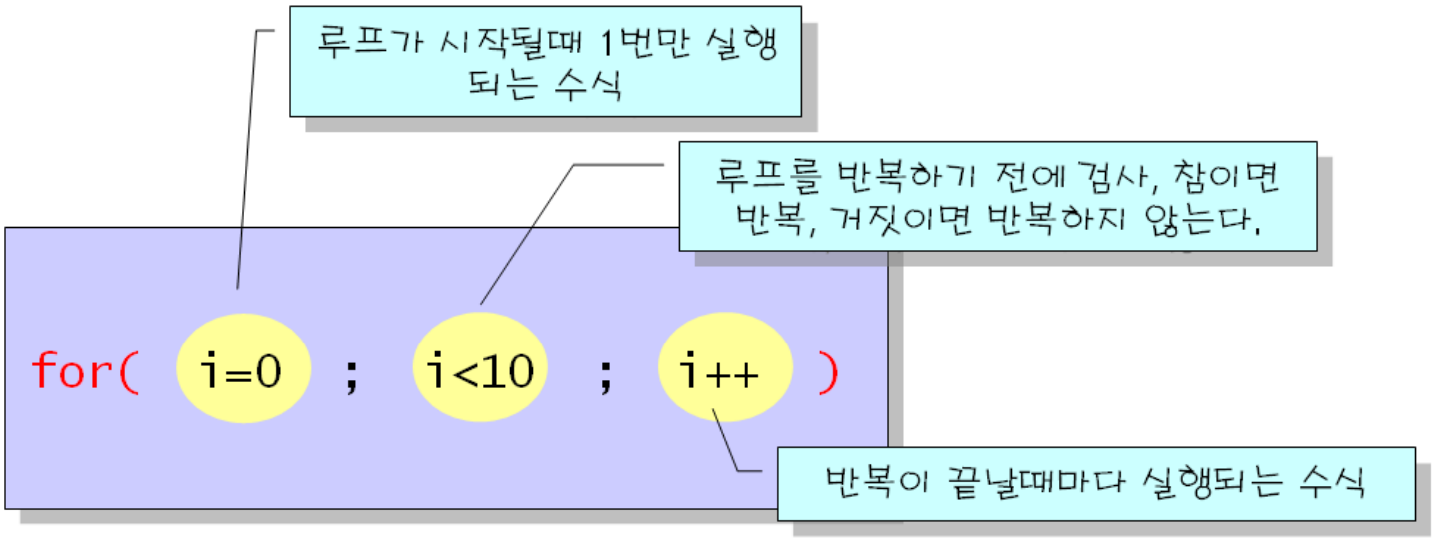
```
for ( 초기화; 조건식; 증감식)
    문장;
```

① 초기화를 실행한다.
② 반복 조건을 나타내는 조건식을 계산한다.
③ 수식의 값이 거짓이면 for 문의 실행이 종료된다.
④ 수식의 값이 참이면 문장이 실행된다.
⑤ 증감을 실행하고 ②로 돌아간다.

