Universiti Malaysia PAHANG
Engineering • Technology • Creativity

BCS3323 – Software Testing and Maintenance

Test Case Design
White Box

Editors
Dr. AbdulRahman A. Alsewari
Faculty of Computer Systems & Software Engineering
alswari@ump.edu.my

White BOX testing

Aims is to discover

- The purpose of White Box testing
- The purpose of White Box testing techniques
- How to use these techniques to design the test cases

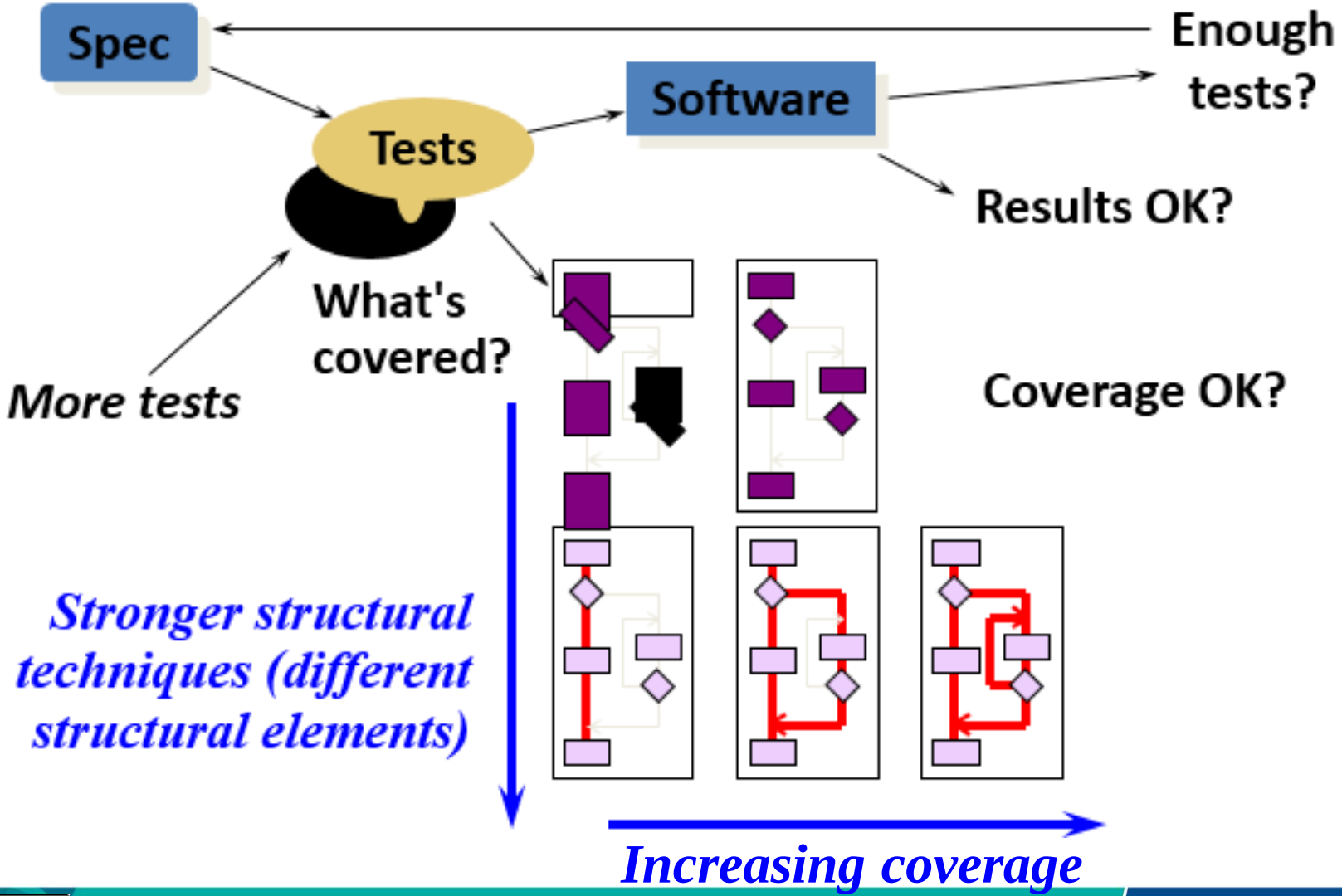
Expected Outcomes

- Students be able to show how to design the test cases based on these techniques
- Students be able to show when to use these techniques .

