Pseudocode & Algorithm

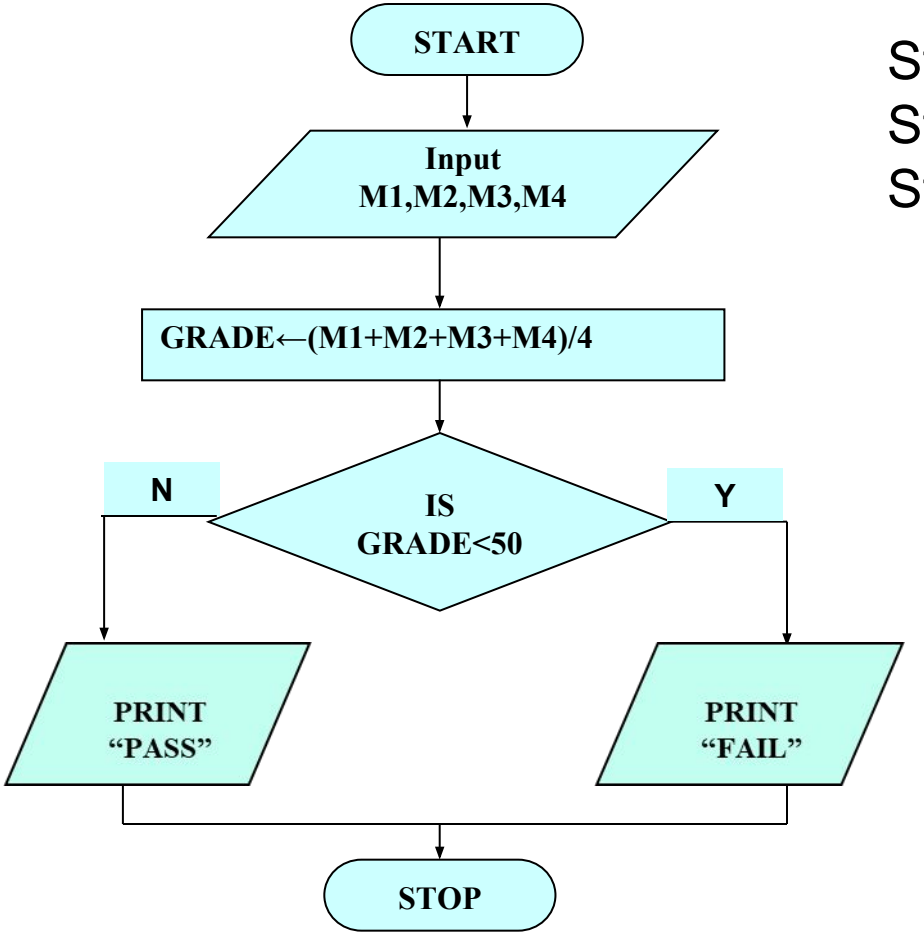
- **Example 1:** Write an algorithm to determine a student's final grade and indicate whether it is passing or failing. The final grade is calculated as the average of four marks.

Pseudocode & Algorithm

- Detailed Algorithm

```
Step 1:    Input M1,M2,M3,M4
    Step 2:    GRADE ← (M1+M2+M3+M4)/4
    Step 3:    if (GRADE < 60) then
                   Print “FAIL”
               else
                   Print “PASS”
               endif
```

Example 1



Step 1: Input M1,M2,M3,M4
Step 2: GRADE ← (M1+M2+M3+M4)/4
Step 3: if (GRADE <50) then
Print “FAIL”
else
Print “PASS”
endif

Example 2

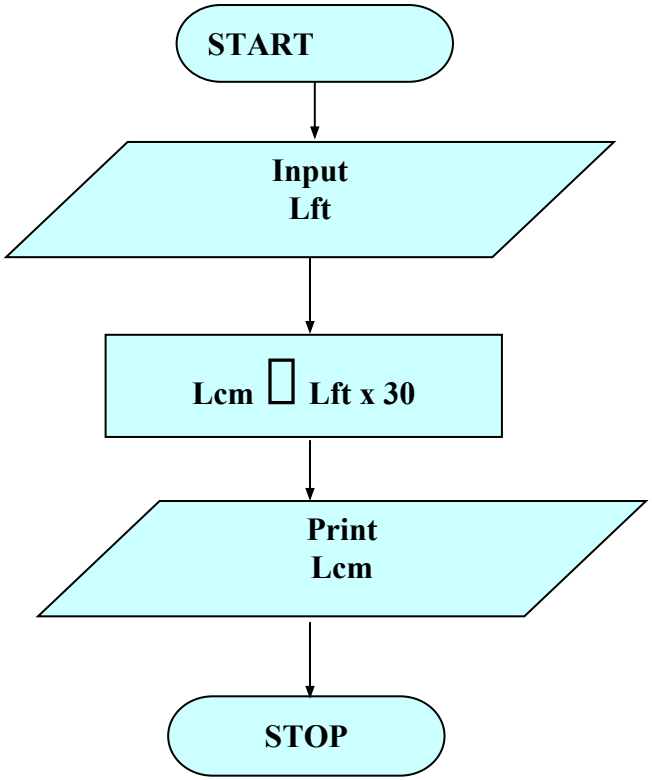
- Write an algorithm and draw a flowchart to convert the length in feet to centimeter.

