

5장 선택문(if, switch~case)



❖ if문의 기본 형식 (1/2)

if문의 기본 형식

소스 코드 예제5-1.c

01 #include <stdio.h>

02

03 int main(void)

04 {

05 int a = 20;

06 int b = 0;

07

08 if (a > 10)

09 {

10 b = a;

11 }

12

13 printf("a : %d, b : %d\n", a, b);

14

15 return 0;

16 }

조건식을 만족할 때
실행할 실행문은
중괄호로 묶어준다.

if (a > 10)

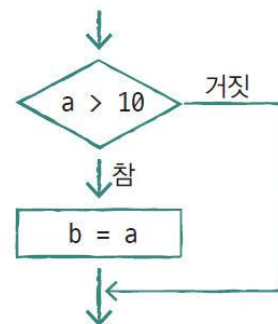
→ 조건식

{

b = a;

→ 실행문

}



// 조건식 : a가 10보다 크므로 조건식은 참

// 실행문 : b = a 대입문 실행

// 대입이 수행되면 두 값은 같음

실행결과

a : 20, b : 20

❖ if문의 기본 형식 (2/2)

- 실행문은 중괄호로 묶고 들여쓰기 한다.

```
if (a > 10)
{
    b = a;
}
```

```
if (a > 10){
    b = a;
}
```

- 실행할 문장이 한 문장이면 중괄호 생략 가능

```
if (a > 10)
    b = a;
```

```
if (a > 10) b = a;
```

- 실행할 문장이 두 문장 이상이면 반드시 중괄호 사용

```
if (a > 10)
중괄호 → {
실행문 →   b = a;           // ① a를 b에 대입하고
            printf("a와 b는 같습니다."); // ② 그 결과를 출력
중괄호 → }
```

❖ if~else문

if ~ else문의 사용

소스 코드 예제 5-2.c

```

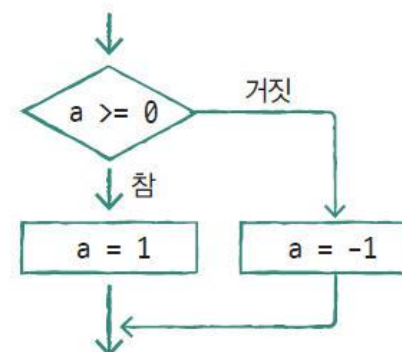
01 #include <stdio.h>
02
03 int main(void)
04 {
05     int a = 10;
06
07     if (a >= 0)
08     { // a가 0보다 크거나 같으면 a에 1 대입
09         a = 1;
10     }
11     else
12     {
13         a = -1;           // a가 0보다 작으면 a에 -1 대입
14     }
15
16     printf("a : %d\n", a);
17
18     return 0;
19 }

```

```

if (a >= 0) → 조건식
{
    a = 1; → 실행문1
}
else
{
    a = -1; → 실행문2
}

```



실행결과

a : 1

❖ if~else if~else문 (1/2)