References

- ISTQB
- MSTB/GTB
- http://www.softwaretestingclass.com/software-testing-tools-list/
- http://www.softwaretestinggenius.com/articalDetails.php?qry=572#commentsList

Using structural coverage



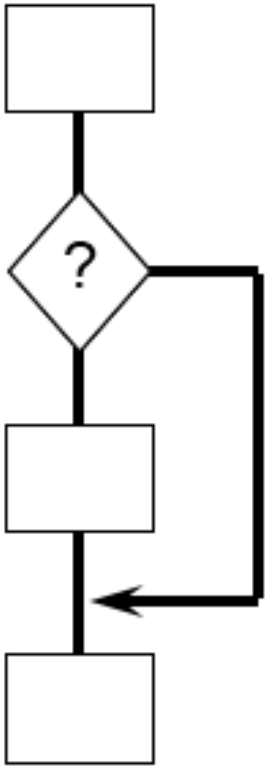
Statement coverage

Statement coverage is normally measured by a software tool.

- percentage of executable statements exercised by a test suite

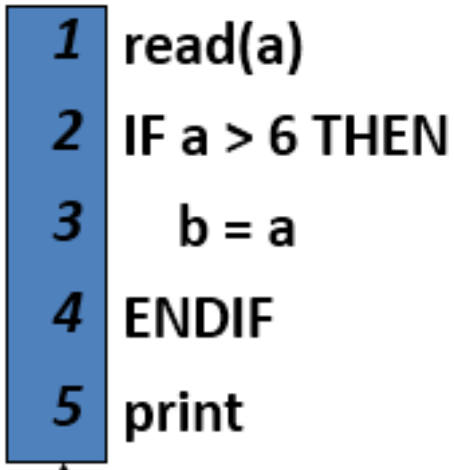
$$= \frac{\text{number of statements exercised}}{\text{total number of statements}}$$

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



Typical ad hoc testing achieves 60 - 75%

Example of statement coverage



Test case	Input	Expected output
1	7	7

Statement numbers

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

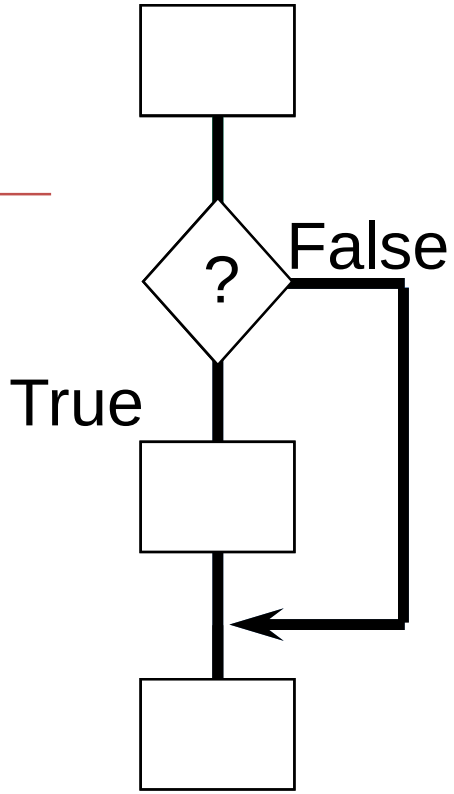
Decision coverage (Branch coverage)

Decision coverage is normally measured by a software tool.

