



C 언어 스터디 3주차



- 변수와 상수의 차이점은??
- 기호상수를 사용하는 법 2가지??

복습

```
1 #include<stdio.h>
2
3 int main(void)
4 {
5     int a, b, c;
6     int deter;
7
8     printf("이차방정식의 계수를 순서대로 입력하시오 : ");
9     scanf("%d %d %d", &a, &b, &c);
10
11    deter = b * b - 4 * a * c;
12
13    deter == 0 ? printf("중근입니다.\n") : (deter > 0 ? printf("근이 2개입니다.\n") : printf("허근입니다.\n"));
14
15    return 0;
16 }
```

복습

```
1 #include<stdio.h>
2
3 int main(void)
4 {
5     int x = 10, y = 20;
6
7     printf("x = %2d, y = %2d\n", ++x, y++);
8     printf("x = %2d, y = %2d\n", x--, ++y);
9     printf("x = %2d, y = %2d\n", --x, y);
10    printf("x = %2d, y = %2d\n", x++, y--);
11
12    printf("x = %2d, y = %2d\n", x, y);
13
14    return 0;
15 }
16
```

복습

```
1 #include<stdio.h>
2
3 int main(void)
4 {
5     int number;
6
7     printf("다섯 자리 수를 입력하시오 : ");
8     scanf("%d", &number);
9
10    printf("일의 자리 수 : %d\n", number % 10);
11    number /= 10;
12
13    printf("십의 자리 수 : %d\n", number % 10);
14    number /= 10;
15
16    printf("백의 자리 수 : %d\n", number % 10);
17    number /= 10;
18
19    printf("천의 자리 수 : %d\n", number % 10);
20    number /= 10;
21
22    printf("만의 자리 수 : %d\n", number);
23
24    return 0;
25 }
```

제어문

제어문 : 문장들이 실행되는 순서에 영향을 주는 문장을 의미함.

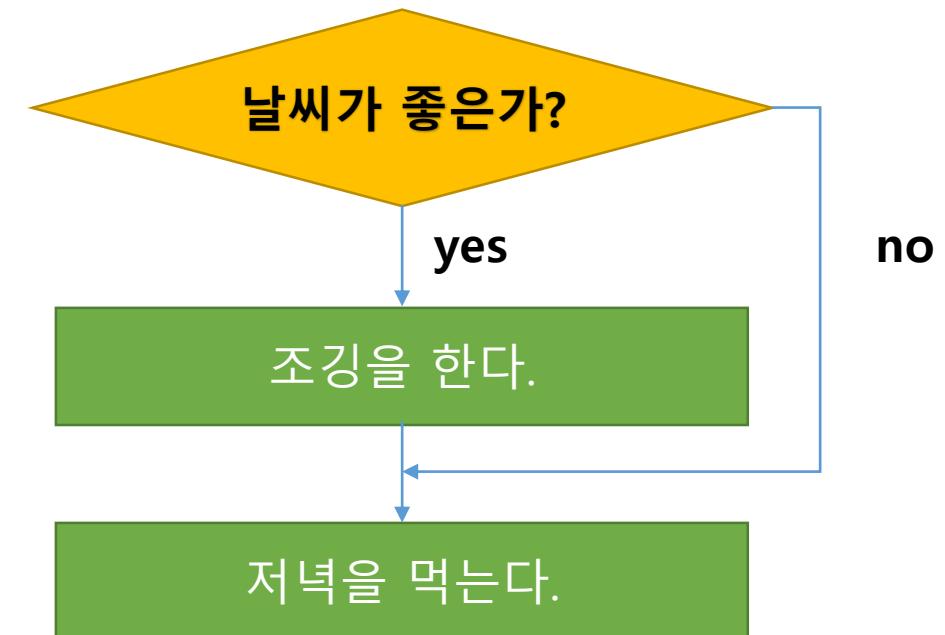
- 1) 조건문 : 어떤 조건에 따라서 여러 개의 실행 경로 가운데 하나를 선택할 때 사용한다. **if**와 **switch** 문장이 여기에 속한다.
- 2) 반복문 : 어떤 조건이 유지되는 한, 정해진 횟수만큼 처리를 되풀이하는 문장을 의미한다. **for**문과 **while**문이 여기에 속한다.

if 문

형식) if (조건식)
문장;

의미) 만약 조건식이 참이면 문장이 실행된다.

Ex) if (number > 0)
printf("양수입니다");

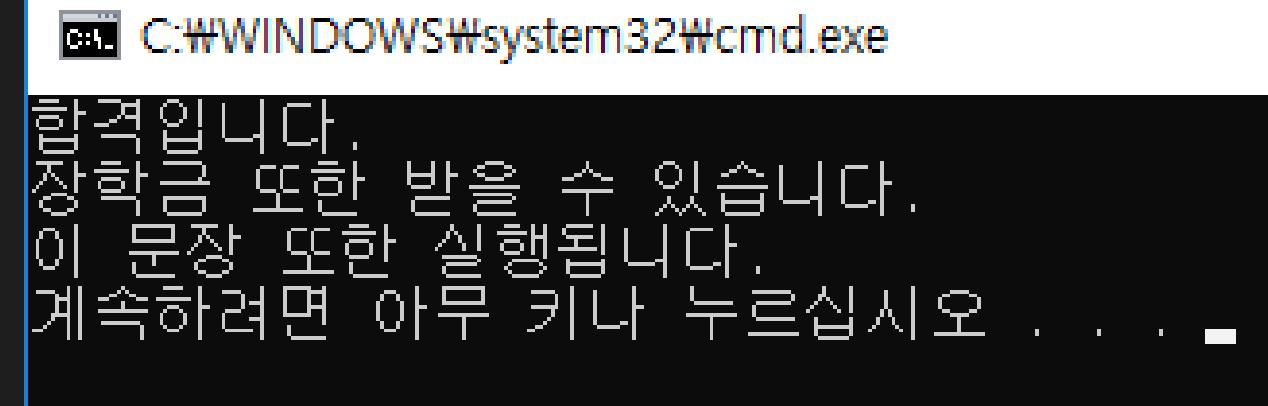


If문이 끝나면 그 다음 문장이 실행된다.

