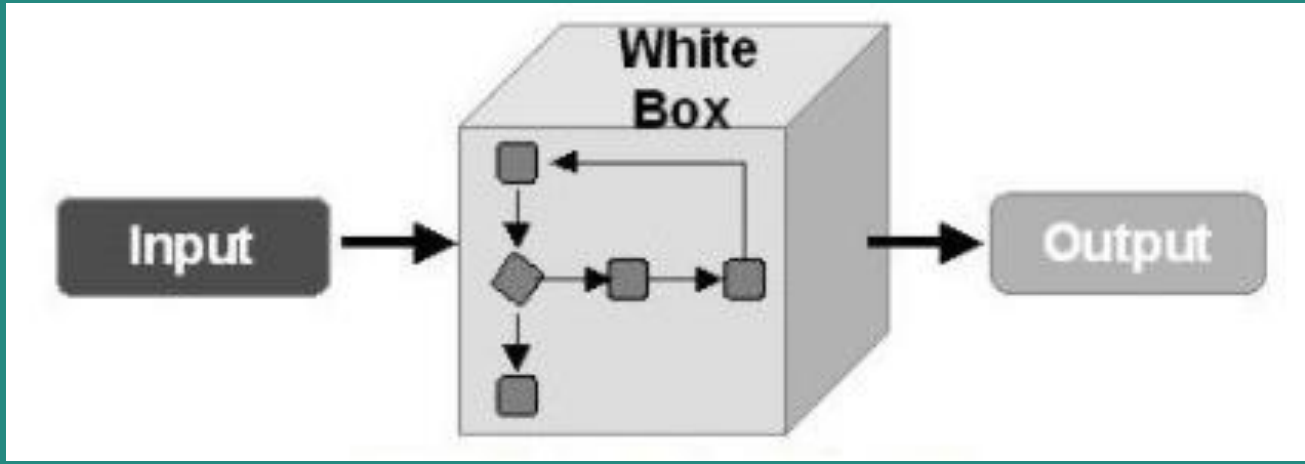
Software Testing

White Box Testing Techniques



Three types of systematic technique

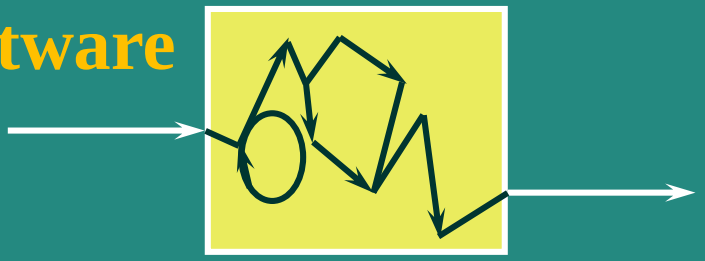
Static (non-execution)

- examination of documentation, source code listings, etc.

Functional (Black Box)

- based on functionality of software

Structural (White Box)



- based on structure of software

White Box test design techniques

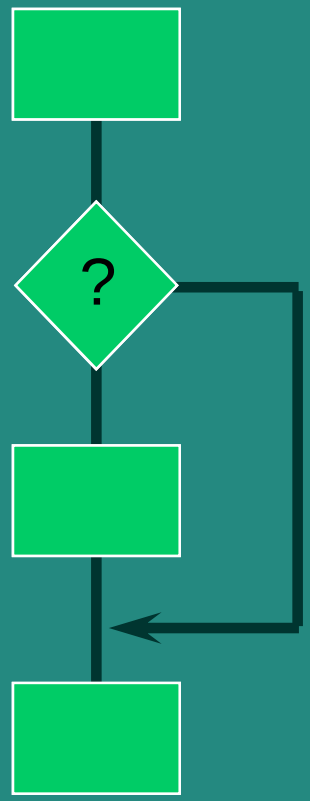
- **Statement** coverage testing
- **Path** coverage testing
- **Condition** coverage testing

Statement coverage

- percentage of executable statements exercised by a test suite

$$= \frac{\text{number of statements exercised}}{\text{total number of statements}}$$

- example:
  - program has 100 statements
  - tests exercise 87 statements
  - statement coverage = 87%



