

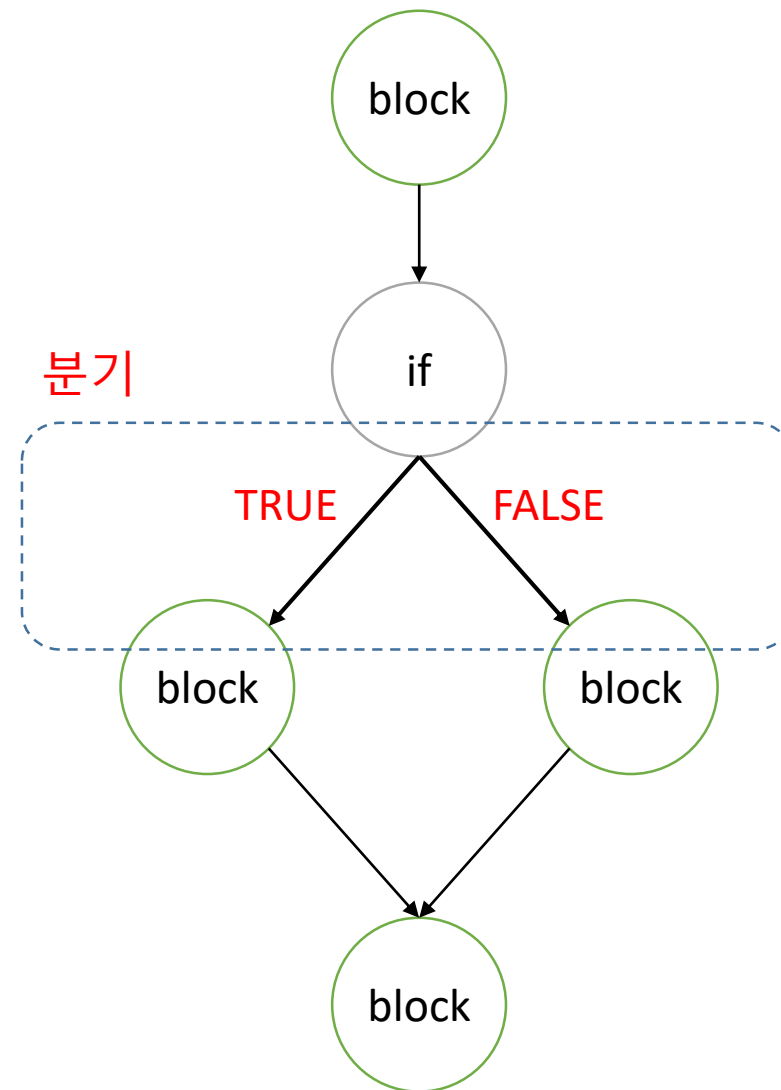
분기커버리지와 MC/DC



1) 분기커버리지

■ 분기란?

프로그래밍 언어에서 **분기**란
코드의 수행흐름이 특정 지점에서 2개 이상으로
분리될 때 **분리된 흐름을 분기**라 한다.
위에서 말한 특정 지점은 일반적으로
if, else if, switch, while, for, do-while를
말한다.