Example 2

Algorithm

- Step 1: Input Lft
- Step 2: Lcm Lft x 30
- Step 3: Print Lcm

Flowchart



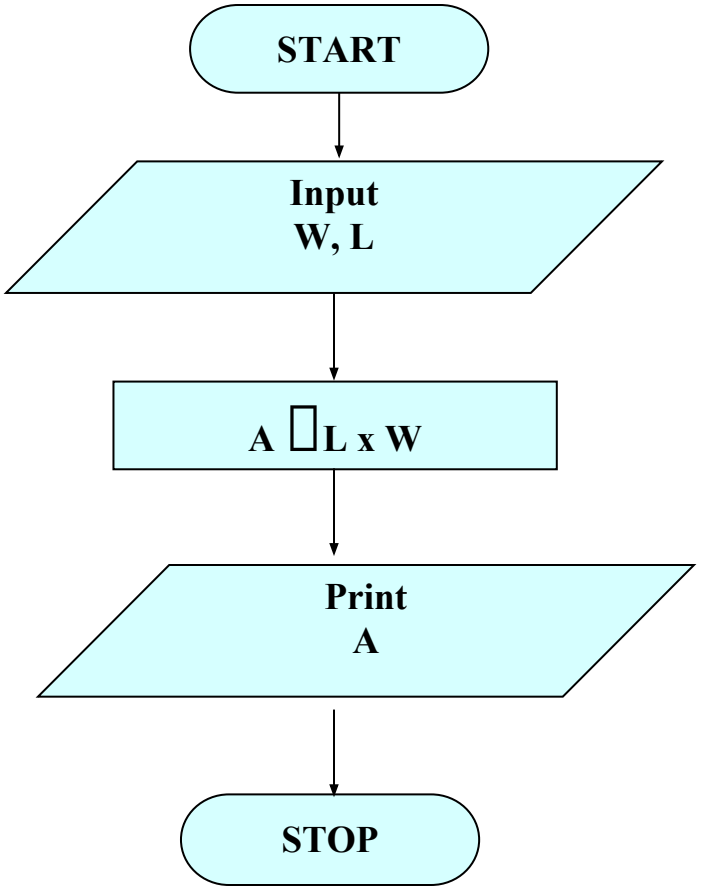
Example 3

Write an algorithm and draw a flowchart that will read the two sides of a rectangle and calculate its area.

Example 3

Algorithm

- Step 1: Input W,L
- Step 2: A ← L x W
- Step 3: Print A



Example 6

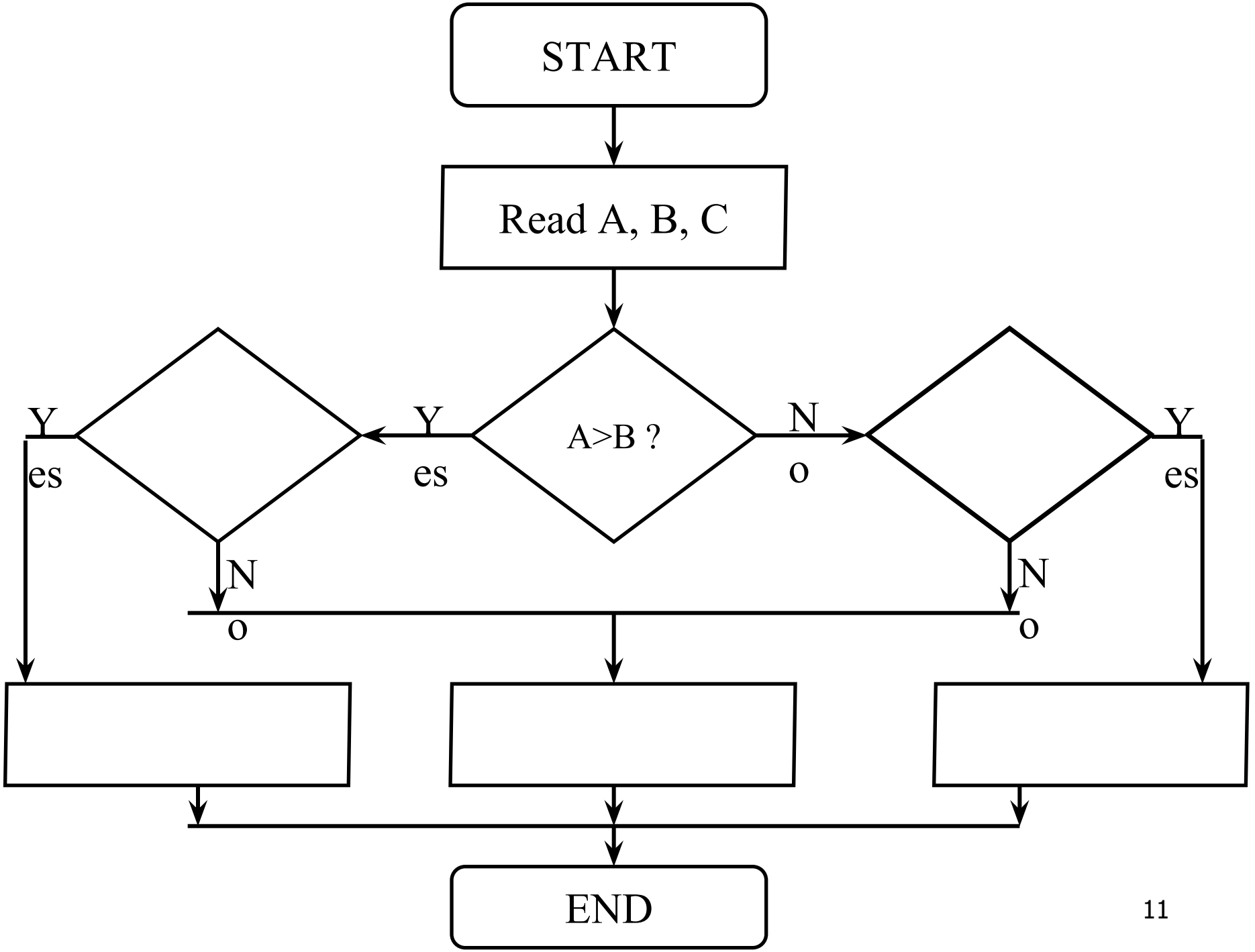
- We want to create a flowchart that prints out the biggest of three inputted numbers