Example of statement coverage

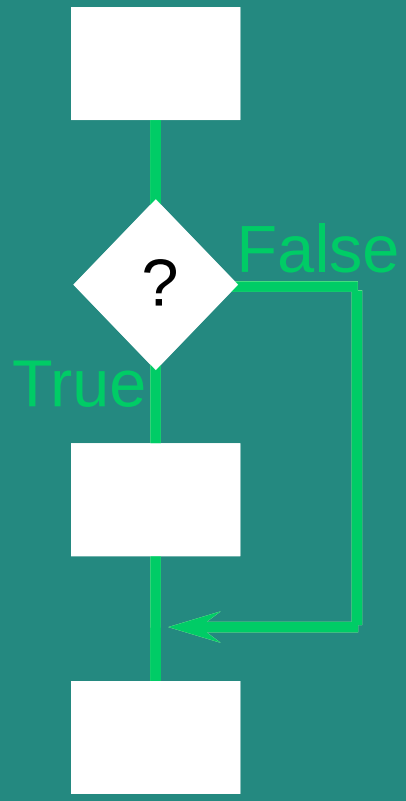
1	**read(a)**
2	**IF a > 6 THEN**
3	**b = a**
4	**ENDIF**
5	**print b**

Test case	Input	Expected output
1	7	7

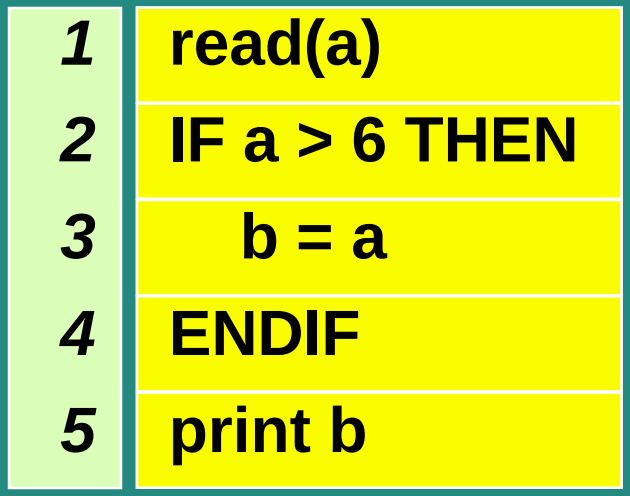
As all 5 statements are ‘covered’ by this test case, we have achieved 100% statement coverage

Path coverage

1	**read(a)**
2	**IF a > 6 THEN**
3	**b = a**
4	**ENDIF**
5	**print b**



Example of path coverage



Test case	Input	Expected output
1	7	7

As all 5 statements (1 path) are ‘covered’ by this test case, we have achieved 50% path coverage

Condition Coverage

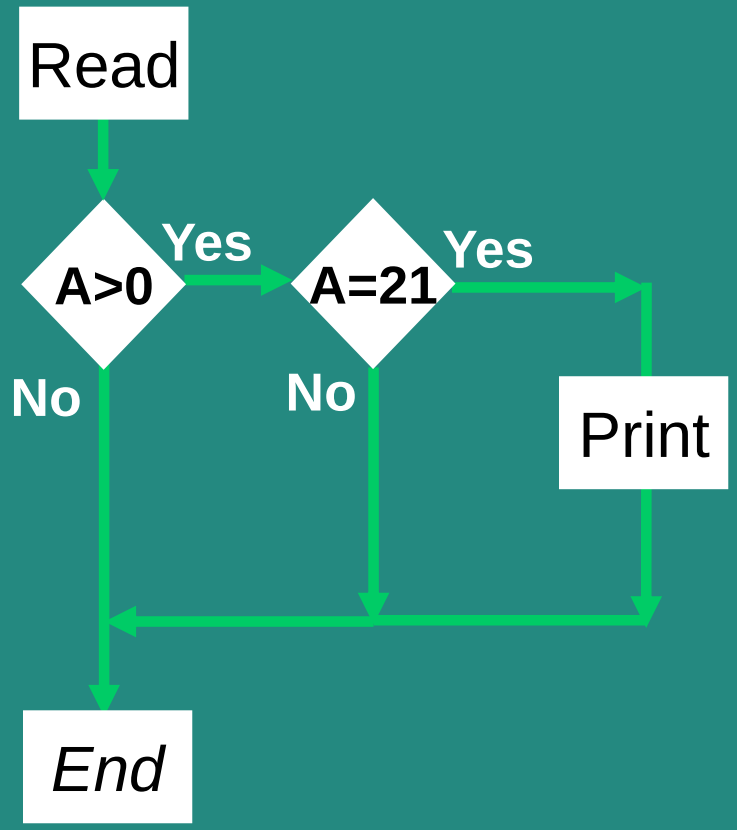
```
1    IF (x < y) AND (a>b) THEN
         Print(a+x)
```