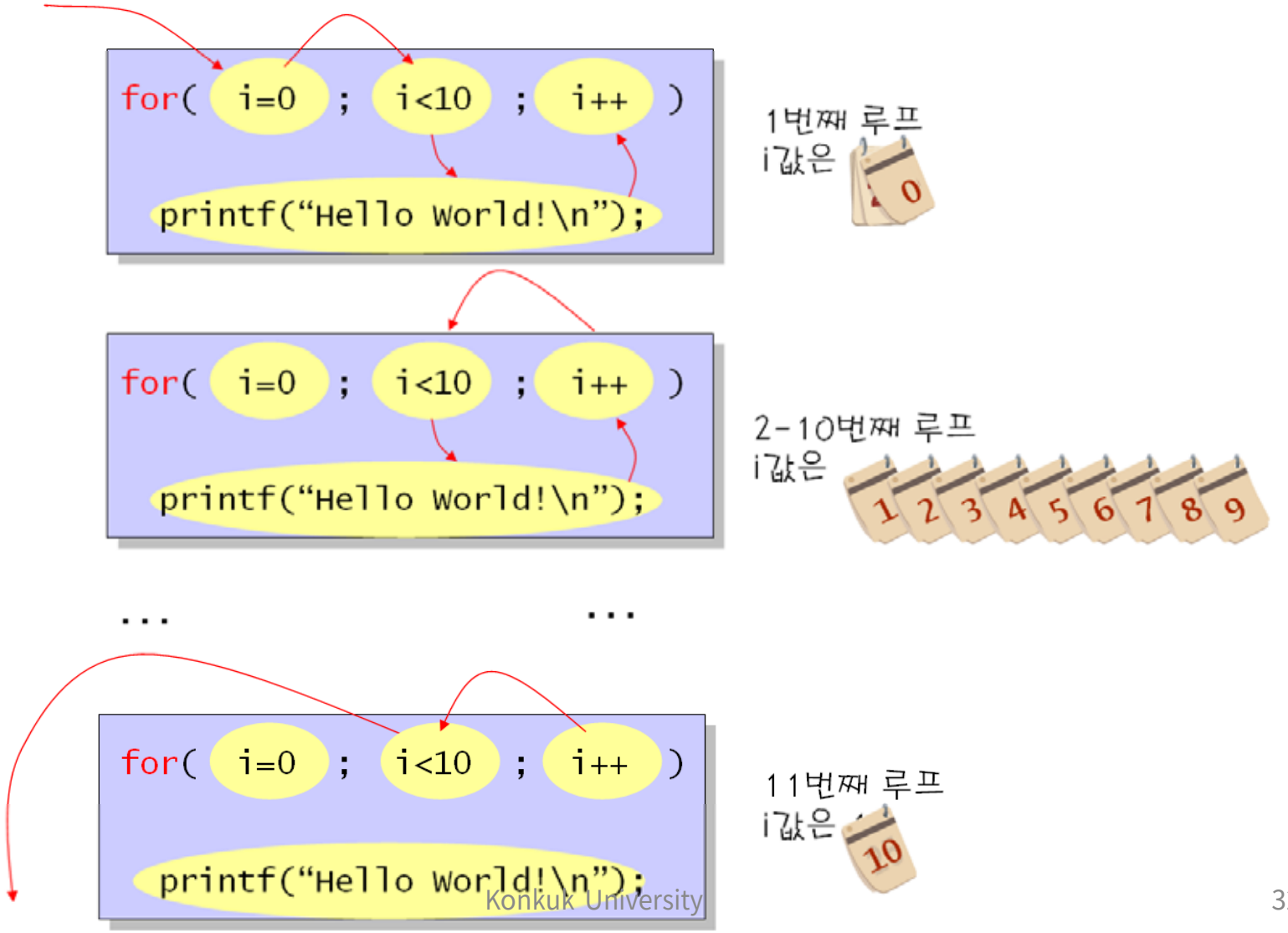


for 문의 구조

```
int i;
for(i = 0; i < 10; i++)
        printf("Hello World!\n");
```



for문의 실행과정


for(i=0 ; i<10 ; i++)
printf("Hello World!\n");
1번째 루프
i값은
0
for(i=0 ; i<10 ; i++)
printf("Hello World!\n");
2-10번째 루프
i값은
1 2 3 4 5 6 7 8 9
...
...
for(i=0 ; i<10 ; i++)
printf("Hello World!\n");
11번째 루프
i값은
10

for문의 실행과정

```
int i;
for(i = 0; i < 10; i++)
        printf("Hello World!\n");
```

반복횟수	i의 값	i<10	반복여부
1번째 반복	0	참	반복
2번째 반복	1	참	반복
3번째 반복	2	참	반복
4번째 반복	3	참	반복
5번째 반복	4	참	반복
6번째 반복	5	참	반복
7번째 반복	6	참	반복
8번째 반복	7	참	반복
9번째 반복	8	참	반복
10번째 반복	9	참	반복
11번째 반복	10	거짓	중지

예제

```c
// 반복을 이용한 정수합 프로그램
#include <stdio.h>

int main(void)
{
        int i, sum;

        sum = 0;
        for(i = 1;i <= 10; i++)
                sum += i;                       // sum = sum + i;와 같음

        printf("1부터 10까지의 정수의 합= %d\n",sum);

        return 0;
}
```

```
1부터 10까지의 정수의 합 = 55
```