Example 6

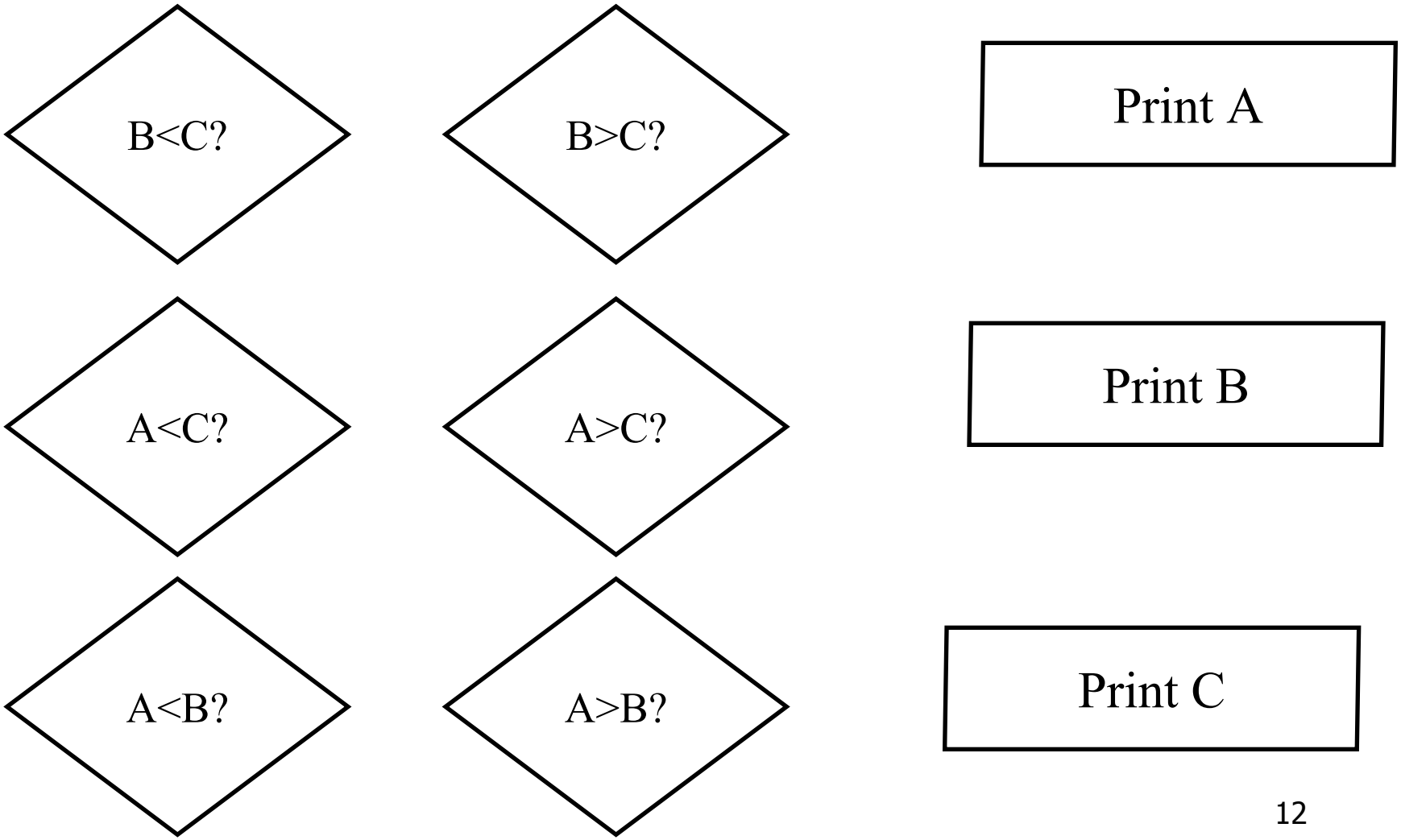
Step 1: ***Input*** **A, B, C**
Step 2: ***if (*****A>B)** ***then***
if (**A>C)** ***then***
MAX A [A>B, A>C]
else
MAX C [C>A>B]
endif
else
if (**B>C)** ***then***
MAX B [B>A, B>C]
else
MAX C [C>B>A]
endif
endif
Step 3: ***Print "The largest number is", MAX***