- 4 possible combinations: TT, FF, TF, FT
- Decision tables

x < y	F	T	T
a > b	-	F	T
(x < y) and (a > b)	F	F	T

Example 1

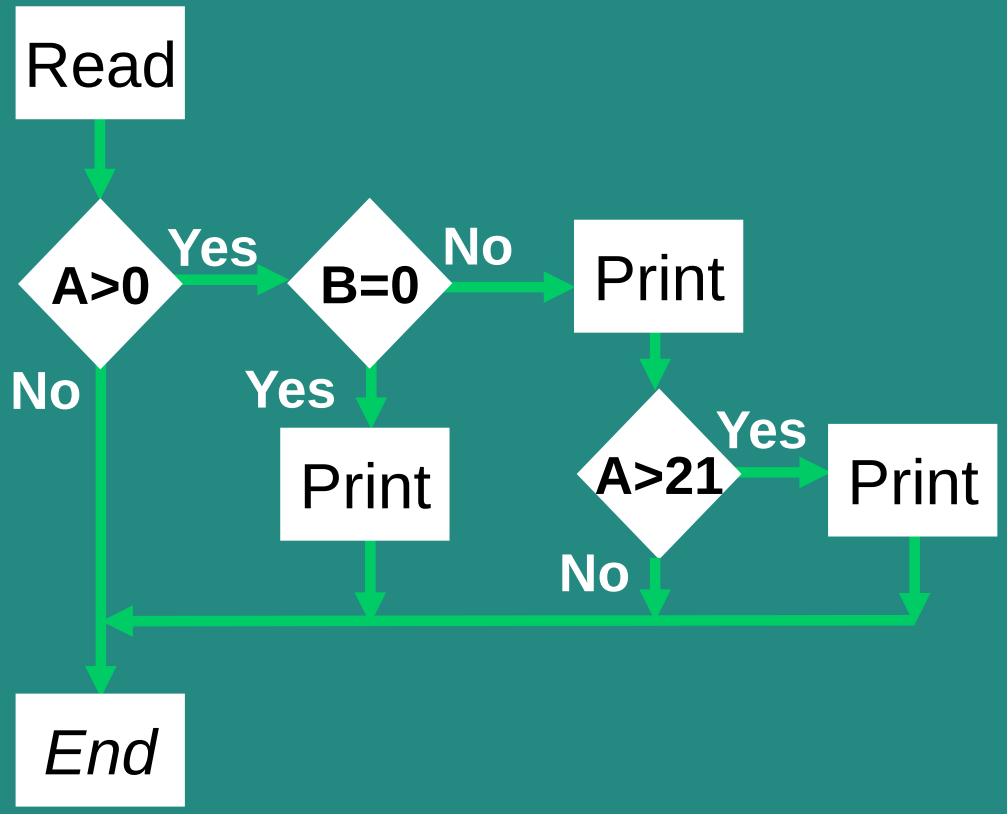
1. Read A
2. IF A > 0 THEN
3. IF A = 21 THEN
4. Print “Key”
5. ENDIF
6. ENDIF



- **Statements: 6**
- **Paths: 3**
- **Conditions: 0**

Example 2

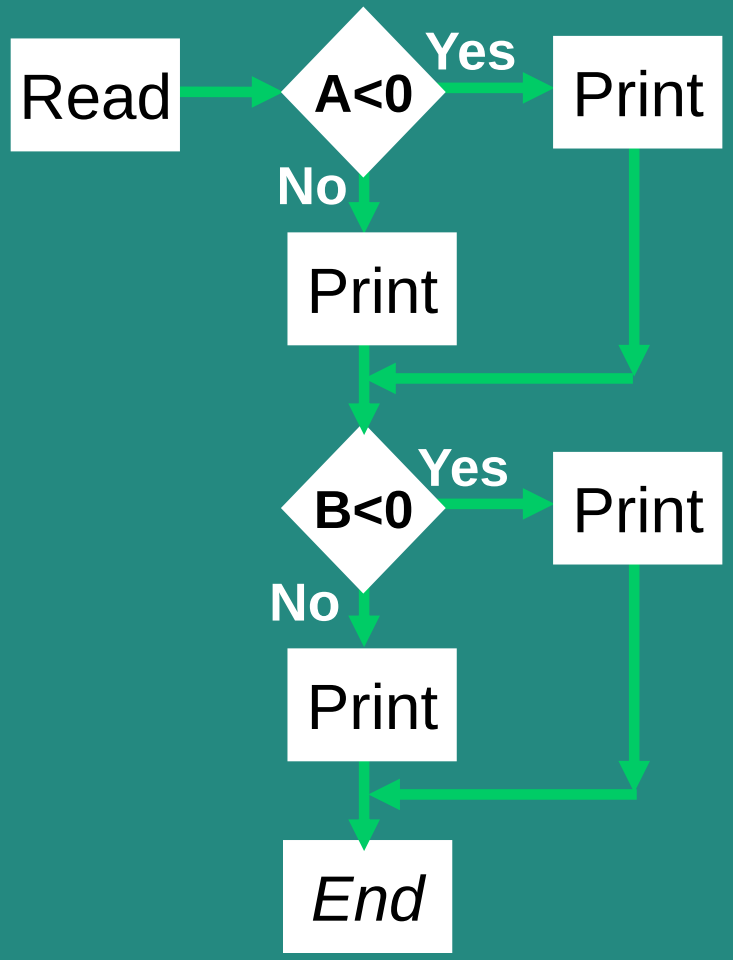
1. Read A
2. Read B
3. IF A > 0 THEN
4. IF B = 0 THEN
5. Print “No values”
6. ELSE
7. Print B
8. IF A > 21 THEN
9. Print A
10. ENDIF
11. ENDIF
12. ENDIF