- percentage of decision outcomes exercised by a test suite

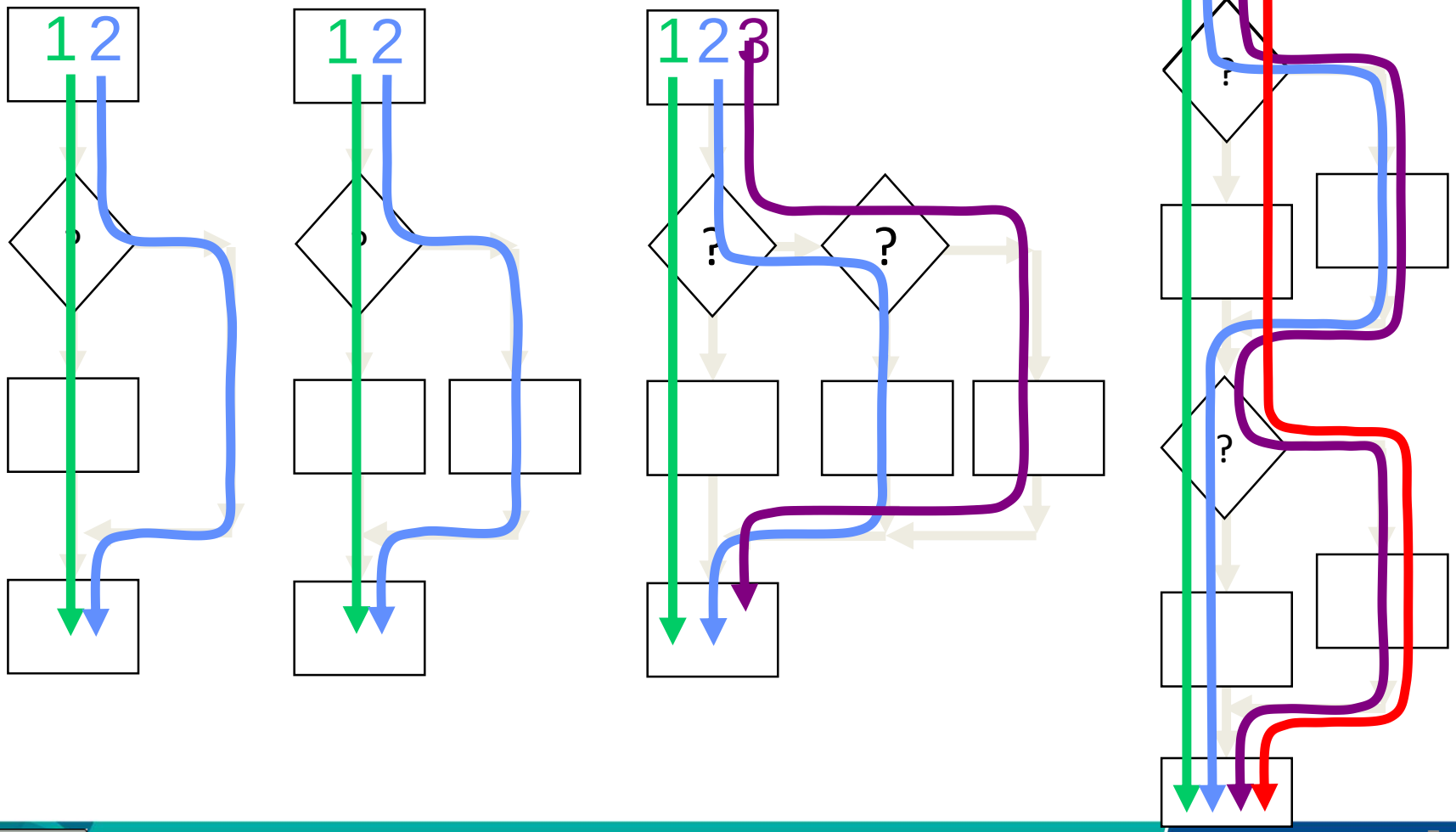
$$= \frac{\text{number of decisions outcomes exercised}}{\text{total number of decision outcomes}}$$

- example:
  - program has 120 decision outcomes
  - tests exercise 60 decision outcomes
  - decision coverage = 50%