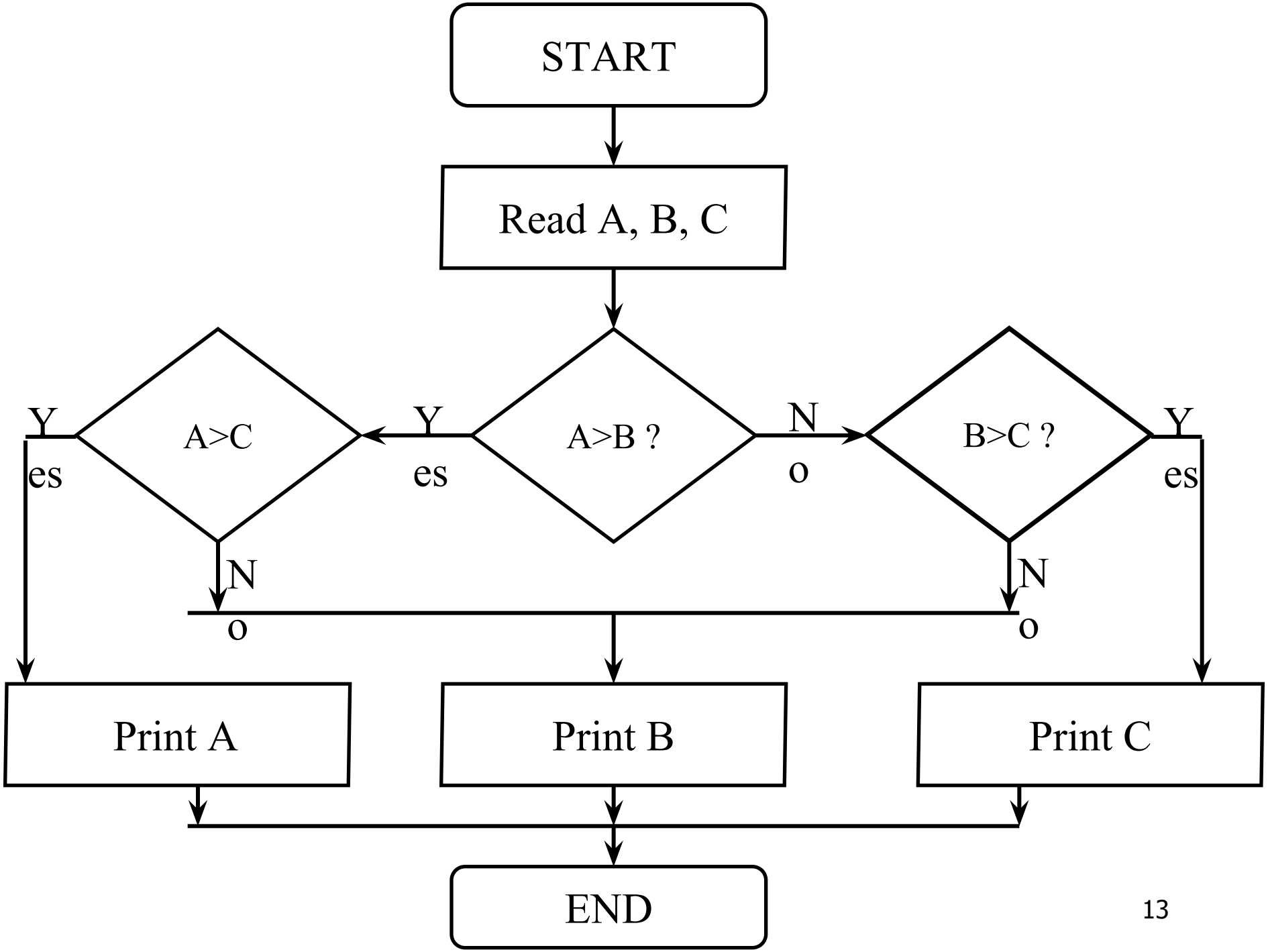
Example 6

- **Flowchart: Draw the flowchart of the above Algorithm.**
- On the following slide, a number of potential boxes you could use to correctly implement the algorithm.


START
Read A, B, C
A>B ?
Y
es
N
o
Y
es
N
o
Y
es
N
o
END

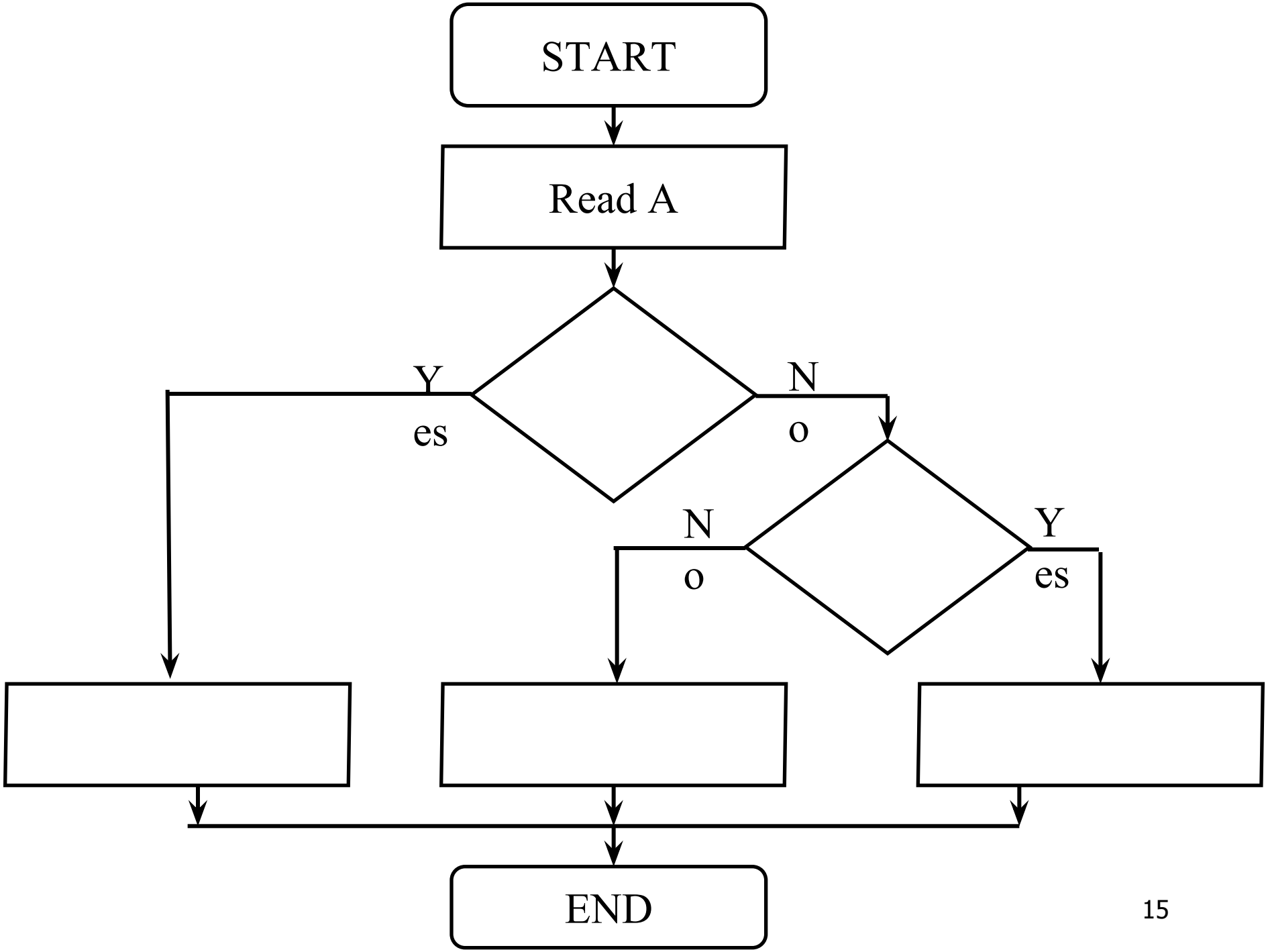
- *Pick the appropriate three of the following boxes that describe the algorithm as described.*