- **Statements: 12**
- **Paths: ?**
- **Conditions: 0**

Example 3

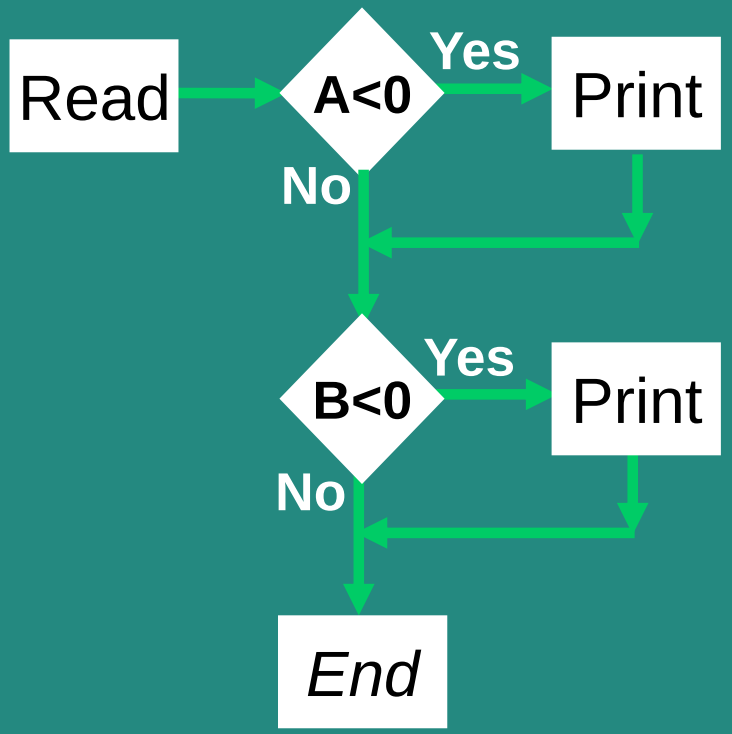
1. Read A
2. Read B
3. IF A < 0 THEN
4. Print “A negative”
5. ELSE
6. Print “A positive”
7. ENDIF
8. IF B < 0 THEN
9. Print “B negative”
10. ELSE
11. Print “B positive”
12. ENDIF



- **Statements: 12**
- **Paths: ?** ?
- **Conditions: 0**

Example 4

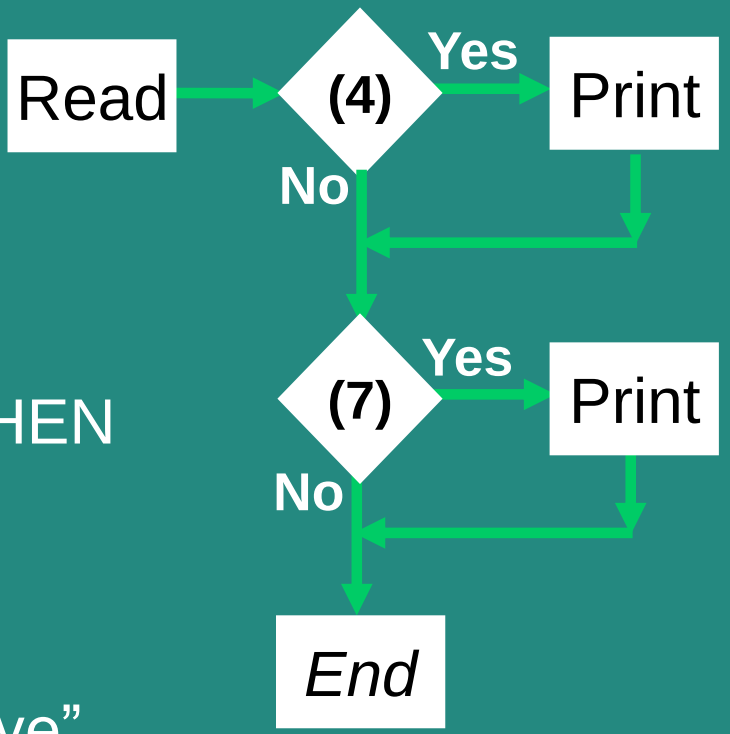
1. Read A
2. Read B
3. IF A < 0 THEN
4. Print “A negative”
5. ENDIF
6. IF B < 0 THEN
7. Print “B negative”
8. ENDIF