복합문 (블록)

조건이 참일 경우에 여러 문장이 실행되어야 하는 경우 중괄호로 묶어서 한꺼번에 실행시킬 수 있다. 이를 **복합문**(compound statement)라고 한다.

```
1 #include<stdio.h>
2
3 int main() {
4     int score = 70;
5
6     if (score >= 60)
7     {
8         printf("합격입니다.\n");
9         printf("장학금도 받을 수 있습니다\n");
10    }
11    printf("이 문장 또한 실행됩니다.");
12
13    return 0;
14 }
```



if 문 예제 (절대값 구하기)

```
dy
1 #include<stdio.h>
2
3 int main() {
4     int number;
5
6     printf("정수를 입력하시오 : ");
7     scanf("%d", &number);
8
9     if (number < 0)
10         number = -number;
11     //number가 0보다 작으면 부호를 바꿈
12
13     printf("절대값은 %d입니다.\n", number);
14
15 }
```

```
C:\WINDOWS\system32\cmd.exe
정수를 입력하시오 : -40
절대값은 40입니다.
계속하려면 아무 키나 누르십시오 . . .
C:\WINDOWS\system32\cmd.exe
정수를 입력하시오 : 24
절대값은 24입니다.
계속하려면 아무 키나 누르십시오 . . .
```

if - else 문

형식) if (조건식)

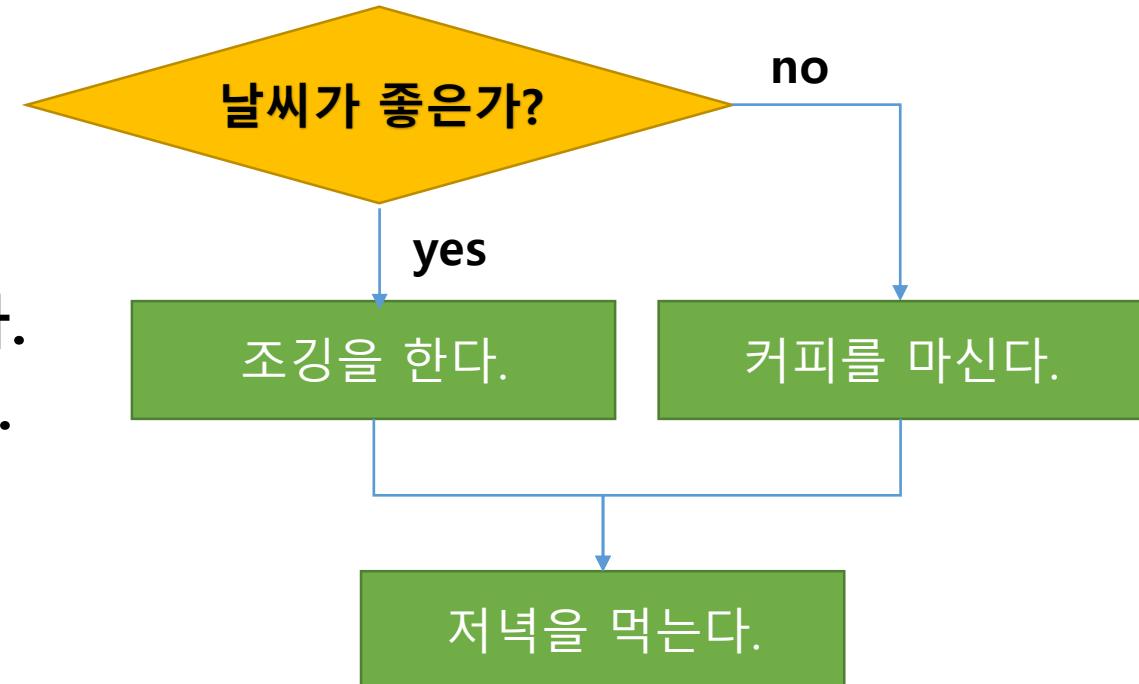
문장1;

else

문장2;

의미) 만약 조건식이 참이면 문장1이 실행된다.
조건식이 거짓이라면 문장2가 실행된다.

Ex) if (number > 0)
 printf("양수입니다 ");
else
 printf("음수입니다 ");



조건문과 조건 연산자

if-else문은 지난 주에 배운 조건 연산자(?)를 이용하여 표현할 수 있다.

```
if (score >= 60)  
    printf("합격입니다.");  
else  
    printf("불합격입니다.");
```

==

```
(score >= 60) ? printf("합격입니다.") : printf("불합격입니다.");
```

동일!!

if - else 문 예제

```
1 #include<stdio.h>
2
3 int main() {
4     int number;
5
6     printf("정수를 입력하시오 : ");
7     scanf("%d", &number);
8
9     if (number % 2 == 0)
10         printf("입력된 정수는 짝수입니다\n");
11     else
12         printf("입력된 정수는 홀수입니다.\n");
13
14     return 0;
15 }
```

```
C:\WINDOWS\system32\cmd.exe
정수를 입력하시오 : 312
입력된 정수는 짝수입니다
계속하려면 아무 키나 누르십시오 . . .
```

```
C:\WINDOWS\system32\cmd.exe
정수를 입력하시오 : 2341
입력된 정수는 홀수입니다.
계속하려면 아무 키나 누르십시오 . . .
```

복잡한 조건식

연산	의미
$x == y$	x와 y가 같은가?
$x != y$	x와 y가 다른가?
$x > y$	x가 y보다 큰가?
$x < y$	x가 y보다 작은가?
$x >= y$	x가 y보다 크거나 같은가?
$x <= y$	x가 y보다 작거나 같은가?