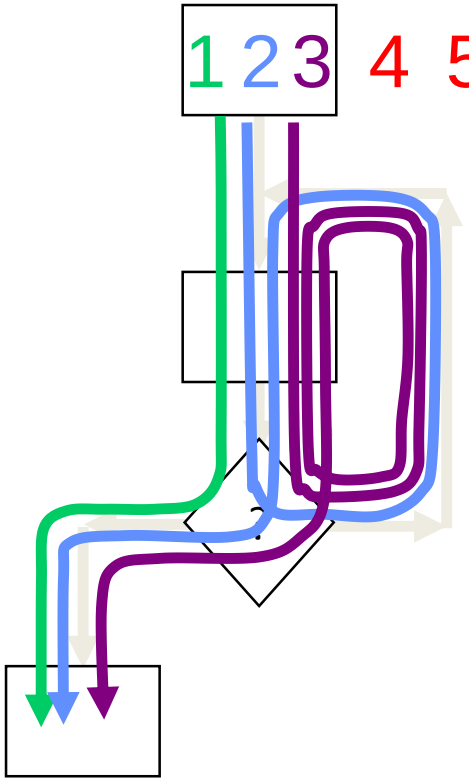


Typical ad hoc testing achieves 40 - 60%

Paths through code



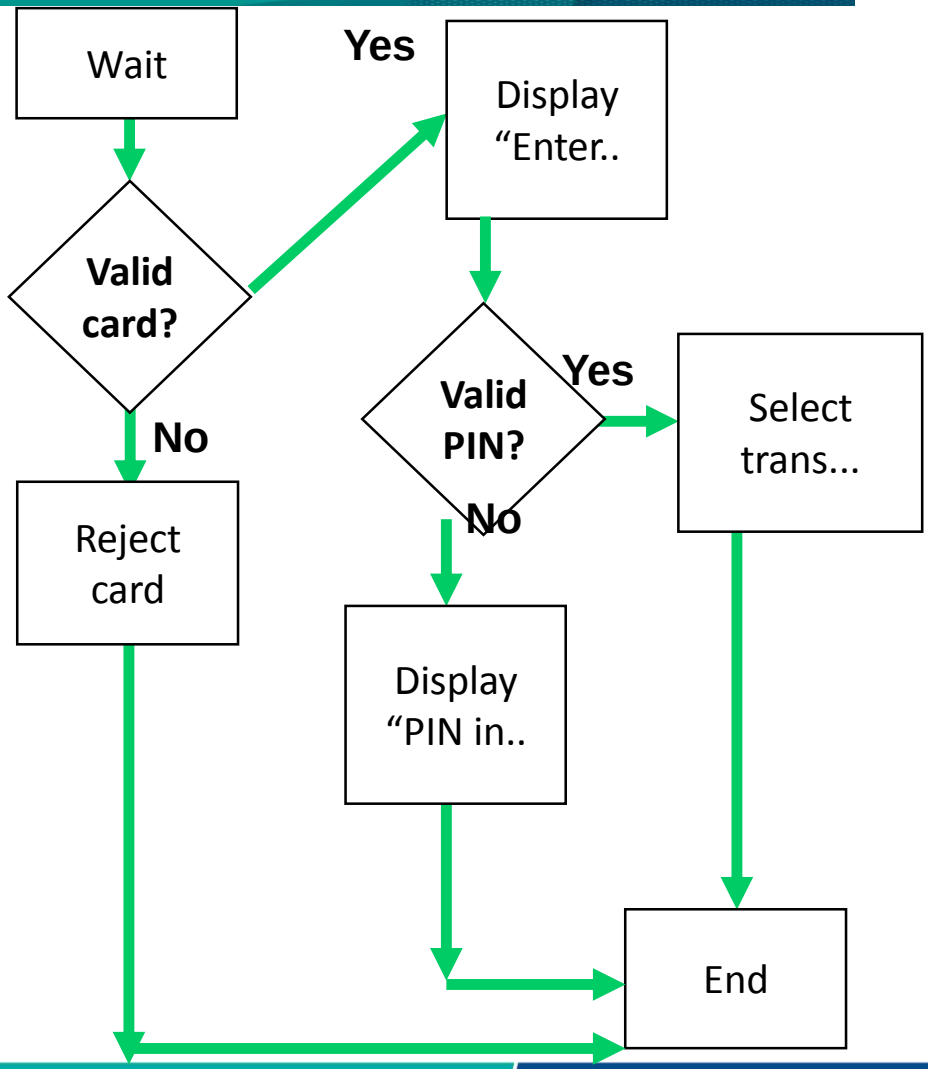
Paths through code with loops



for as many times as it is possible to go round the loop (this can be unlimited, i.e. infinite)