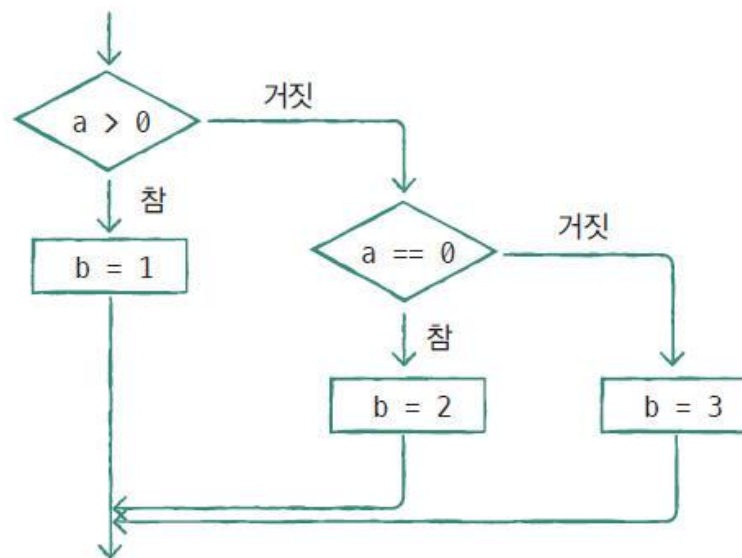
if ~ else if ~ else문 사용

소스 코드 예제5-3.c

```

01 #include <stdio.h>
02
03 int main(void)
04 {
05     int a = 0, b = 0;
06
07     if (a > 0)           → 조건식1
08     {                   → 실행문1
09         b = 1;
10     }
11     else if (a == 0)     → 조건식2
12     {                   → 실행문2
13         b = 2;
14     }
15     else
16     {                   → 실행문3
17         b = 3;
18     }
19
20     printf("b : %d\n", b); // if문으로 결정된 b 값 출력
21
22     return 0;
23 }

```



실행결과

b : 2

❖ if~else if~else문 (2/2)

- 조건식을 차례로 검사하므로 이전 조건의 결과가 반영된다.

```
if (a >= 10)
{
    b = 1;
}
else if (a >= 0)
{
    b = 2;
}
else
{
    b = 3;
}
```

같다

```
if (a >= 10)
{
    b = 1;
}
else if ((a < 10) && (a >= 0))
{
    b = 2;
}
else if (a < 0)
{
    b = 3;
}
```

키워드로 끝내는 핵심 포인트

- ❖ 조건에 따라 실행 문장을 선택할 때 **선택문**을 사용한다.
- ❖ **if문**은 한 가지의 선택을 고민할 때 사용한다.
- ❖ **if~else문**은 둘 중에 하나를 고를 때 사용한다.
- ❖ **if~else if~else문**은 세 가지 이상에서 하나를 고를 때 사용한다.

표로 정리하는 핵심 포인트

표 5-1 if문의 형식

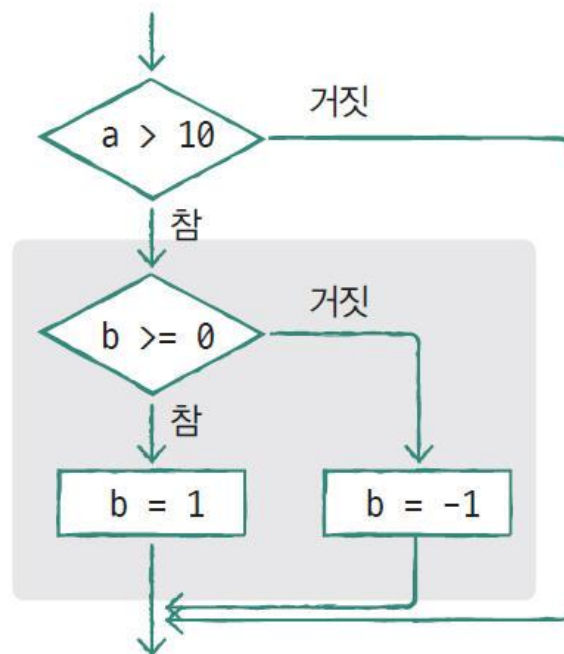
구분 형식	실행 방식
if (조건식) 실행문;	조건식이 참이면 실행문 실행 거짓이면 아무것도 실행하지 않음
if (조건식) 실행문1; else 실행문2;	조건식이 참이면 실행문1 실행 거짓이면 실행문2 실행
if (조건식1) 실행문1; else if (조건식2) 실행문2; else 실행문3;	조건식1이 참이면 실행문1만 실행 조건식1이 거짓이고 조건식2가 참이면 실행문2만 실행 모든 조건식이 거짓이면 실행문3 실행

❖ if문 중첩 (1/4)

- if문의 실행문으로 if문 사용

```
if (a > 10) → 조건식  
{  
    if (b >= 0) {  
        b = 1;  
    }  
    else {  
        b = -1;  
    }  
}
```

실행문



❖ if문 중첩 (2/4)

중첩된 if문

소스 코드 예제5-4.c

```
01 #include <stdio.h>
02
03 int main(void)
04 {
05     int a = 20, b = 10;
06
07     if (a > 10)           // a가 10보다 크면 9 ~ 16행 실행, 작거나 같으면 18행으로 이동
08     {
09         if (b >= 0)       // b가 0 이상이면 b에 1 대입하고 18행으로 이동
10         {
11             b = 1;
12         }
13         else
14         {
15             b = -1;       // b가 0보다 작으면 b에 -1 대입하고 18행으로 이동
16         }
17     }
18
19     printf("a : %d, b : %d\n", a, b);
20
21     return 0;
22 }
```