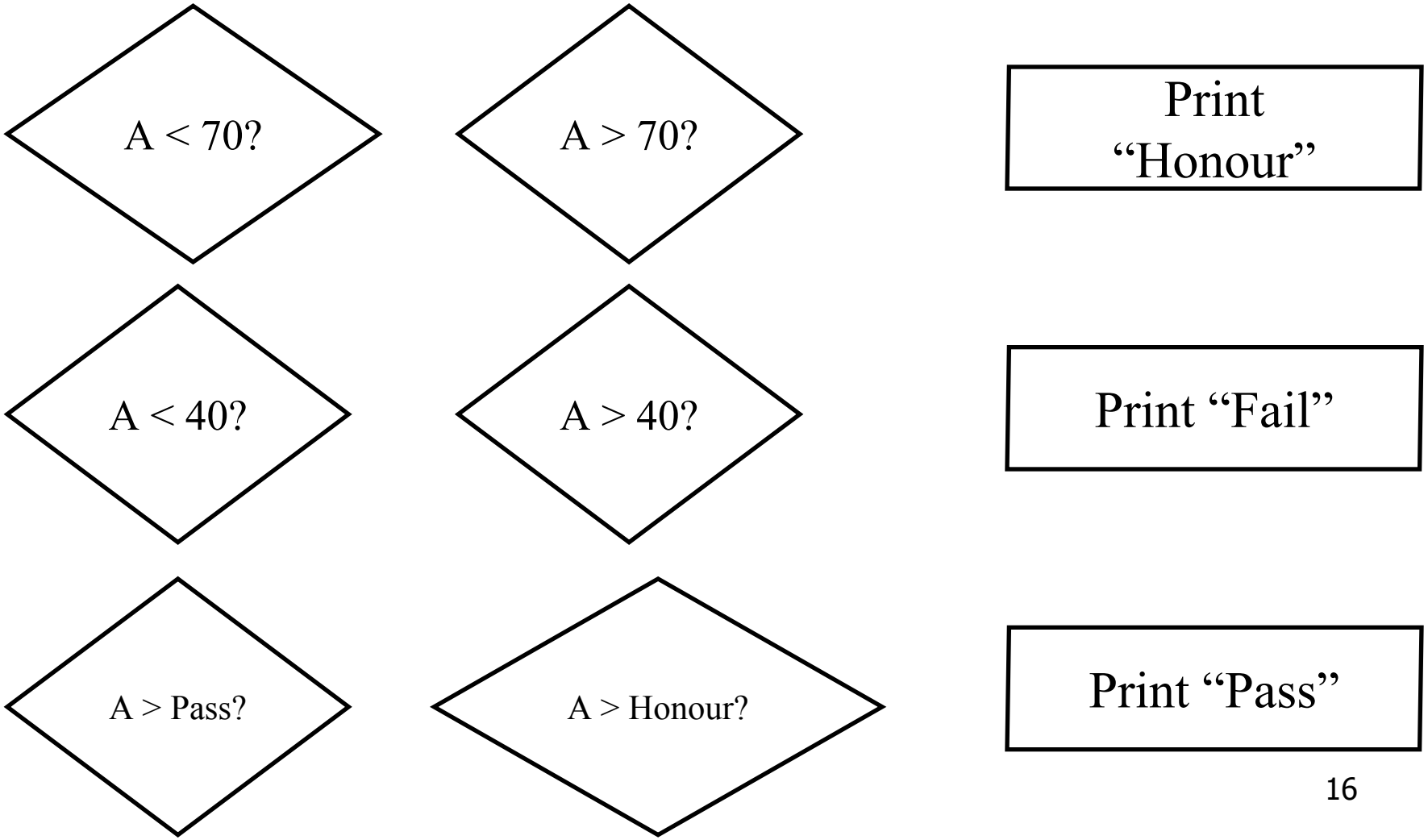

START
Read A, B, C
A>B ?
Y
es
A>C
Y
es
N
o
B>C ?
Y
es
N
o
N
o
Print A
Print B
Print C
END

