

# 조건에 따른 흐름의 분기 (1)

- if 문, if ~ else 문, if ~ else if ~ else 문 -

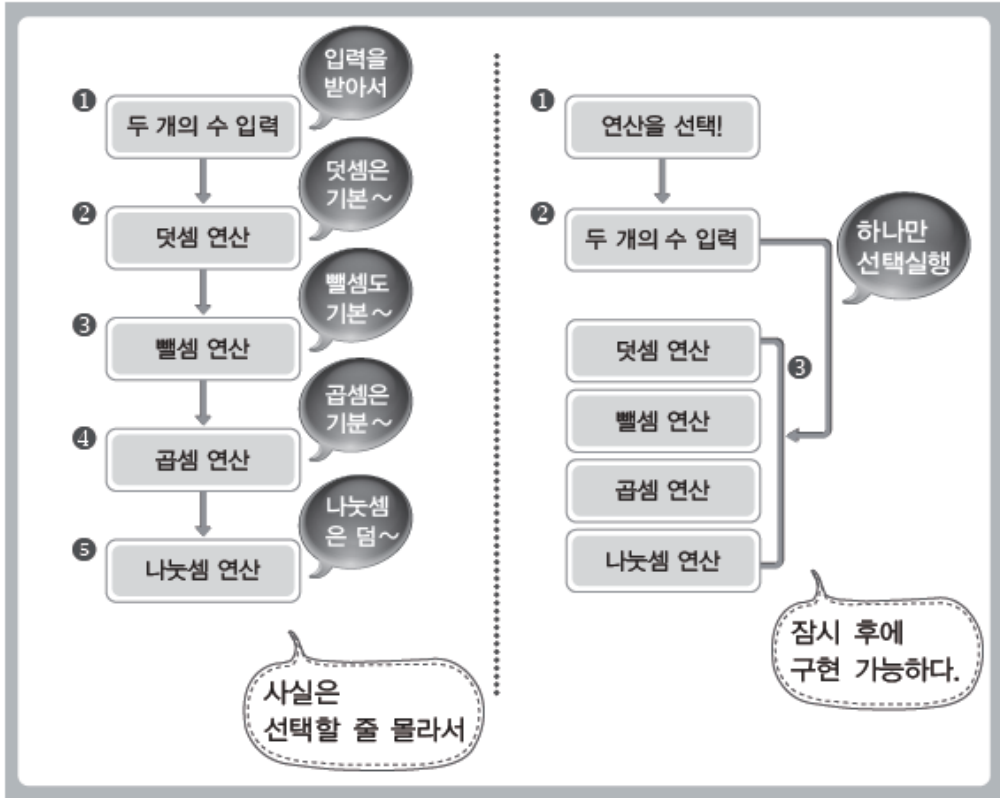
성공회대학교 IT융합자율학부  
소프트웨어공학전공  
홍 성 준



## 조건적 실행과 흐름의 분기

### ◎ 흐름의 분기가 필요한 이유

- 계산기 프로그램



- 프로그램을 구현할 때 상황에 따라 선택적으로 실행해야 하는 영역이 존재



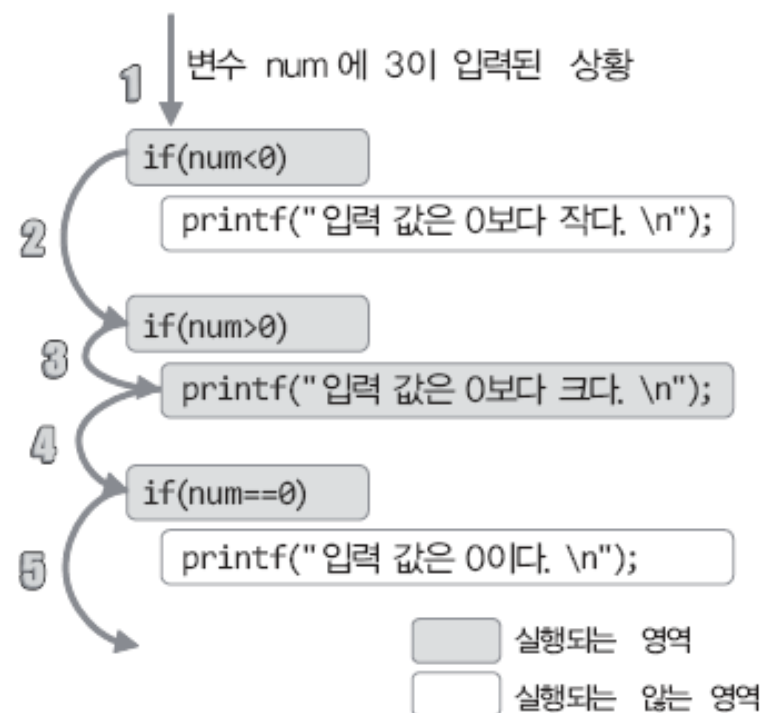
## 조건적 실행과 흐름의 분기

### ◎ if 문을 이용한 조건적 실행

- if (실행 조건) : 실행 조건이 참인 경우에만 명령을 실행

```
if(num1>num2)
    printf("num1이 num2보다 큽니다. \n");
```

```
if(num1>num2)
{
    printf("num1이 num2보다 큽니다. \n");
    printf("%d > %d \n", num1, num2);
}
```





## 조건적 실행과 흐름의 분기

### ◎ if 문을 이용한 조건적 실행

- if (실행 조건) : 실행 조건이 참인 경우에만 명령을 실행
- IfBasic.c

```
int main(void)
{
    int num;
    printf("정수 입력: ");
    scanf("%d", &num);

    if(num<0)    // num이 0보다 작으면 아래의 문장 실행
        printf("입력 값은 0보다 작다. \n");

    if(num>0)    // num이 0보다 크면 아래의 문장 실행
        printf("입력 값은 0보다 크다. \n");

    if(num==0)   // num이 0이면 아래의 문장 실행
        printf("입력 값은 0이다. \n");

    return 0;
}
```

정수 입력: 3  
입력 값은 0보다 크다.

정수 입력: 0  
입력 값은 0이다.



## 조건적 실행과 흐름의 분기

### ◎ if 문을 이용한 계산기 프로그램

- CalOne.c

```
int main(void)
{
    int opt;
    double num1, num2;
    double result;

    printf("1.덧셈 2.뺄셈 3.곱셈 4.나눗셈 \n");
    printf("선택? ");
    scanf("%d", &opt);
    printf("두 개의 실수 입력: ");
    scanf("%lf %lf", &num1, &num2);

    if(opt==1)
        result = num1 + num2;