Example 1

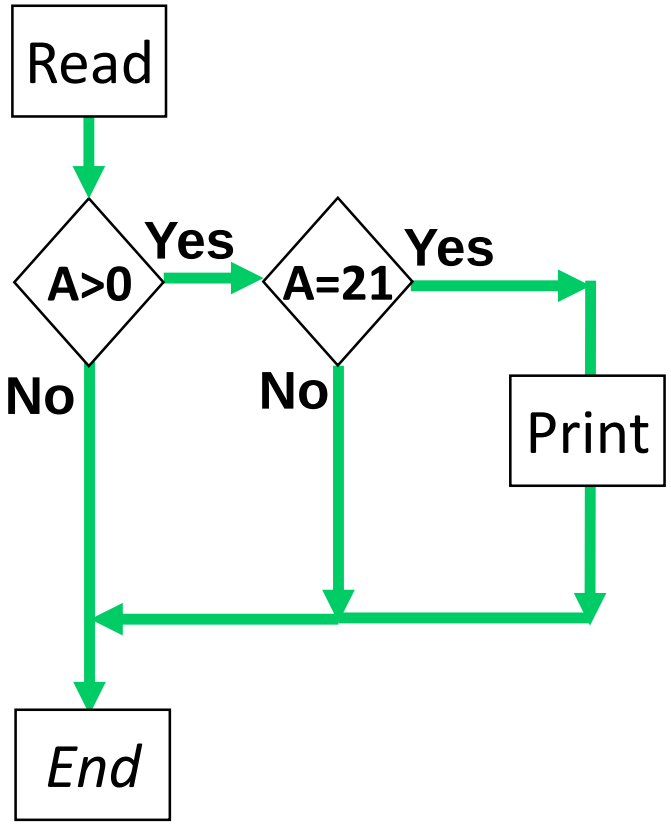
Wait for card to be inserted
IF card is a valid card THEN
    display “Enter PIN number”
    IF PIN is valid THEN
        select transaction
    ELSE (otherwise)
        display “PIN invalid”
ELSE (otherwise)
    reject card



Example 2

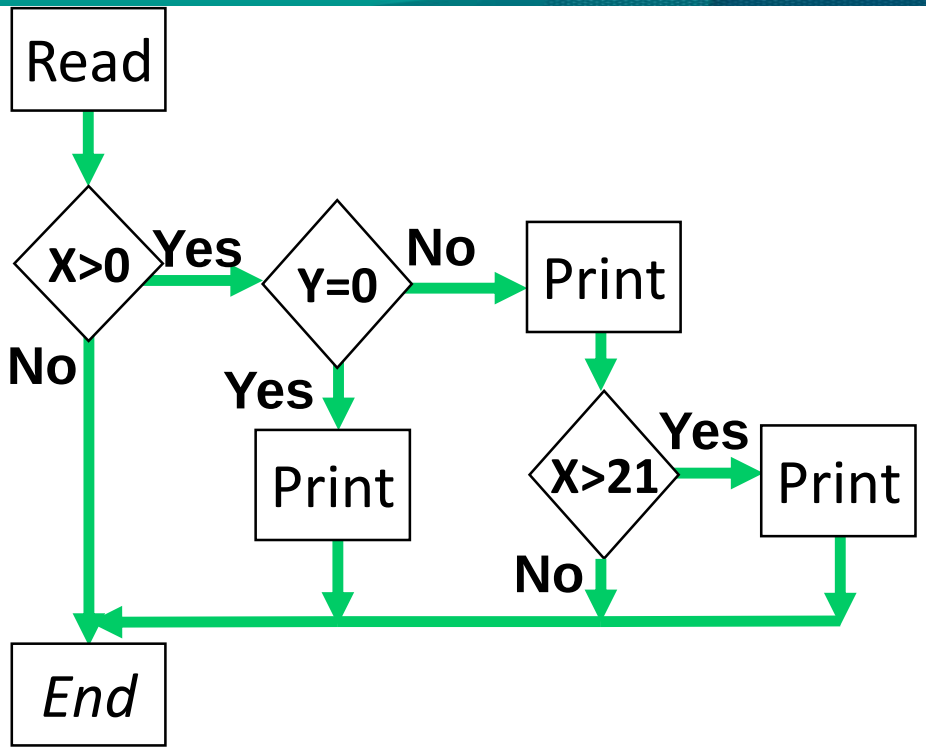
```
Read A
IF A > 0 THEN
    IF A  = 21 THEN
     Print “Key”
    ENDIF
ENDIF
```