예제

```c
// 반복을 이용한 세제곱값구하기
#include <stdio.h>

int main(void)
{
	int i, n;

	printf("정수를 입력하시요:");
	scanf("%d", &n);

	printf("====================\n");
	printf("  i      i의 세제곱\n");
	printf("====================\n");
	for(i = 1;i <= n; i++)
		printf("%5d    %5d\n", i, i*i*i);

	return 0;
}
```

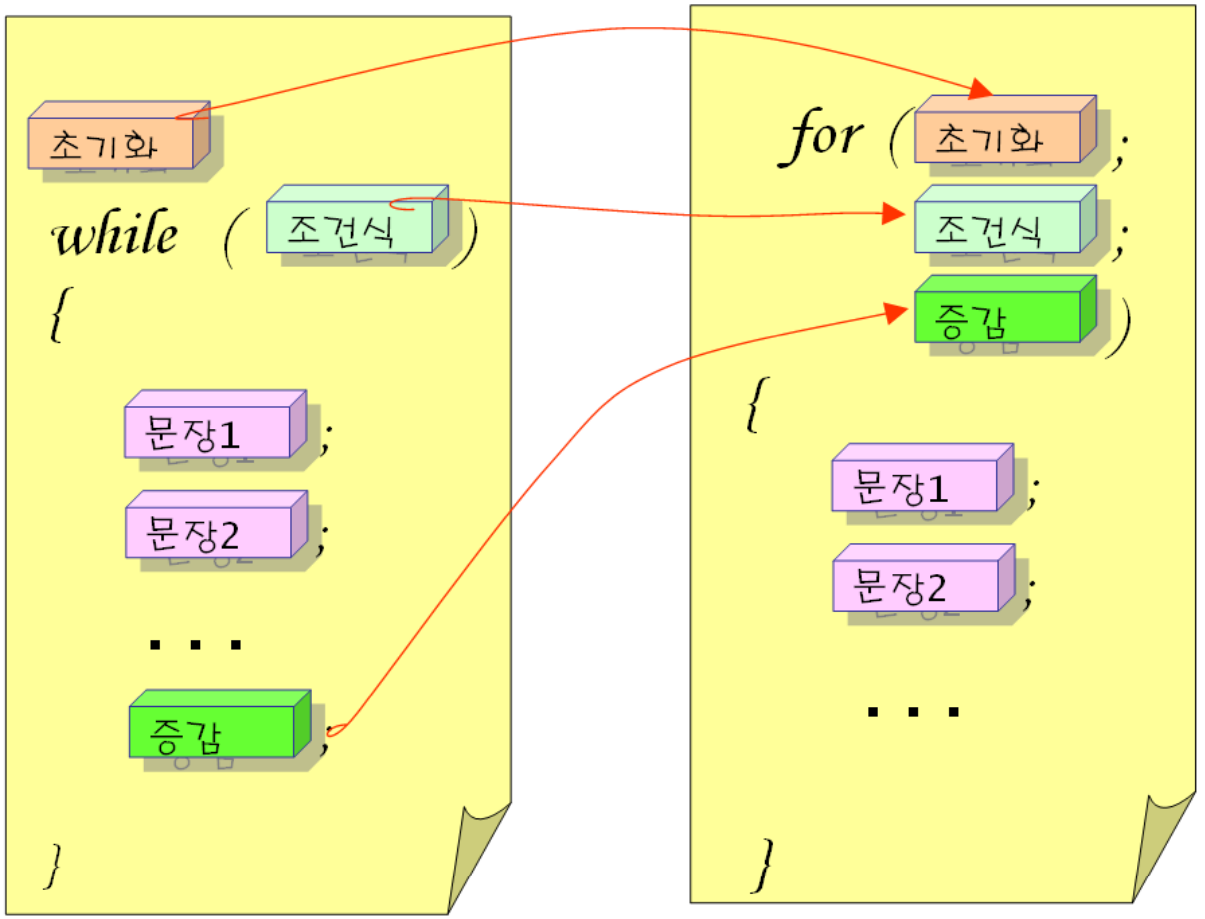
```
정수를 입력하시요:5
====================
  i      i의 세제곱
====================
    1        1
    2        8
    3       27
    4       64
    5      125
```