    if(opt==2)
        result = num1 - num2;
    if(opt==3)
        result = num1 * num2;
    if(opt==4)
        result = num1 / num2;

    printf("결과: %f \n", result);
    return 0;
}
```

1.덧셈 2.뺄셈 3.곱셈 4.나눗셈  
선택? 3  
두 개의 실수 입력: 2.14 5.12  
결과: 10.956800



## 조건적 실행과 흐름의 분기

### ◎ if ~ else 문을 이용한 흐름의 분기

- if (실행 조건)

{ 참인 경우 실행할 코드 블록 }

else

{ 거짓인 경우 실행할 코드 블록 }

```
if(num1>num2)
{
    // if 블록
    printf("num1이 num2보다 큽니다. \n");
    printf("%d > %d \n", num1, num2);
}
else
{
    // else 블록
    printf("num1이 num2보다 크지 않습니다. \n");
    printf("%d <= %d \n", num1, num2);
}
```



## 조건적 실행과 흐름의 분기

◎ if ~ else 문을 이용한 흐름의 분기

- IfElseBasic.c

```
int main(void)
{
    int num;
    printf("정수 입력: ");
    scanf("%d", &num);
    if(num<0)
        printf("입력 값은 0보다 작다. \n");
    else
        printf("입력 값은 0보다 작지 않다. \n");

    return 0;
}
```

정수 입력: 7

입력 값은 0보다 작지 않다.



## 조건적 실행과 흐름의 분기

### ◎ if ~ else if ~ else 문을 이용한 흐름의 분기

- if (실행 조건1)  
{ 실행 조건 1이 참인 경우 실행할 코드 블록 }
- else if (실행 조건2)  
{ 실행 조건 2가 참인 경우 실행할 코드 블록 }
- else if (실행 조건 3)  
{ 실행 조건 k가 참인 경우 실행할 코드 블록 }
- else  
{ 모두 불 만족(거짓)인 경우 실행할 코드 블록 }

```
if( 조건 1 )  
{  
    // 조건 1만족 시 실행  
}  
else if( 조건 2 )  
{  
    // 조건 2 만족 시 실행  
}  
else if( 조건 3 )  
{  
    // 조건 3 만족 시 실행  
}  
else  
{  
    // 모두 불 만족 시 실행  
}
```

얼마든지 추가 삽입 가능!