실행결과

a : 20, b : 1

❖ if문 중첩 (3/4)

- 선행조건이 있으면 불필요한 조건 검사를 하지 않는다.

```
if (a > 10)
{
    if (b >= 0)
    {
        b = 1;
    }
    else
    {
        b = -1;
    }
}
```

같다

```
if ((a > 10) && (b >= 0))
    // a가 10보다 크고 b가 0 이상이면 b에 1 대입
{
    b = 1;
}
if ((a > 10) && (b < 0))
    // a가 10보다 크고 b가 0보다 작으면 b에 -1 대입
{
    b = -1;
}
```

❖ if문 중첩 (4/4)

- 실행 효율을 높일 수 있다.

```
if (a == 1)
{
    printf("일");
}
else if (a == 2)
{
    printf("이");
}
else if (a == 3)
{
    printf("삼");
}
else if (a == 4)
{
    printf("사");
}
else if (a == 5)
{
    printf("오");
}
else
{
    printf("육");
}
```



```
if (a <= 3)
{
    if (a == 1)
    {
        printf("일");
    }
    else if (a == 2)
    {
        printf("이");
    }
    else
    {
        printf("삼");
    }
}
```

```
else
{
    if (a == 4)
    {
        printf("사");
    }
    else if (a == 5)
    {
        printf("오");
    }
    else
    {
        printf("육");
    }
}
```

❖ else 결합 문제

- 중첩 if문의 중괄호가 없으면 else는 가장 가까운 if와 결합된다.

중첩 if문에서 중괄호가 반드시 필요한 경우

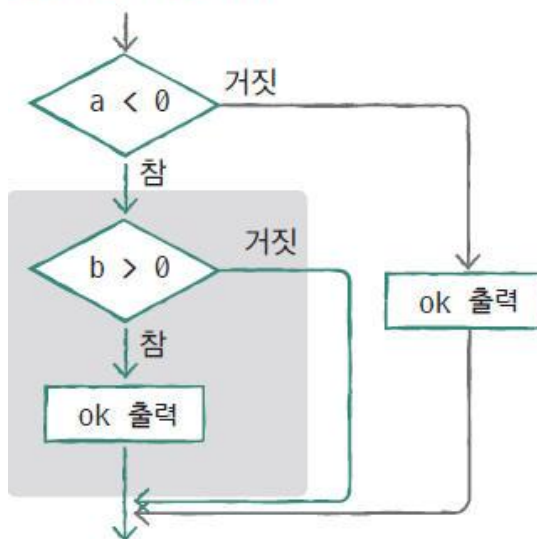
소스 코드 예제 5-5.c

```

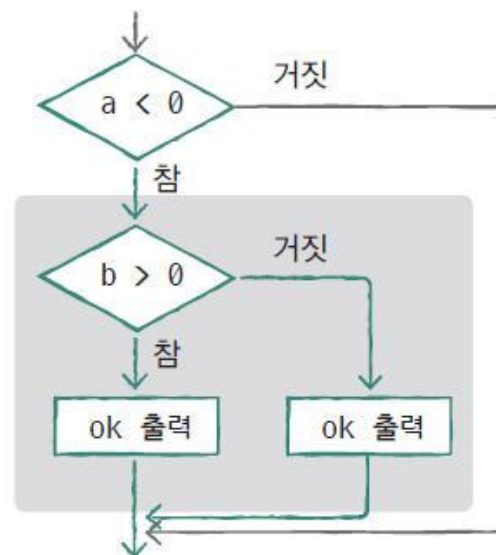
01 #include <stdio.h>
02
03 int main(void)
04 {
05     int a = 10, b = 20;
06
07     if (a < 0)
08     {
09         if (b > 0)
10         {
11             printf("ok");
12         }
13     }
14     else
15     {
16         printf("ok");
17     }
18
19     return 0;
20 }