연산	의미
$x \&\& y$	AND연산. x와 y가 모두 참이여야 참
$x y$	OR연산. x나 y 중 하나만 참이면 참
$!x$	NOT연산. x가 참이면 거짓, 거짓이면 참

- 성적이 80점 이상이고 90점 미만인 경우 : `if (score >= 80 && score < 90)`
- 문자가 'a'이거나 'b'인 경우 : `if (ch == 'a' || ch == 'b')`
- 윤년인지 아닌지 판단 : `if ((year % 4 == 0 && year % 100 != 0) || year % 400 == 0)`
(윤년 : 1. 연도가 4로 나누어 떨어질 것. 단, 100으로 나누어 떨어지는 해 제외.
2. 400으로 나누어 떨어지는 해는 윤년.)

다중 if 문

1. 중첩 if문 : if문 안에 if문이 포함

Ex) 국어 점수가 80점 이상인 학생들 중에 영어 점수가 90점 이상이면 우수학생 스티커를 주고, 영어가 90점 이하면 준우수학생 스티커를 주려고 한다.

```
if (Korean >= 80)
    if (English >= 90)
        printf("우수학생입니다");
    else
        printf("준우수학생입니다 " );
```

중첩 if문에서 else절 매칭

Ex) 국어 점수가 80점 이상인 학생들 중에 영어 점수가 90점 이상이면 우수학생 스티커를 주고, 영어가 90점 이하면 준우수학생 스티커를 주려고 한다.

```
if (Korean >= 80)
    if (English >= 90)
        printf("우수학생입니다");
    else
        printf("준우수학생입니다 " );
```



```
if (Korean >= 80)
    if (English >= 90)
        printf("우수학생입니다");
    else
        printf("준우수학생입니다 " );
```

If와 else가 어떻게 매칭이 될까?
: else절은 **가장 가까운 if절**과 매칭이 된다.
밑의 else절과 매칭시키고 싶다면 **중괄호**를 이용한다.

```
if (Korean >= 80)
{
    if (English >= 90)
        printf("우수학생입니다");
}
else
    printf("해당되지 않습니다. " );
```

연속적인 if 문

2. **연속적인 if문** : if 다음에 else if 를 이어서 쓰는 것.

Ex) 시험 점수가 90점 이상이면 A, 90점 미만 80점 이상이면 B, 80점 미만 70점 이상이면 C 그 외에는 F학점을 준다.

```
if (score >= 90)
    printf("A학점입니다.");
else if (score >= 80)
    printf("B학점입니다.");
else if (score >= 70)
    printf("C학점입니다.");
else
    printf("F학점입니다.");
```

연속적인 if 문

Ex) 이차방정식의 근 판별

```
if (deter == 0)
    printf("중근입니다.");
else if (deter > 0)
    printf("근이 2개입니다.");
else
    printf("허근입니다.");
```

`(deter==0) ? printf("중근입니다.") : ((deter > 0) ? printf("근이 2개입니다.") : printf("허근입니다."));`

Switch문

: If문과 달리 실행경로가 여러 개인 조건문.

형식) switch(제어식)

```
{  
    case c1:  
        문장1;  
        break;  
  
    case c2:  
        문장2;  
        break;  
  
    ...  
    default:  
        문장d;  
        break;  
}
```

의미) 제어식을 계산하여 c_1, c_2, \dots 중 제어식과 일치하는 값의 문장을 차례로 실행하고 **break**를 만나면 종료된다. 만약 일치하는 값을 찾지 못한다면 default 값의 문장이 실행된다.