예제

```c
// 반복을 이용한 팩토리얼 구하기
#include <stdio.h>

int main(void)
{
	long fact=1;
	int i, n;

	printf("정수를 입력하시요:");
	scanf("%d", &n);

	for(i = 1;i <= n; i++)
		fact = fact * i;

	printf("%d!은 %d입니다.\n",n,fact);

	return 0;
}
```

```
정수를 입력하시요: 10
10!은 3628800입니다.
```

while 루프와 for 루프와의 관계



다양한 증감수식의 형태

```
for (i = 10; i > 0; i-- )
    printf("Hello World!\n");
```

빨셈 사용

```
for (i = 0; i < 10; i += 2 )
    printf("Hello World!\n");
```

2씩 증가

```
for (i = 1; i < 10; i *= 2 )
    printf("Hello World!\n");
```

2를 곱한다.

```
for (i = 0; i < 100; i = (i * i) + 2 )
    printf("Hello World!\n");
```

어떤 수식이라도 가능

```
for  ( ; i<100; i++ )
    printf("Hello World!\n");
```

한부분이 없을 수도 있다.

```
for (i = 0, k = 0; i < 100; i++ )
    printf("Hello World!\n");
```

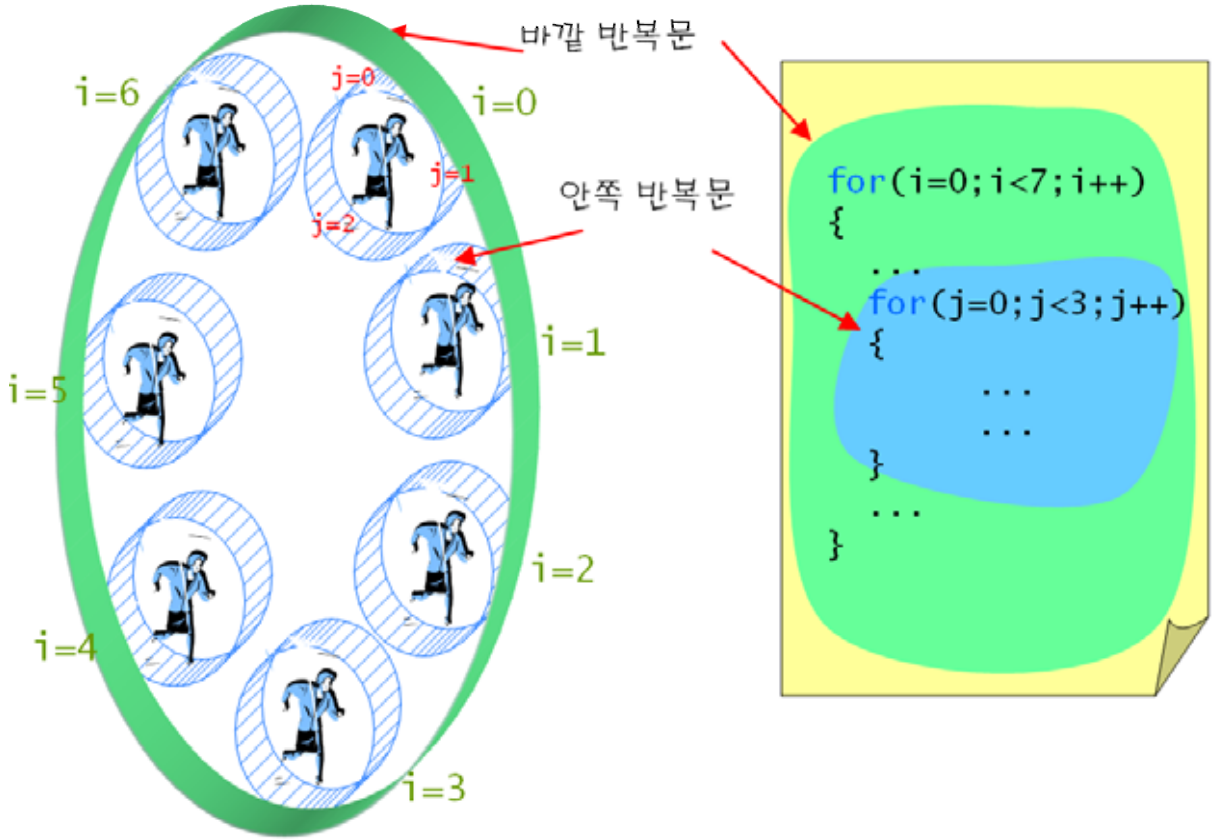
2개 이상의 변수 초기화

```
for (printf("반복시작"), i = 0; i < 100; i++ )
    printf("Hello World!\n");
```