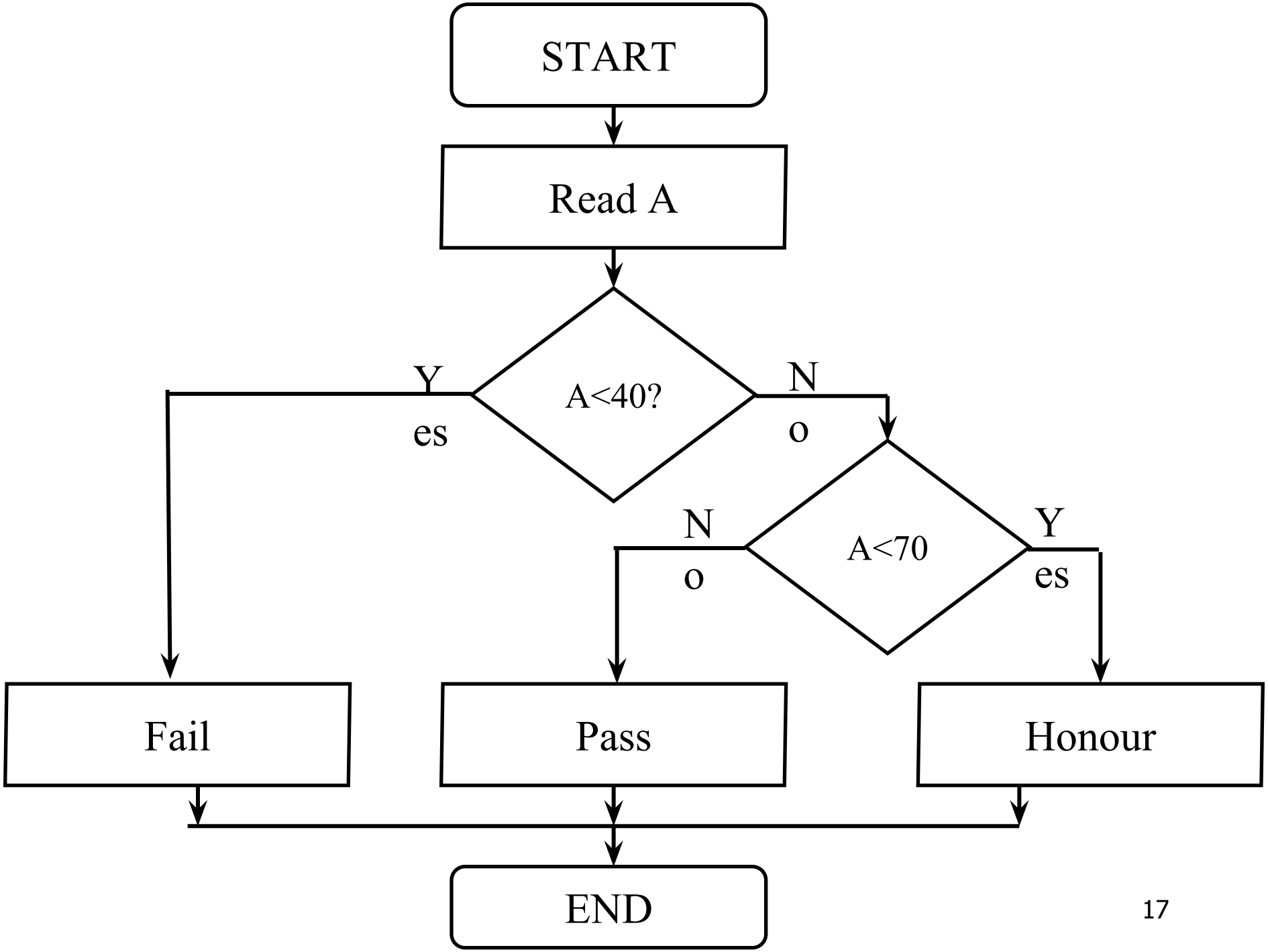
Example 7

- We want to create a flowchart that prints out the word “Honour” if the number input is 70, if the number is less than 40 print out the word “Fail”, otherwise print out the word “Pass”.
- On the following slide, a number of potential boxes you could use to correctly implement the algorithm.