- Cyclomatic complexity: _____
- Minimum tests to achieve:
  - Statement coverage: _____
  - Branch coverage: _____



Example 3

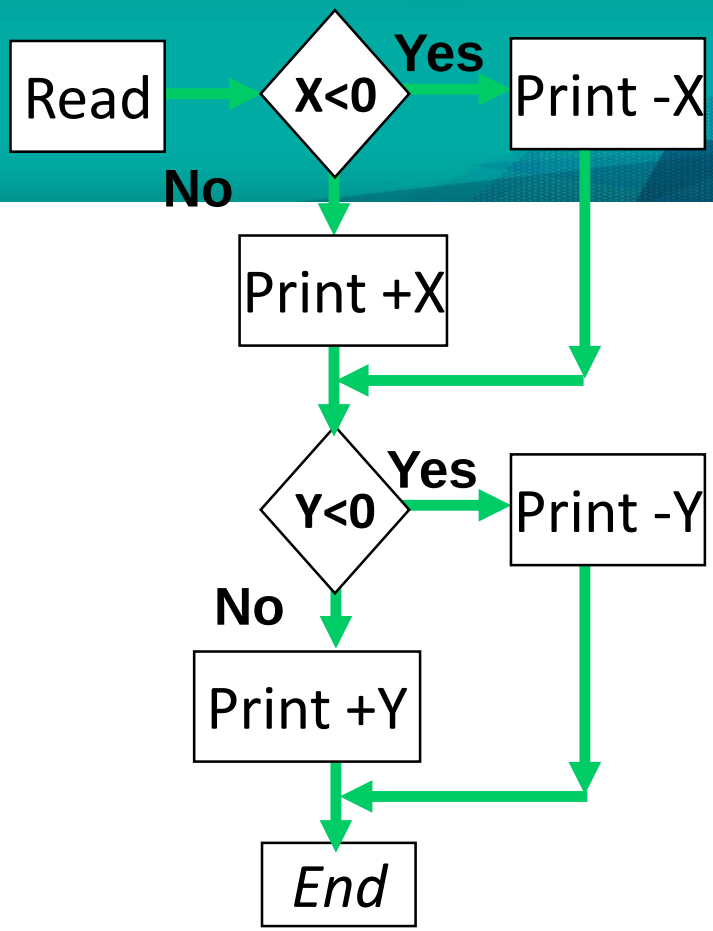
```
Read X
Read Y
IF X > 0 THEN
     IF Y  = 0 THEN
      Print “No values”
     ELSE
      Print Y
      IF X > 21 THEN
          Print X
      ENDIF
     ENDIF
ENDIF
```



- Cyclomatic complexity: _____
- Minimum tests to achieve:
  - Statement coverage: _____
  - Branch coverage: _____