어떤 수식도 가능

중첩 반복문

- 중첩 반복문(nested loop): 반복문 안에 다른 반복문이 위치



예제

```c
// 중첩 for 문을 이용하여 *기호를 사각형 모양으로 출력하는 프로그램
#include <stdio.h>

int main(void)
{
	int x, y;

	for(y = 0;y < 5; y++)
	{
		for(x = 0;x < 10; x++)
			printf("*");
		printf("\n");
	}

	return 0;
}
```

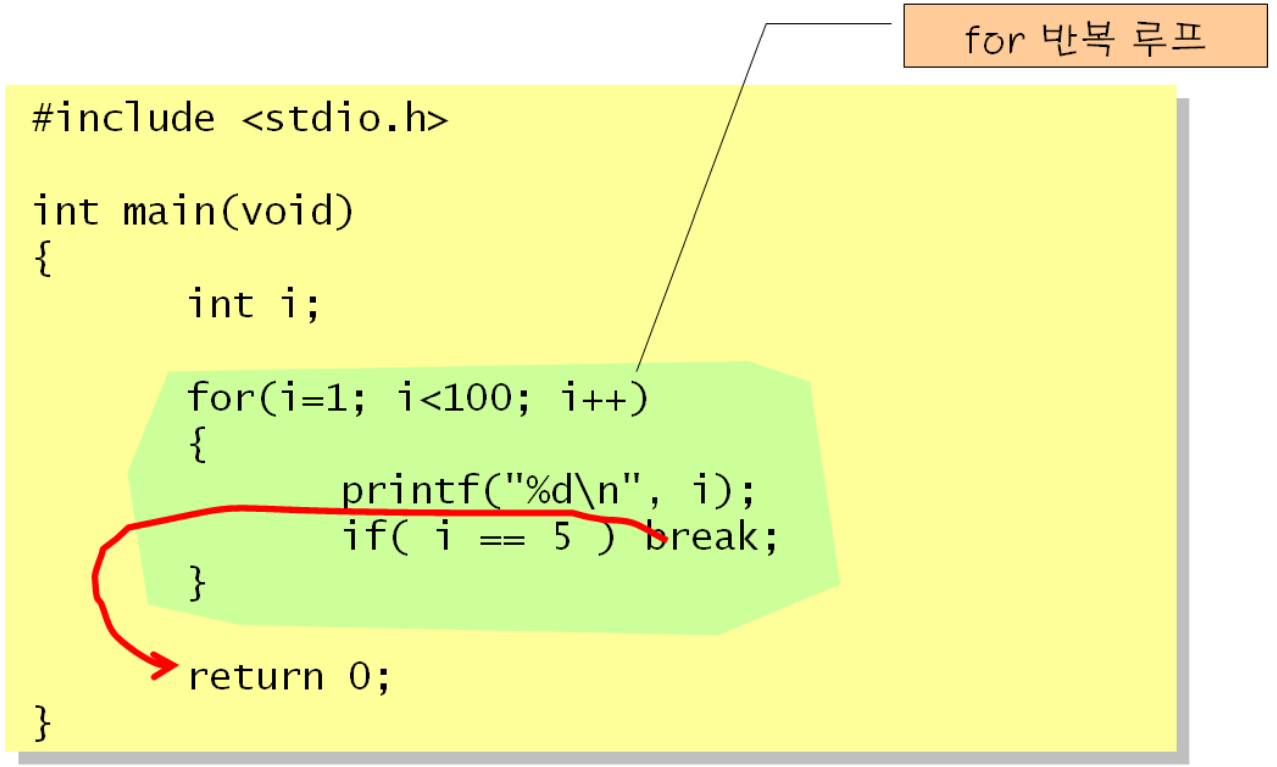
```
**********
**********
**********
**********
**********
```

예제

```c
#include <stdio.h>

int main(void)
{
	int s, t;
	float score, total, average;

	for(s = 0;s < 20; s++)
	{
		total = 0;
		for(t = 0;t < 5; t++)
		{
			printf("%d번 학생에 대한 %번째 성적: ");
			scanf("%f", &score);
			total += score;
		}
		average = total / 5.0;
		printf("%d번 학생의 성적 평균: %f\n", average);
	}
	return 0;
}
```

break 문

- break 문은 반복 루프를 빠져 나오는데 사용된다.