```

중괄호를 사용한 경우

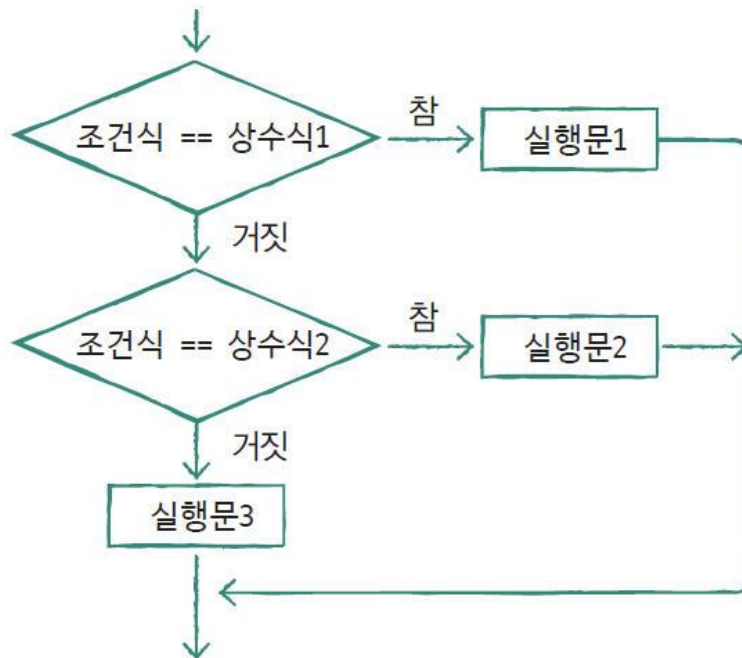


중괄호를 생략한 경우



❖ switch~case문

```
switch ( 조건식 )  
{  
  case 상수식1:  
    실행문1;  
    break;  
  case 상수식2:  
    실행문2;  
    break;  
  default:  
    실행문3;  
    break;  
}
```



❖ switch~case문

switch ~ case문의 사용 소스 코드 예제5-6.c

```

01 #include <stdio.h>
02
03 int main(void)
04 {
05     int rank = 2, m = 0;
06
07     switch (rank)                // rank의 값이 얼마인지 확인
08     {
09         case 1:                  // rank가 1이면
10             m = 300;             // m = 300을 수행하고
11             break;              // 블록을 벗어나 22행으로 이동
12         case 2:                  // rank가 2면
13             m = 200;             // m = 200을 수행하고
14             break;              // 블록을 벗어나 22행으로 이동
15         case 3:                  // rank가 3이면
16             m = 100;             // m = 100을 수행하고
17             break;              // 블록을 벗어나 22행으로 이동
18         default:                 // rank와 일치하는 case의 값이 없으면
19             m = 10;             // m = 10을 수행하고
20             break;              // 블록을 벗어나 22행으로 이동
21     }

```

```

22
23     printf("m : %d\n", m);
24
25     return 0;
26 }

```

 실행결과 

m : 200

❖ switch~case문

- break를 생략하면 모든 문장 실행
break가 하나도 없는 경우

① rank의
값이 2

② case 2로
이동

```
switch (rank)
{
    case 1:
        m += 300;
    case 2:
        m += 200;
    case 3:
        m += 100;
}
```

③ m에 200을
더하고 이어서
m에 100도
더합니다.

키워드로 끝내는 핵심 포인트

- ❖ **if문 중첩**은 if문 안에 실행문으로 if문을 사용한 것을 말한다.
- ❖ **switch~case문**은 정수 값으로 실행할 문장을 결정한다.
- ❖ **break**를 생략할 때는 설명과 함께 제한적으로 사용해야 한다.
- ❖ **default**의 위치는 블록 안 어디에 와도 상관 없으나
주로 마지막에 두어 예외 상황을 처리한다.

표로 정리하는 핵심 포인트

표 5-2 if문이 중첩된 예

구분	if문에 if ~ else문 중첩	if ~ else문에 if문 중첩
사용 예	<pre> if (조건식1) { if (조건식2) { 실행문1; } else { 실행문2; } } </pre>	<pre> if (조건식1) { if (조건식2) { 실행문1; } } else { 실행문2; } </pre>
실행 과정	조건식1이 거짓이면 아무것도 실행하지 않음	조건식1이 참이어도 조건식2가 거짓이면 아무것도 실행하지 않음