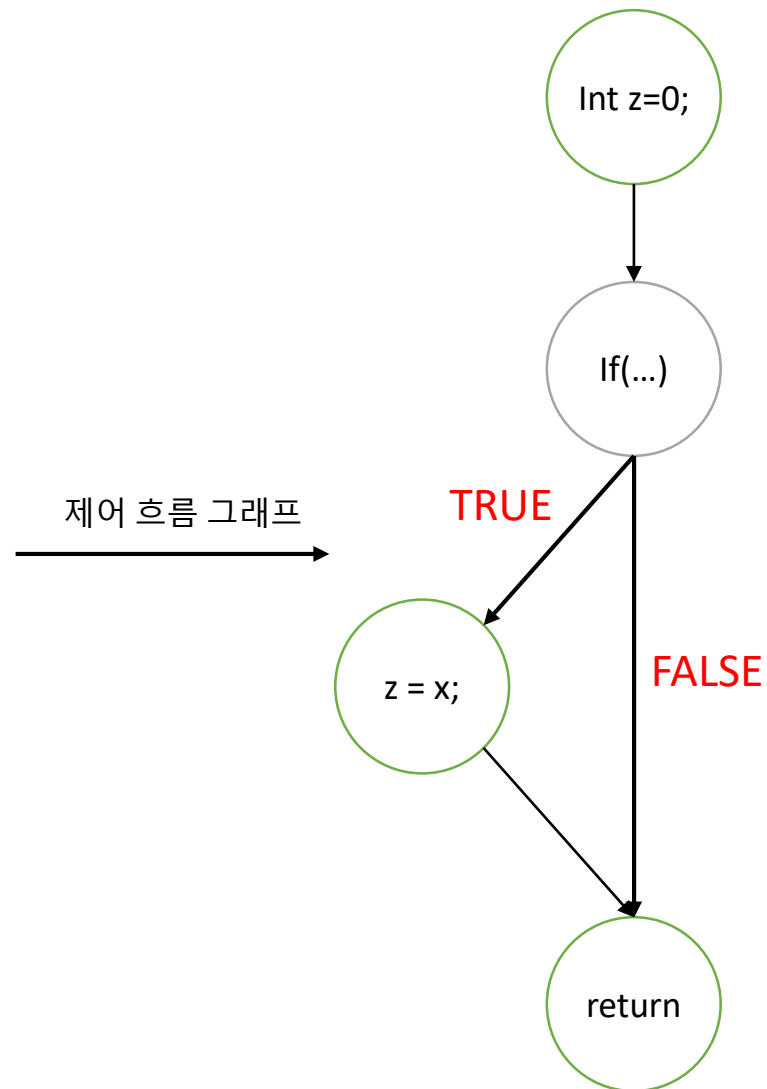
1) 분기커버리지

- 분기 커버리지란?

모든 분기중 수행한 분기의 수행률을 나타낸다.

Ex)

```
int foo (int x, int y)
{
    int z = 0;
    if ((x>0) && (y>0))
    {
        z = x;
    }
    return z;
}
```

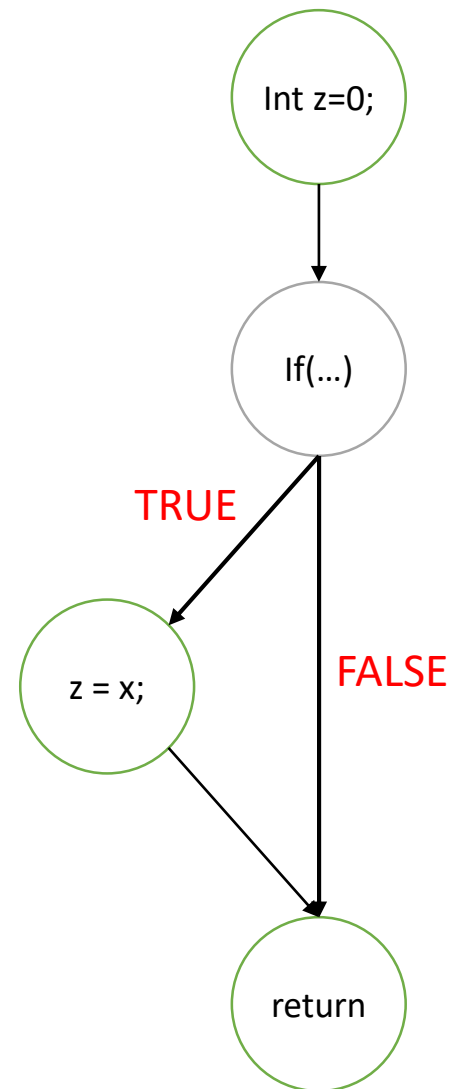


1) 분기커버리지

- foo함수에
if ((x>0) && (y>0))와 같은 분기문이 있습니다.