예제

```c
// break를 이용하여 무한루프를 탈출한다.
#include <stdio.h>
#include <math.h>

int main(void)
{
        double v;

        while(1)
        {
                printf("실수값을 입력하시오: ");
                scanf("%lf", &v);
                if( v < 0.0 )
                        break;
                printf("%f의 제곱근은 %f입니다.\n", v, sqrt(v));
        }

        return 0;
}
```

```
실수값을 입력하시오: 9.0
9.000000의 제곱근은 3.000000입니다.
실수값을 입력하시오: 12.0
12.000000의 제곱근은 3.464102입니다.
실수값을 입력하시오: 25.0
25.000000의 제곱근은 5.000000입니다.
실수값을 입력하시오: -1
```

예제

```c
// break를 이용하여 무한루프를 탈출한다.
#include <stdio.h>

int main(void)
{
	float grade, sum = 0.0, average;
	int count = 0;

	while(1)
	{
		printf("학생 성적을 입력하시오: ");
		scanf("%f", &grade);

		if( grade < 0.0 )
			break;
		count++;
		sum += grade;
	}
	average = sum / count;
	printf("학생들의 성적의 평균은 %f입니다.\n", average);
	return 0;
}
```

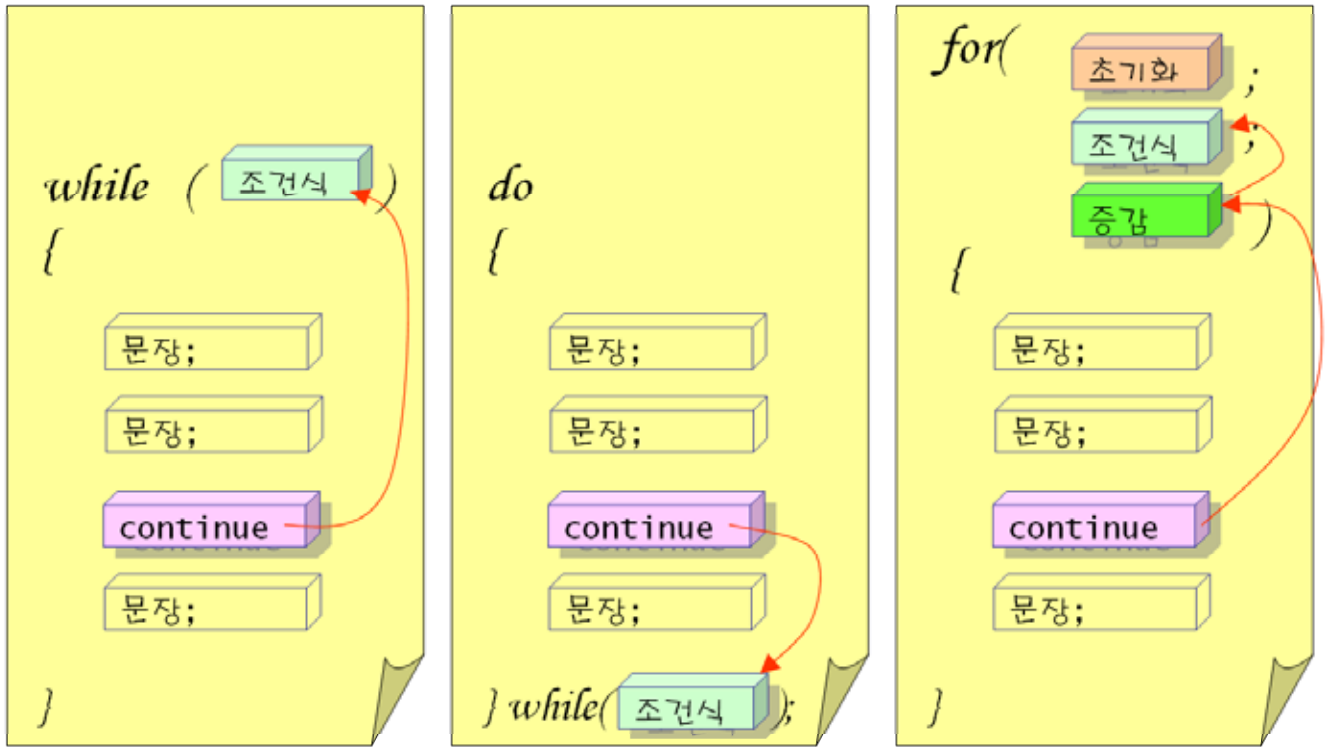
```
학생 성적을 입력하시오: 90
학생 성적을 입력하시오: 90
학생 성적을 입력하시오: 80
학생 성적을 입력하시오: 70
학생 성적을 입력하시오: -1
학생들의 성적의 평균은 82.500000입니다.
```