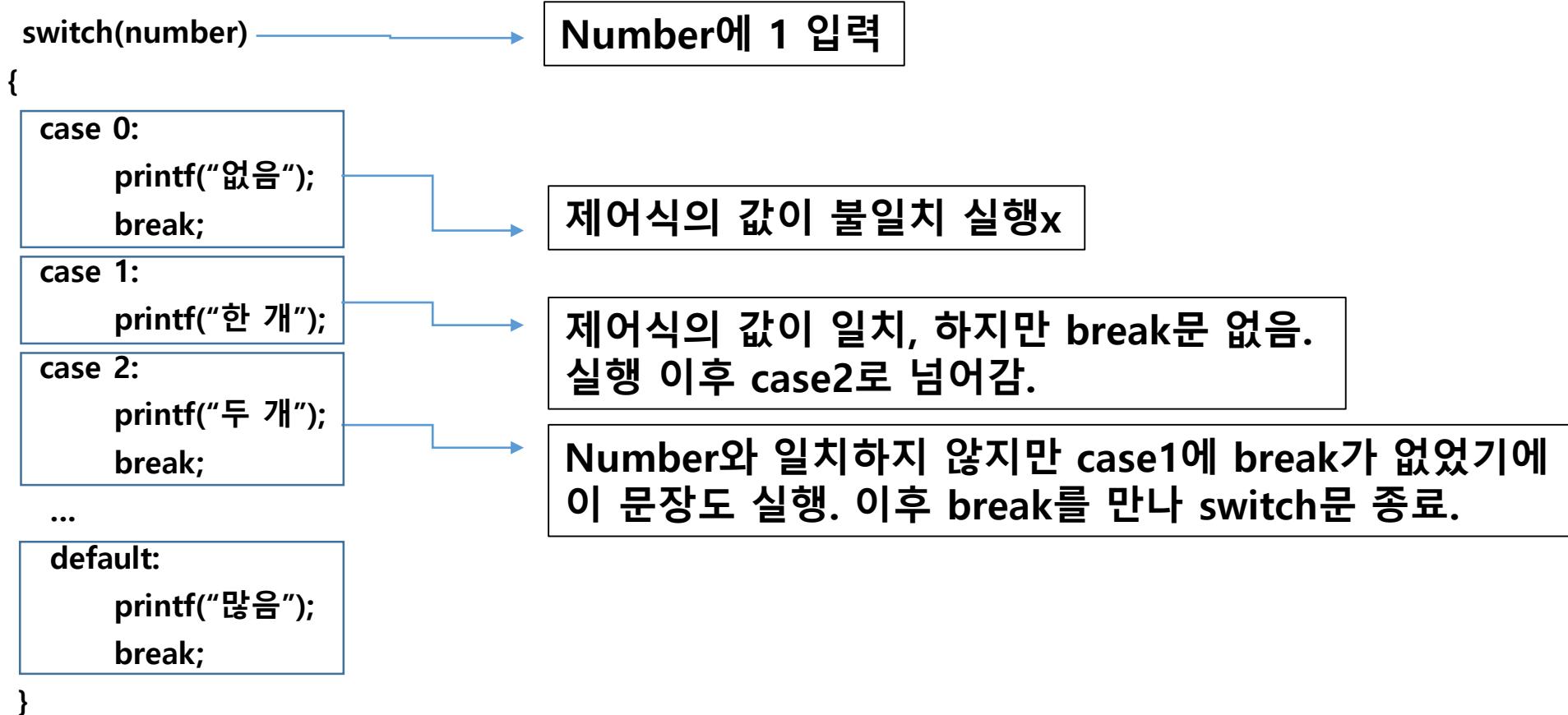
제어식의 값이 c_1 이라면 실행

제어식의 값이 c_2 이라면 실행

일치하는 값이 없으면 실행

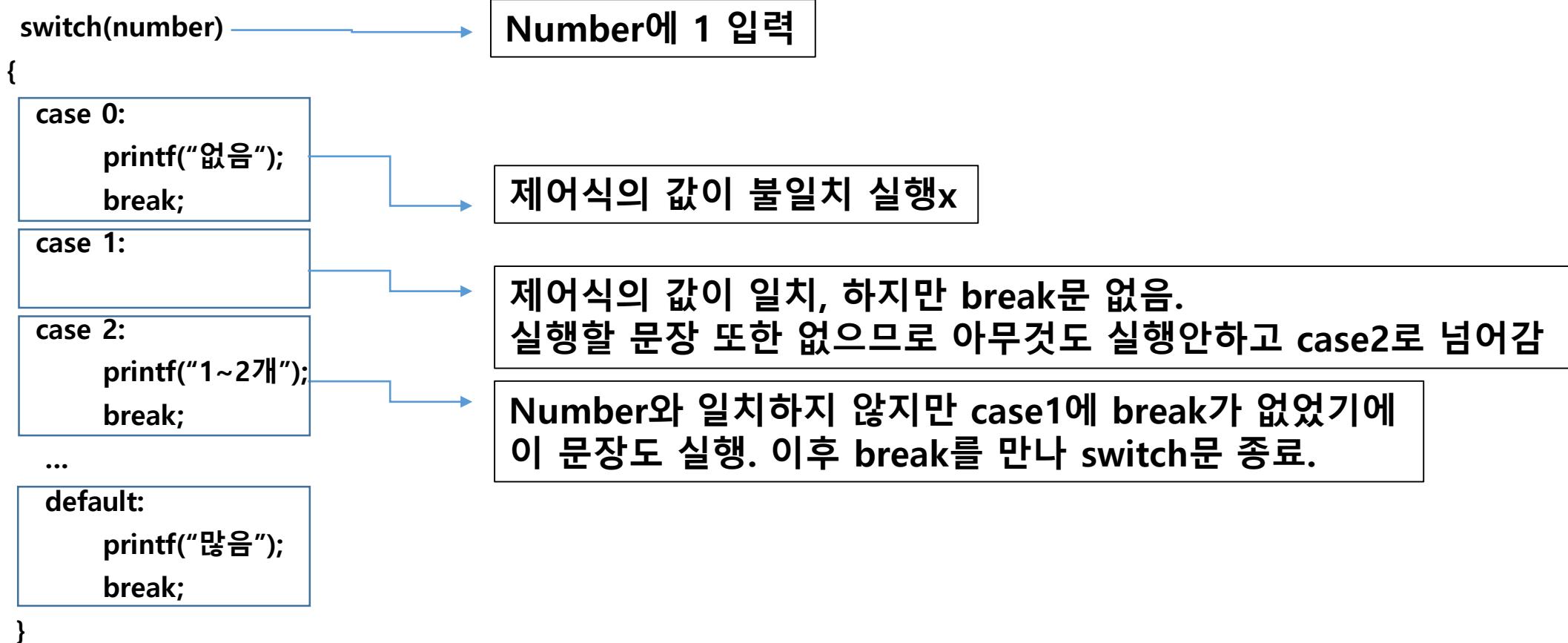
switch문 - break

: break문이 없다면?



switch문 - break

: 의도적인 break문 생략



중간 퀴즈



- 조건문 2가지와 반복문 2가지?

실습 문제

- 하나의 문자를 입력 받아 모음과 자음을 구분하는 프로그램을 작성하라. 단, switch문을 이용한다. (영어 소문자만 입력된다고 가정)

hint) 1. 문자를 입력 받을 때는 %c

2. 자음은 많으므로 default를 이용한다.

```
c:\ C:\WINDOWS\system32\cmd.exe
```

문자를 입력하세요 : a

모음입니다.

계속하려면 아무 키나

```
c:\ C:\WINDOWS\system32\cmd.exe
```

문자를 입력하세요 : k

자음입니다.

계속하려면 아무 키나 누르십시오 . . .

실습 문제

```
3 int main(void)
4 {
5     char word;
6
7     printf("문자를 입력하세요 : ");
8     scanf("%c", &word);
9
10    switch (word)
11    {
12        case 'a':
13        case 'e':
14        case 'i':
15        case 'o':
16        case 'u':
17            printf("모음입니다.\n");
18            break;
19        default:
20            printf("자음입니다.\n");
21            break;
22    }
23
24    return 0;
25 }
```

실습 문제

- 놀이공원의 자유이용권 가격을 계산하는 프로그램을 작성하라.

hint) 시간은 24시간 단위를 이용한다.(0~24)

시간을 먼저 검사한다.