START
Read A
Y
es
N
o
N
o
Y
es
END

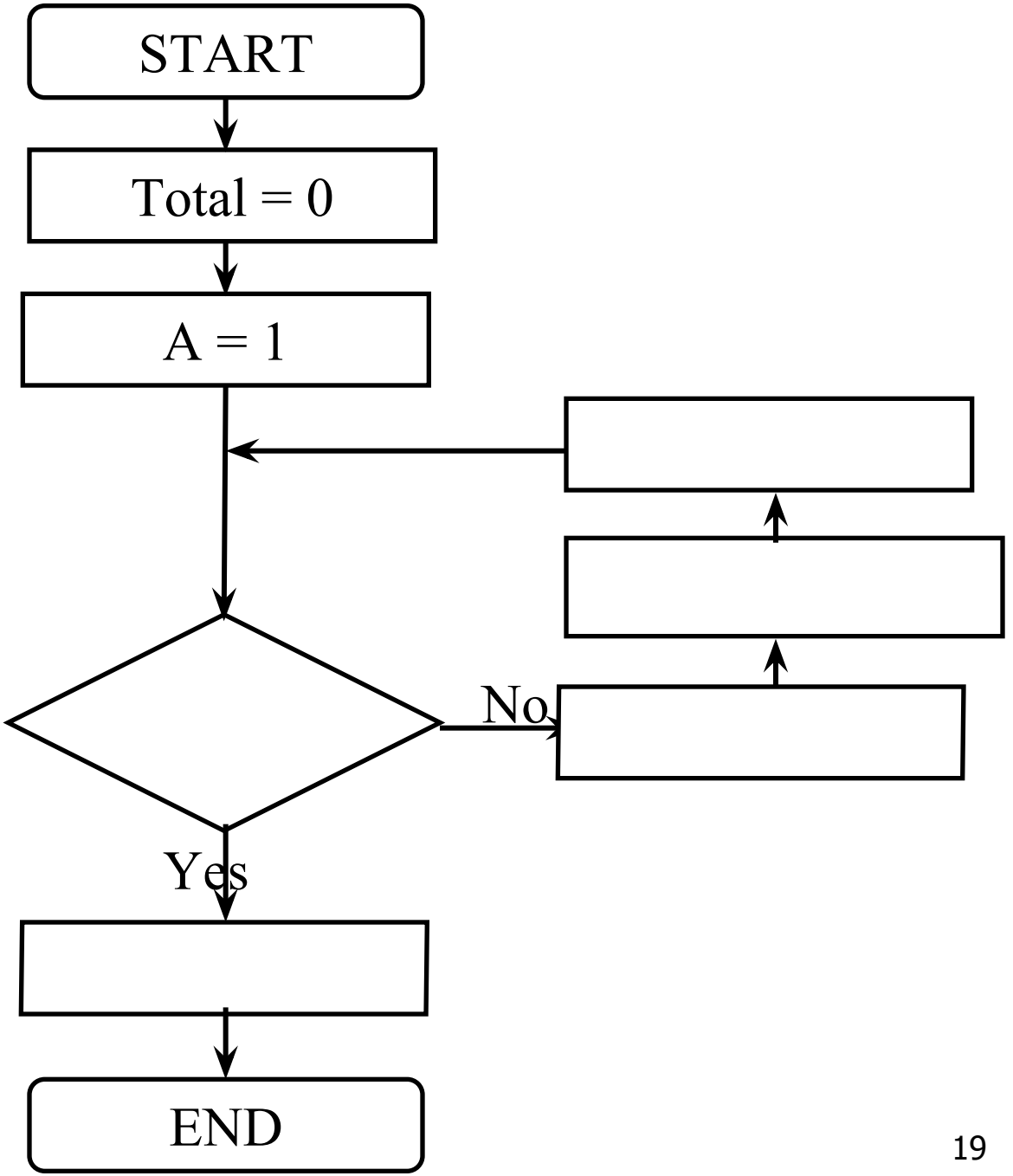
- *Pick the appropriate three of the following boxes that describe the algorithm as described*



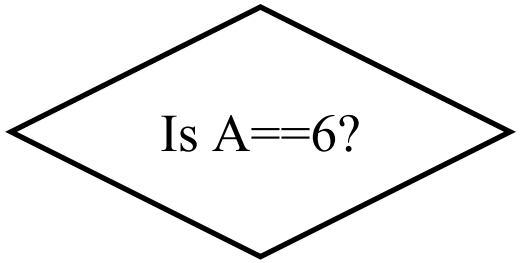

START
Read A
A<40?
Y
es
N
o
A<70
N
o
Y
es
Fail
Pass
Honour
END

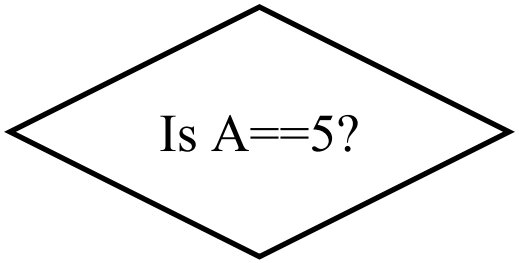
Example 8

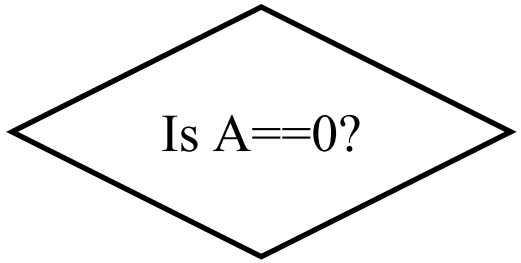
- We want to create a flowchart that prints out the average value of five numbers input in.
- On the following slide, a number of potential boxes you could use to correctly implement the algorithm.


START
Total = 0
A = 1
No
Yes
END

- *Pick the appropriate three of the following boxes that describe the algorithm as described.*