- **Statements: 8**
- **Paths: ?**
- **Conditions: 0**

Example 5

1. Read A
2. Read B
3. Read C
4. IF A < 0 and B<0 and C<0 THEN
5. Print “All negative”
6. ENDIF
7. IF A+B+C < 0 THEN
8. Print “At least one negative”
9. ENDIF



- **Statements: 9**
- **Paths: ?**
- **Conditions: ?**

Testcases (1)

```
1.  Read A
2.  Read B
3.  IF A > 0 AND B < 0 THEN
4.      IF A+B = 0 THEN
5.          Print “Opposite numbers”
6.      ELSE
7.          Print “Unknown numbers”
8.      ENDIF
9.  ENDIF
```

No	Description	Test Data	Expected Output	Note
Statements				
1	Bao phủ tất cả các dòng code ngoại trừ dòng 6 và 7	A=5, B=-5	Opposite numbers	
2	Bao phủ dòng 6 và 7	A=5, B=-6	Unknown numbers	
Paths				
3	Path1: 12**3(F)**9	A = -10, B = 10	Nothing to output	
4	Path2: 12**3(T)4(T)**589	A=15, B=-15	Opposite numbers	Trùng TC1, chọn data khác
5	Path3: 12**3(T)4(F)**6789	A=15, B=-20	Unknown numbers	Trùng TC2, chọn data khác

Testcases (2)

Decision Table	(1)	(2)	(3)
A>0	F	T	T
B<0	-	F	T
A>0 AND B<0	**F**	**F**	**T**
	TC6	TC7	TC8

```
1.  Read A
2.  Read B
3.  IF A > 0 AND B < 0 THEN
4.      IF A+B = 0 THEN
5.          Print “opposite numbers”
6.      ELSE
7.          Print “unknown numbers”
8.      ENDIF
9.  ENDIF
```

No	Description	Test Data	Expected Output	Note
Statements				
1	Bao phủ tất cả các dòng code ngoại trừ dòng 6 và 7	A=5, B=-5	Opposite numbers	
2	Bao phủ dòng 6 và 7	A=5, B=-6	Unknown numbers	
Paths				
3	Path1: 12**3(F)**9	A = -10, B = 10	Nothing to output	
4	Path2: 12**3(T)4(T)**589	A=15, B=-15	Opposite numbers	Trùng TC1, chọn data khác
5	Path3: 12**3(T)4(F)**6789	A=15, B=-20	Unknown numbers	Trùng TC2, chọn data khác
Conditions				
6	A>0: **F,** 12**3(F)**9	A = -10 B bất kỳ	Nothing to output	Cột (1) bảng quyết định, trùng với Path1
7	A>0: T và B<0: F, 12**3(F)**9	A = 10 B = 10	Nothing to output	Cột (2) bảng quyết định, trùng với Path1
8	A>0: T và B<0: T: 12**3(T)4(T)**589	A=15, B=-15	Opposite numbers	Cột (3) bảng quyết định, trùng với Path2

Exercise (nhóm)

1. Viết white box testcase cho hàm **DaysInMonth** sao cho bao phủ 100% điều kiện
2. Đề xuất 1 hàm tự chọn có biểu thức điều kiện gồm tối thiểu 3 toán hạng. Viết Unit TCs bao phủ 100% statements, paths và conditions cho hàm này
3. Mẫu testcase tham khảo slide 15

Exercise

```
boolean IsValidateDate(int year, int month, int day){
    if (month>=1 && month<=12){
        if (day>=1 && day<=DaysInMonth(year, month)){
            return true;
        }else{
            return false;
        }
    }else{
        return false;
    }
}
int DaysInMonth(int year, int month){
    if (month==1 || month==3 || month==5 || month==7 || month==8 || month==10 || month==12){
        return 31;
    }else if (month==4 || month==6 || month==9 || month==11){
        return 30;
    }else if (month == 2 && (year%400==0 || (year%4==0 && year%100!=0))){
        return 29;
    }else{
        return 28;
    }
}
```