구분	대인	소인(12세 미만/65세 이상)
자유이용권(오후 5시 이전)	34,000	25,000
야간이용권(오후 5시 이후)		10,000



C:\WINDOWS\system32\cmd.exe

```
현재 시간과 나이를 입력하시오(시간, 나이) : 18 33
요금은 10000원입니다.
계속하려면 아무 키나 누르십시오 . . .
```

실습 문제

```
3 int main(void)
4 {
5     int time, age;
6     int fee;
7
8     printf("현재 시간과 나이를 입력하시오(시간, 나이) : ");
9     scanf("%d %d", &time, &age);
10
11    if (time < 17)
12    {
13        if (age < 12 || age >= 65)
14        {
15            fee = 25000;
16        }
17        else
18        {
19            fee = 34000;
20        }
21    }
22    else
23    {
24        fee = 10000;
25    }
26
27    printf("요금은 %d원입니다.\n", fee);
28
29    return 0;
30 }
```

제어문

제어문 : 문장들이 실행되는 순서에 영향을 주는 문장을 의미함.

- 1) 조건문 : 어떤 조건에 따라서 여러 개의 실행 경로 가운데 하나를 선택할 때 사용한다. **If**와 **switch** 문장이 여기에 속한다.
- 2) 반복문 : 어떤 조건이 유지되는 한, 정해진 횟수만큼 처리를 되풀이하는 문장을 의미한다. **For**문과 **while**문이 여기에 속한다.

반복문의 필요성

Ex) 구구단 9단을 출력해보자.

(반복문 없이)

```
Printf("9 * 1 = 9\n");
Printf("9 * 2 = 18\n");
Printf("9 * 3 = 27\n");
...
Printf("9 * 9 = 81\n");
```

(반복문 사용)

```
i=1;
while(i<=9)
{
    Printf("9 * %d = %d \n", i, i*9);
    i++;
}
```

while 문

형식) while(조건식)

문장;

의미) 조건식이 참이면 문장이 반복 실행된다.

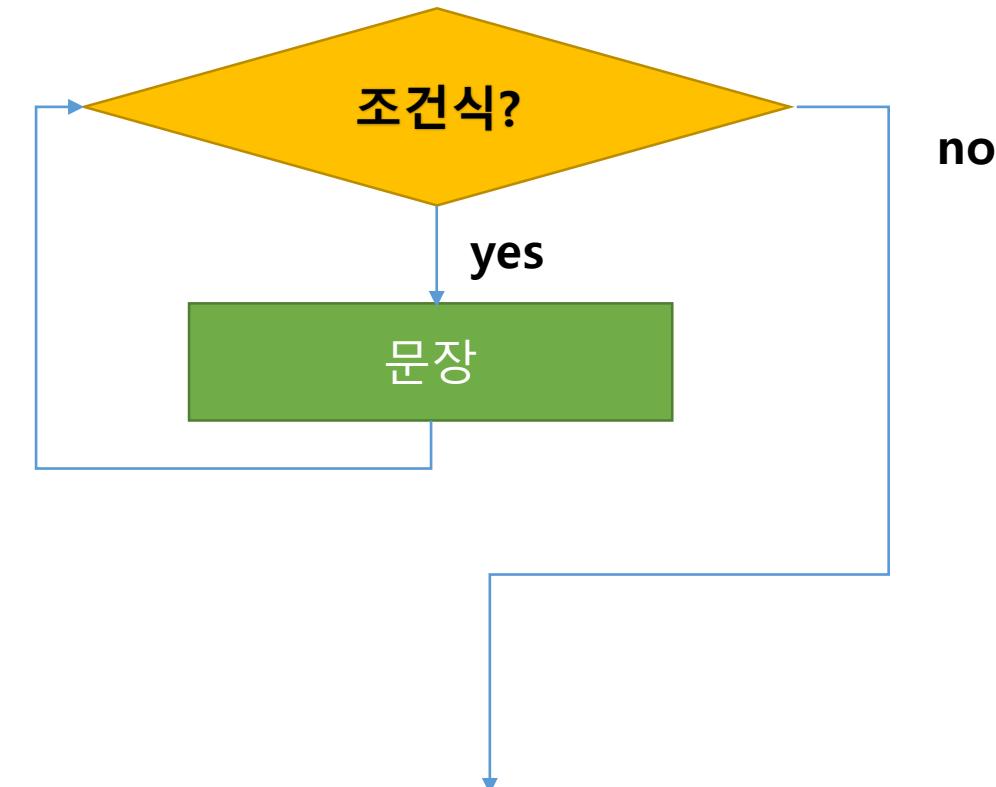
Ex) while($i \leq 9$)

{

 printf("9 * %d = %d \n", i, i*9);

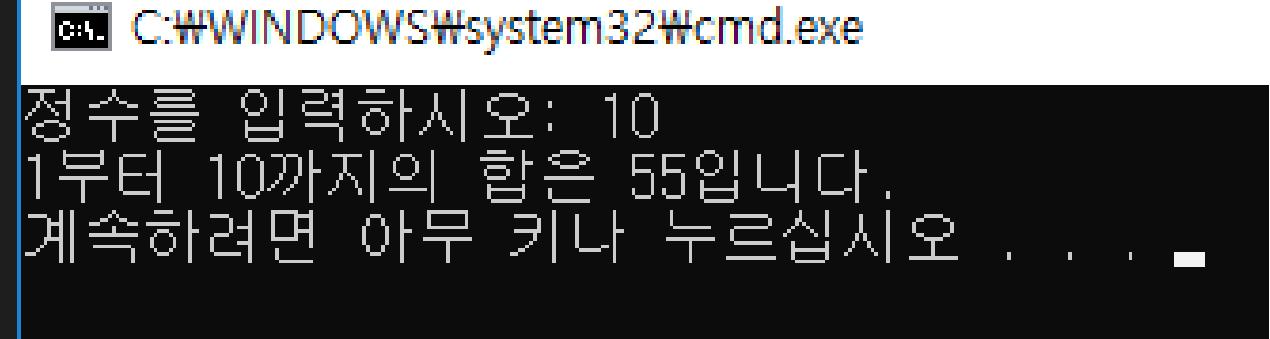
 i++;