테스트 시나리오

- foo함수내 분기수는 2개
- 테스트 하지 않을 경우 2개중 0개 수행하여
분기 커버리지는 0%
- 1차 테스트 하여
x 에 1 y에 1을 입력하여 테스트시
분기문에서 TRUE가 수행되어
분기 2개중 1개를 수행하여
분기 커버리지는 50%
- 2차 테스트 하여
x 에 0 y에 0을 입력하여 테스트시
분기문에서 FALSE가 수행되어
분기 2개중 1개를 수행하여
분기 커버리지는 50%
- 1차 2차 테스트를 통합하여(TRUE+FALSE)
분기 2개중 2개를 수행하여
분기 커버리지는 100%



2) MC/DC

- Condition이란?

프로그래밍 언어에서 condition이란
TRUE/FALSE를 표현할 수 있는 가장 하위 수준입니다.

ex)

if($a > 0$) 과 같은 분기문이 있을 때
condition은 " $a > 0$ " 이고

if($(a > 0) \&\& (b > 0)$) 과 같은 분기문에서는
condition은 " $a > 0$ "과 " $b > 0$ "2개 입니다.

2) MC/DC

- MC/DC (Modified condition_decision coverage) 란 ?

특정 조건(condition)이 TRUE/FALSE를 수행할 때

분기(결과)도 TRUE/FALSE를 수행하게 될 때

특정 조건이 MC/DC를 만족했다고 한다.
MC/DC 측정 대상은 condition이며 (분모)

MC/DC 정의를 만족한 condition을 (분자)
계산한다.

우측과 같은 예시코드가 있을 때

condition은 "x>0"과 "y>0" 2개이다.

if ((x>0) && (y>0)) 의 분기문에서
MC/DC를 만족하는 경우를

다음장에서 계속

```
int foo (int x, int y)
{
    int z = 0;
    if ((x>0) && (y>0))
    {
        z = x;
    }
    return z;
}
```

2) MC/DC

```
int foo (int x, int y)
{
    int z = 0;
    if ((x>0) && (y>0))
    {
        z = x;
    }
    return z;
}
```

TruthTable

case	condition1 (x>0)	condition2 (y>0)	decision ((x>0)&&(y>0))
1	TRUE	TRUE	TRUE
2	TRUE	FALSE	FALSE
3	FALSE	TRUE	FALSE
4	FALSE	FALSE	FALSE

- “if ((x>0) && (y>0))” 에 대한 TruthTable(진리표)를 우측 표와 같다.

위 Table에서 나타나듯이 분기문(if)에서 각 condition에 상태의 조합을 나타내고, 각 case에 대해 숫자를 부여했다.

“x>0” condition의 MC/DC를 만족하는 case의 조합은 1과 3이다. 아래는 그 설명이다.

“y>0”의 condition을 상태는 변경 TRUE값만을 가진 상태에서

“x>0”의 condition이 TRUE/FALSE를 상태 값을 가지고

분기(결과)가 TRUE/FALSE 모두 가지게 됨으로 MC/DC를 만족한다.

같은 방식으로 “y>0” condition의 MC/DC를 만족하는 case의 조합은 1과 2다.

“x>0”의 condition은 TRUE이고, “y>0”만의 condition 변경으로 decision에 영향을 끼쳤기 때문이다

Software for safe world

Thank you for your kind attention

www.suresofttech.com

3) 참조문서

- 분기 커버리지 참조 문서
"Code coverage - Wikipedia, the free encyclopedia.html"
-> 일반적인 code coverage에 대한 설명이 적혀 있음
- MC/DC 참조 문서
"Modified condition_decision coverage
- Wikipedia, the free encyclopedia.html"