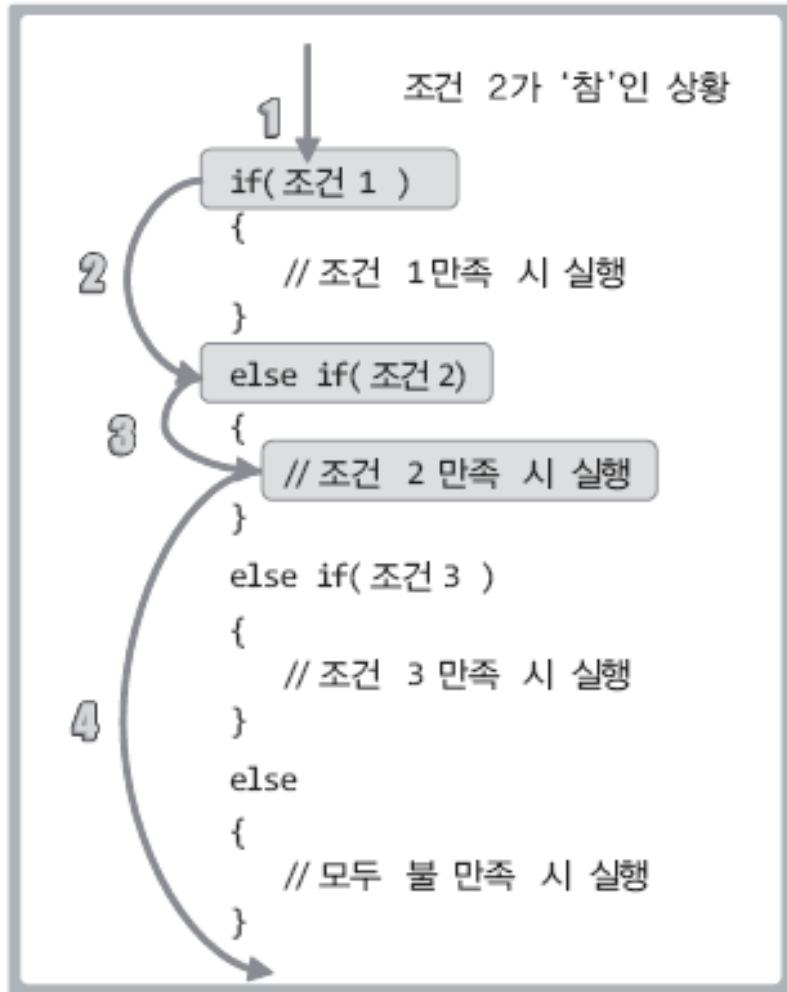
```
else if( 조건 4 )  
{  
    // 조건 4 만족 시 실행  
}
```





## 조건적 실행과 흐름의 분기

◎ if ~ else if ~ else 문을 이용한 흐름의 분기





## 조건적 실행과 흐름의 분기

◎ if ~ else if ~ else 문을 이용한 흐름의 분기

- CalTwo.c

```
int main(void)
{
    int opt;
    double num1, num2;
    double result;
    printf("1.덧셈 2.뺄셈 3.곱셈 4.나눗셈 \n");
    printf("선택? ");
    scanf("%d", &opt);
    printf("두 개의 실수 입력: ");
    scanf("%lf %lf", &num1, &num2);

    if(opt==1)
        result = num1 + num2;
    else if(opt==2)
        result = num1 - num2;
    else if(opt==3)
        result = num1 * num2;
    else
        result = num1 / num2;

    printf("결과: %f \n", result);
    return 0;
}
```



## 조건적 실행과 흐름의 분기

### ◎ if ~ else if ~ else 문을 이용한 흐름의 분기

- if ~ else if ~ else 문은 if ~ else 문의 중첩 시킨 형태임

```
if(num<0)
    printf("입력 값은 0보다 작다. \n");
else if(num>0)
    printf("입력 값은 0보다 크다. \n");
else
    printf("입력 값은 0이다. \n");
```

```
if(num<0)
{
    printf("입력 값은 0보다 작다. \n");
}
else
{
    if(num>0)
        printf("입력 값은 0보다 크다. \n");
    else
        printf("입력 값은 0이다. \n");
}
```