}



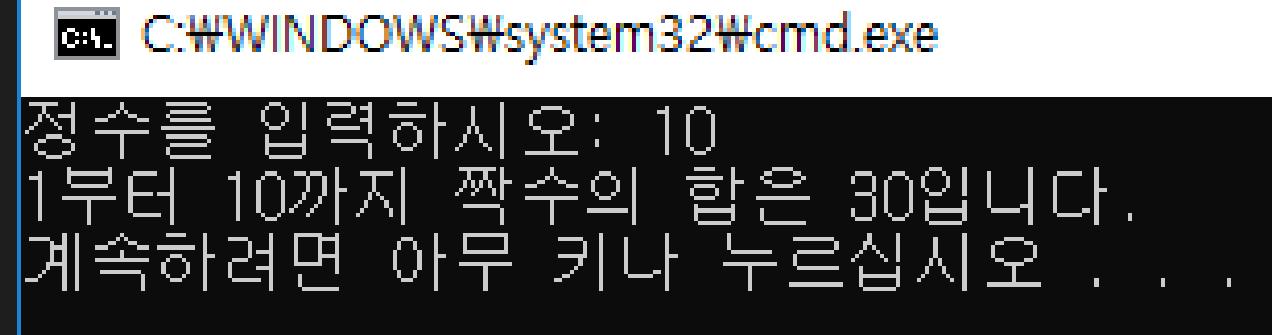
while문 - 예제

```
1 #include<stdio.h>
2
3 int main() {
4     int i, n, sum;
5
6     i = 1;
7     sum = 0;
8
9     printf("정수를 입력하시오: ");
10    scanf("%d", &n);
11
12    while (i <= n)
13    {
14        sum += i;
15        i++;
16    }
17    printf("1부터 %d까지의 합은 %d입니다.\n", n, sum);
18
19    return 0;
20 }
```



while문 - 예제

```
1  #include<stdio.h>
2
3  int main() {
4      int i, n, sum;
5
6      i = 0;
7      sum = 0;
8
9      printf("정수를 입력하시오: ");
10     scanf("%d", &n);
11
12     while (i <= n)
13     {
14         sum += i;
15         i += 2;
16     }
17     printf("1부터 %d까지 짝수의 합은 %d입니다.\n", n, sum);
18
19     return 0;
20 }
```



while문 - 센티널

데이터의 개수가 미리 알려지지 않았거나 너무 많아서 개수를 알기 어려울 때? **센티널(sentinel)** 혹은 **보초 값** 이용

: 센티널이 입력되면 데이터 입력을 중단함. 일반적으로
입력되지 않을 값을 이용

ex) 성적을 반복 입력하는 프로그램에서는 음수

while문 - 센티널

```
1 #include<stdio.h>
2
3 int main() {
4     int grade = 0;
5     int n = 0;
6     double sum = 0;
7     double average = 0;
8     printf("종료하시려면 음수를 입력하세요.\n");
9
10    while (grade >= 0) {
11        printf("성적을 입력하세요 : ");
12        scanf("%d", &grade);
13        sum += grade;
14        n++;
15    }
16    sum = sum - grade;
17    n--;
18    average = sum / n;
19    printf("성적의 평균은 %lf입니다\n", average);
20    return 0;
21 }
```

```
C:\WINDOWS\system32\cmd.exe
종료하시려면 음수를 입력하세요.
성적을 입력하세요 : 80
성적을 입력하세요 : 100
성적을 입력하세요 : 75
성적을 입력하세요 : 90
성적을 입력하세요 : -1
성적의 평균은 86.250000입니다
계속하려면 아무 키나 누르십시오 . . .
```

주의) 이러한 알고리즘 사용시 맨 마지막
grade와 n을 제외해줘야 함.

while문 - 주의할점

1) 중괄호 생략

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

조건문에서 말했듯이 그 제어문에서
같이 실행되어야 하는 것은 중괄호로 묶기!

2) 조건식 실수

```
int main() {  
    int i = 0;  
    while (i != 9)  
        i += 2;
```

조건식에서 i는 9가 절대 될 수 없으므로 무한루프에
빠지게 된다. !=나 == 보다는 <=, >= 사용이 안전
 $(i \neq 9) \rightarrow (i < 9)$

while문 - 주의할점

3) 세미콜론

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

조건문과 마찬가지로 while뒤에 세미콜론 붙이지 않음

4) = 과 == 구분하기...

```
int main() {  
    int i = 0;  
    while (i = 2);  
    {  
        printf("반복중입니다.\n");  
        i++;  
    }  
    return 0;  
}
```

i = 2는 i에 2를 대입한다는 의미로, 0이 아닌 값은 항상 참으로 인식하므로 무한루프에 빠진다.

※ 0이 아닌 값은 참, 0만 참으로 인식함.

do...while문

형식) do

반복문장;
while(조건식);

의미) 일단 문장을 실행한 후에 조건을 검사하여
반복 여부를 결정한다.