Example 4

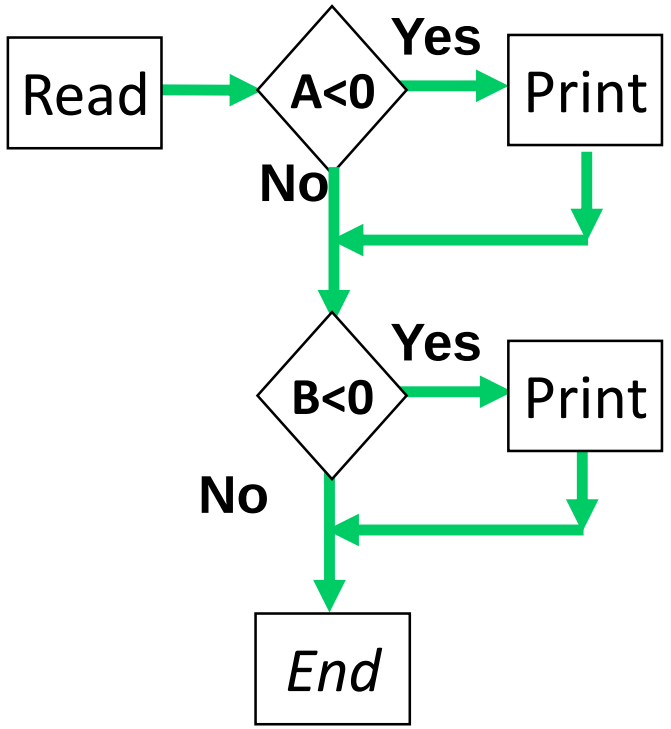
```
Read X
Read Y
IF X < 0 THEN
    Print  “X negative”
ELSE
    Print  “X positive”
ENDIF
IF Y < 0 THEN
    Print  “Y negative”
ELSE
    Print  “Y positive”
ENDIF
```



- Cyclomatic complexity: _____
- Minimum tests to achieve:
  - Statement coverage: _____
  - Branch coverage: _____

Example 5

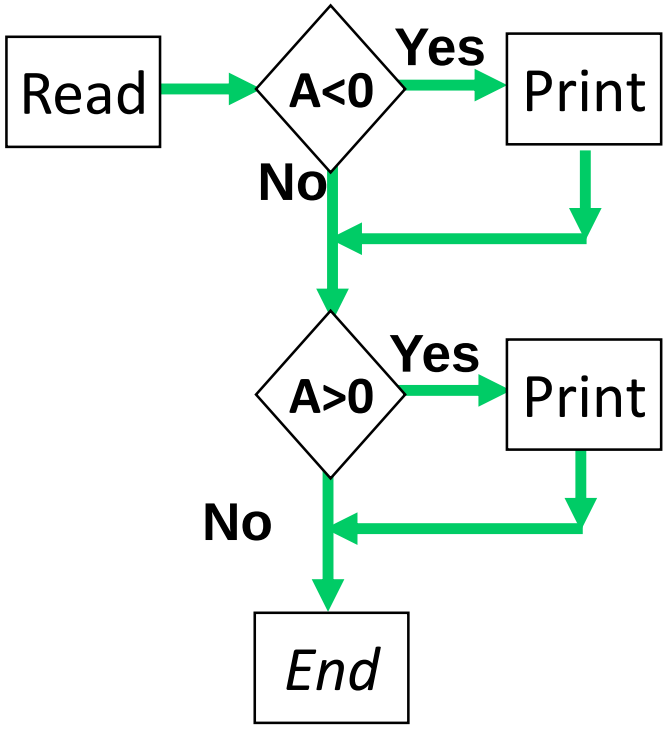
```
Read A
Read B
IF A < 0 THEN
    Print  “A negative”
ENDIF
IF B < 0 THEN
    Print  “B negative”
ENDIF
```



- Cyclomatic complexity: _____
- Minimum tests to achieve:
  - Statement coverage: _____
  - Branch coverage: _____

Example 6

```
Read A
IF A < 0 THEN
    Print  "A negative"
ENDIF
IF A > 0 THEN
    Print  "A positive"
ENDIF
```



- Cyclomatic complexity: _____
- Minimum tests to achieve:
  - Statement coverage: _____
  - Branch coverage: _____