goto문의 사용

```
#include <stdio.h>

int main(void)
{
        int x, y;

        for(y = 1; y < 10000; y++)
        {
                for(x = 1; x < 50; x++)
                {
                        if( _kbhit() ) goto OUT;
                        printf("*");
                }
                printf("\n");
        }
OUT:
        return 0;
}
```

continue 문

- 현재의 반복을 중단하고 다음 반복을 시작하게 한다.



예제

```
#include <stdio.h>

int main(void)
{
        int i = 0;
        int sum = 0;

        for(i = 0;i < 100; i++)
        {
                if(i % 2 == 1)
                        continue;
                sum += i;
        }
        printf("sum = %d\n", sum);

        return 0;
}
```

```
sum = 245
```

예제

```c
// 소문자를 대문자로 변경한다.
#include <stdio.h>

int main(void)
{
	char letter;

	while(1)
	{
		printf("소문자를 입력하시오: ");
		scanf(" %c", &letter);

		if( letter == 'Q' )
			break ;
		if( letter < 'a' || letter > 'z' )
			continue ;

		letter -= 32;
		printf("변환된 대문자는 %c입니다.\n", letter);
	}

	return 0;
}
```

```
소문자를 입력하시오: a
변환된 대문자는 A입니다.
소문자를 입력하시오: b
변환된 대문자는 B입니다.
소문자를 입력하시오: c
변환된 대문자는 C입니다.
소문자를 입력하시오: Q
```

예제

```c
// 복리이자계산
#include <stdio.h>

#define RATE 0.07                  // 이율
#define INVESTMENT  10000000       // 초기 투자금
#define YEARS       10             // 투자 기간

int main(void)
{
      int i;
      double total = INVESTMENT; // 원리금 합계

      printf("==============\n");
      printf("연도  원리금\n");
      printf("==============\n");

      for(i = 1; i <= YEARS; i++)
      {
            total = total * ( 1 + RATE );      // 새로운 원리금 계산
            printf("%2d  %10.1f\n", i, total);
      }

      return 0;
}
```

```
==============
연도  원리금
==============
 1     10700000.0
 2     11449000.0
 3     12250430.0
 4     13107960.1
 5     14025517.3
 6     15007303.5
 7     16057814.8
 8     17181861.8
 9     18384592.1
10     19671513.6
```

예제

```
#include <stdio.h>
#define START_DAY         3     // 첫번째 날이 수요일
#define DAYS_OF_MONTH     31    // 달의 일수

int main(void)
{
        int day, date;
        printf("====================\n");
        printf("일 월 화 수 목 금 토\n");
        printf("====================\n");

        for(day = 0; day < START_DAY ; day++)    // 월요일부터 수요일까지
                printf("   ");                   // 공백 출력
        for(date = 1; date <= DAYS_OF_MONTH ; date++)
        {
                if( day == 7 )
                {
                        day = 0;      // 일요일이면 줄바꿈을 출력
                        printf("\n");
                }
                day++;
                printf("%2d ", date);            // 날을 출력한다.
        }
        printf("\n====================\n");
        return 0;
}
```

```
====================
일 월 화 수 목 금 토
====================
          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
26 27 28 29 30 31
====================
```

Q & A