Ex) i=10;
do
{
 printf("do..while문\n");
} while(i < 3);

반복 조건이 맞지는 않지만 일단 한번 실행된다.
do..while문은 while뒤에 세미콜론을 붙인다.

for 문

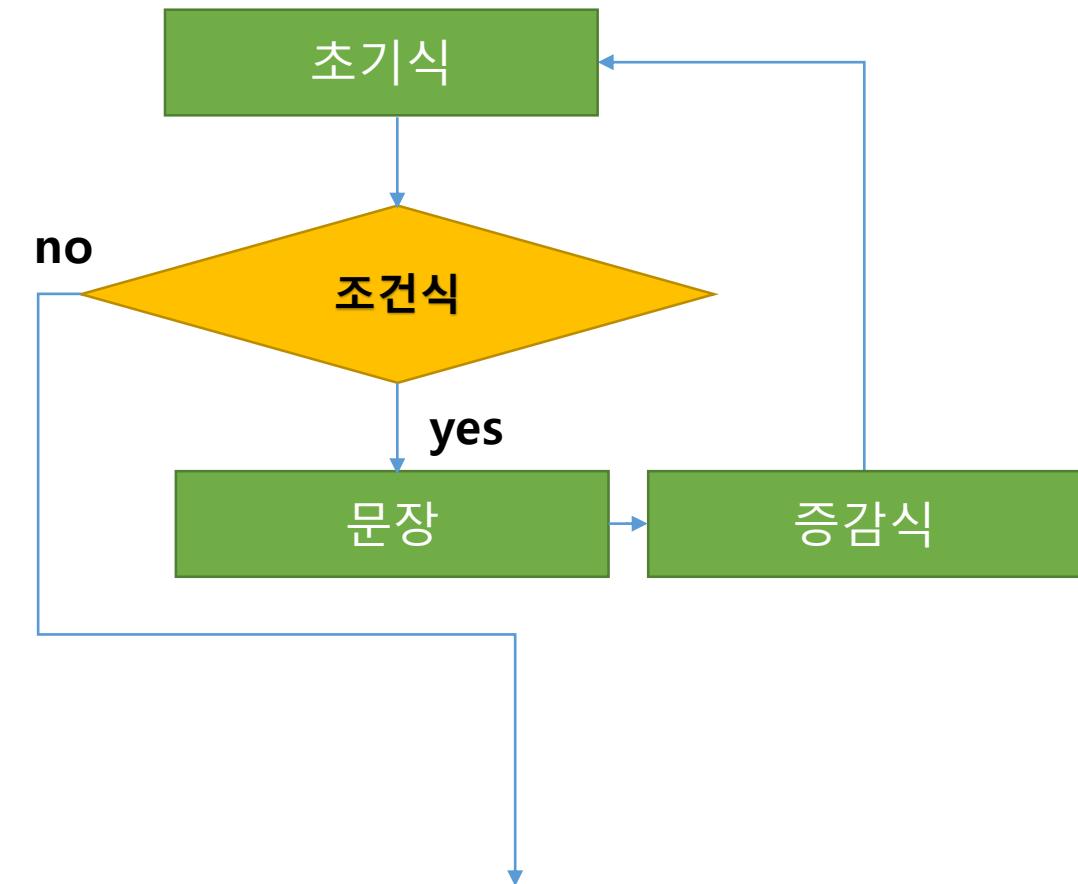
형식) for (초기식; 조건식; 증감식)

반복문장;

의미) 초기식을 실행한 후에 조건식의 값이 참인 동안 반복문장을 실행한다. 한번 반복이 끝날 때 마다 증감식이 실행된다.

Ex) for (i=0; i<10; i++)

```
printf("Hello World!");
```

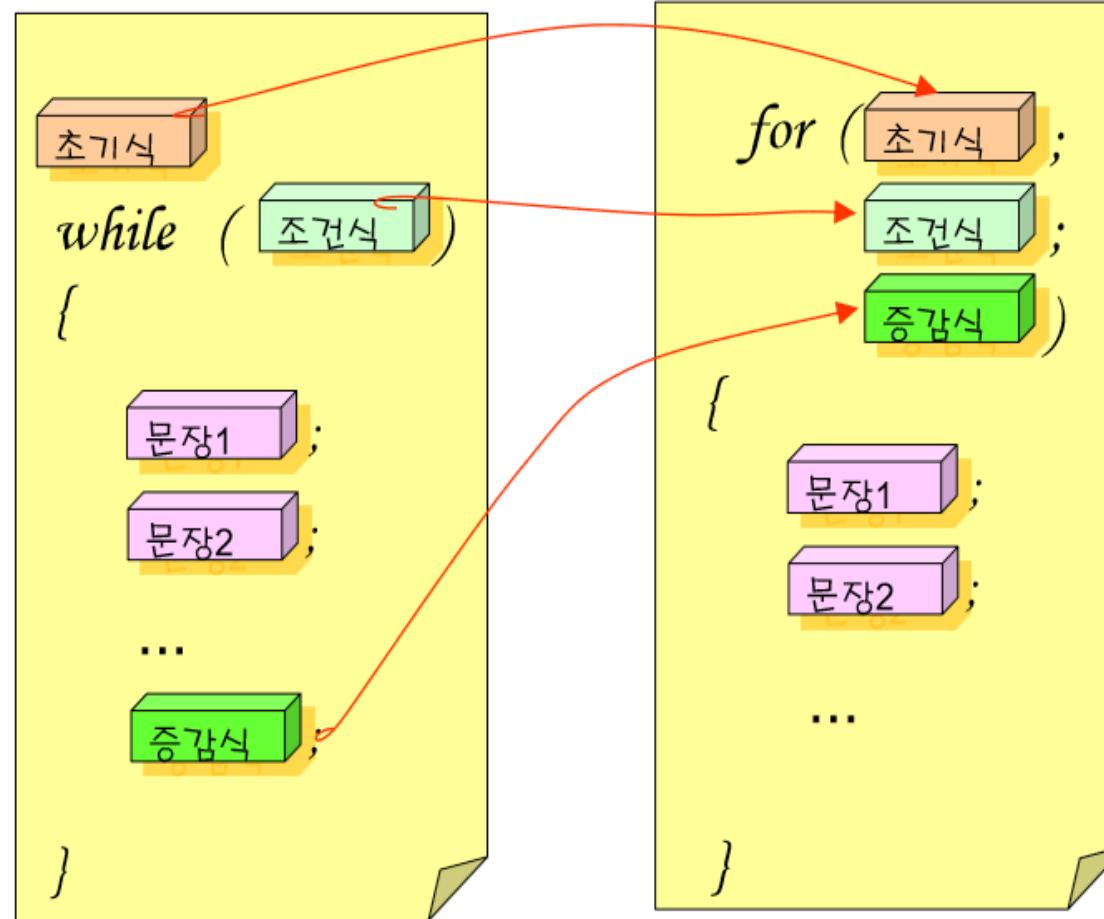


for 문 - 예제

```
1 #include<stdio.h>
2
3 int main() {
4     int fact = 1;
5     int i, n;
6     printf("정수를 입력하세요 : ");
7     scanf("%d", &n);
8
9
10    for (i = 1; i <= n; i++) {
11        fact = fact * i;
12    }
13    printf("%d!은 %d입니다.\n", n, fact);
14    return 0;
15 }
```

```
C:\WINDOWS\system32\cmd.exe
정수를 입력하세요 : 6
6!은 720입니다.
계속하려면 아무 키나 누르십시오 . . .
```

for문 - while문



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

```
for (i = 0; i < 3; i++)  
    printf("%d\n", i);  
return 0;
```

다양한 for문

초기식, 조건식, 증감식에는 아무거나 들어갈 수 있다.

예를 들어,

1. **for(; ;)와 같은 for문도 가능**
2. **for(; i<10 ; i++)와 같이 하나가 빠져도 된다.**
3. **for(printf("반복 시작\n"); i<3 ; i++)** 같이 출력문장이 들어가도 된다.
4. **for(int i=0; i<10; i++)** 과 같이 초기식에 변수를 선언하여도 된다.

But)) 초기식에서 선언한 변수 i는 for문 안에서만 사용 가능하다. 이에 관련해서는 뒤에서 배움.

중첩반복문

```
1 #include<stdio.h>
2
3 int main() {
4     int x, y;
5     for (y = 1; y <= 5; y++)
6     {
7         for (x = 0; x < y; x++)
8         {
9             printf("*");
10        }
11        printf("\n");
12    }
13    return 0;
14 }
```

```
C:\WINDOWS\system32\cmd.exe
*
**
***
****
*****
계속하려면 아무 키나 누르십시오 . . .
```

break와 continue

break 문은 루프를 빠져나갈 때 사용한다. 반복문이나 조건검사시에 유용하다.

```
1 #include<stdio.h>
2
3 int main() {
4     int a, b;
5     while (1) {
6         printf("문자와 분모를 입력하세요 : ");
7         scanf("%d %d", &a, &b);
8         if (b == 0)
9             break;
10        printf("소수로 변환하면 %.1f입니다.\n", (double)a / b);
11    }
12    return 0;
13 }
```

```
C:\WINDOWS\system32\cmd.exe
문자와 분모를 입력하세요 : 4 6
소수로 변환하면 0.666667입니다.
문자와 분모를 입력하세요 : 2 4
소수로 변환하면 0.500000입니다.
문자와 분모를 입력하세요 : 1 3
소수로 변환하면 0.333333입니다.
문자와 분모를 입력하세요 : 12 0
계속하려면 아무 키나 누르십시오 . . .
```

break와 continue

Continue 문은 반복과정의 나머지를 무시하고 다음 루프로 넘어갈 때 사용한다.

```
1 #include<stdio.h>
2
3 int main() {
4     int a, b;
5     while (1) {
6         printf("문자와 분모를 입력하세요 : ");
7         scanf("%d %d", &a, &b);
8         if (b == 0)
9             continue;
10        printf("소수로 변환하면 %.1f입니다.\n", (double)a / b);
11    }
12    return 0;
13 }
```

```
C:\WINDOWS\system32\cmd.exe
문자와 분모를 입력하세요 : 1 2
소수로 변환하면 0.50000입니다.
문자와 분모를 입력하세요 : 3 5
소수로 변환하면 0.60000입니다.
문자와 분모를 입력하세요 : 10 25
소수로 변환하면 0.40000입니다.
문자와 분모를 입력하세요 : 3 0
문자와 분모를 입력하세요 : 3 6
소수로 변환하면 0.50000입니다.
문자와 분모를 입력하세요 : 2 0
문자와 분모를 입력하세요 : ^C계속하려면 아무 키나 누르세요.
```

실습 문제

- 
- $1^2 + 2^2 + 3^2 + \dots + n^2$ 를 계산하여 출력하여라.
 - hint) i를 1부터 n까지 증가시키면서 result에 $i*i$ 를 더한다.



C:\> C:\WINDOWS\system32\cmd.exe

```
n을 입력하세요 : 10
```

```
계산값은 385입니다.
```

```
계속하려면 아무 키나 누르십시오 . . .
```

실습 문제

```
1 #include<stdio.h>
2
3 int main(void)
4 {
5     int i, n;
6     int sum = 0;
7
8     printf("정수를 입력하세요 : ");
9     scanf("%d", &n);
10
11    for (i = 1;i <= n;i++)
12    {
13        sum += i * i;
14    }
15
16    printf("계산값은 %d입니다.\n", sum);
17
18    return 0;
19 }
```

실습 문제

- 중첩반복문을 사용하여 다음과 같이 출력하는 프로그램을 작성하라.

```
C:\WINDOWS\system32\cmd.exe
정수를 입력하시오 : 5
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
계속하려면 아무 키나 누르십시오 . . .
```

실습 문제

```
1 #include<stdio.h>
2
3 int main(void)
4 {
5     int i, j, n;
6
7     printf("정수를 입력하시오 : ");
8     scanf("%d", &n);
9
10    for (i = 1;i <= n;i++)
11    {
12        for (j = 1;j <= i;j++)
13        {
14            printf("%d ", j);
15        }
16        printf("\n");
17    }
18
19    return 0;
20 }
```

다음주



함수와 변수



감사합니다