Print Total

Print Total / 5

Print Total++

Total = X

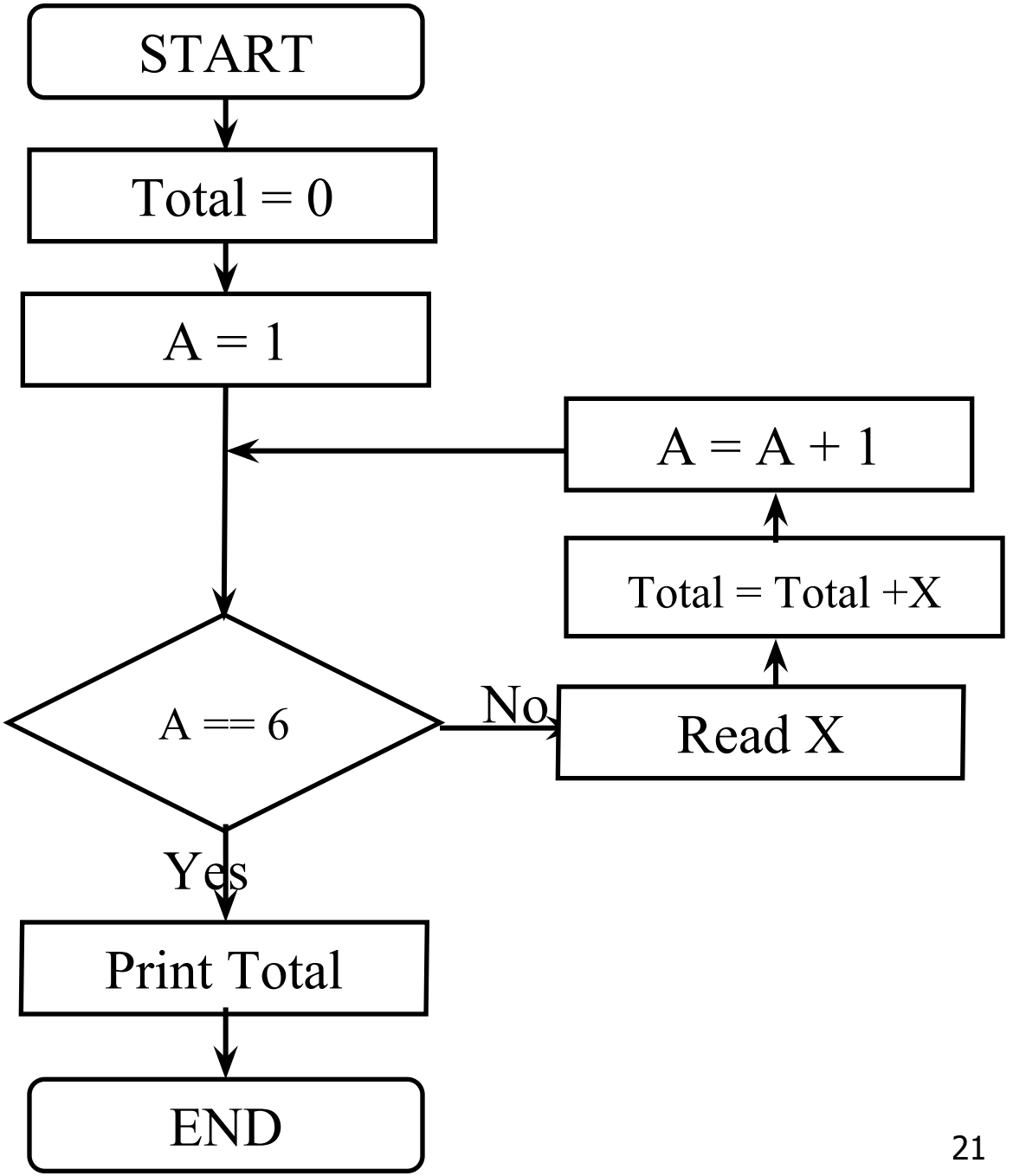
Total = Total

Total = Total + X

Read A

Read Total

Read X


START
Total = 0
A = 1
A = A + 1
Total = Total +X
A == 6
No
Read X
Yes
Print Total
END

Example 9

- Write an algorithm and draw a flowchart that will calculate the roots of a quadratic equation

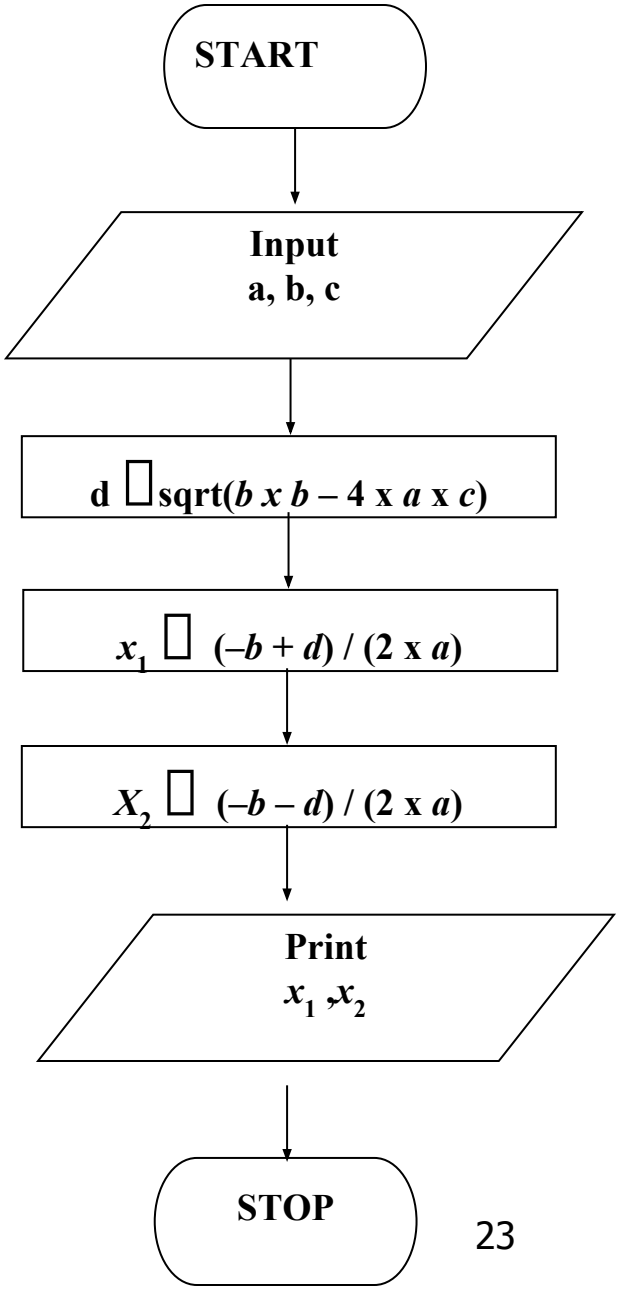
  $ax^2 + bx + c = 0$

- Hint: **d** = sqrt ($b^2 - 4ac$),
- and the roots are:
- If d<0 no solution
- else

  $\boldsymbol{x1} = (-b + d)/2a$ and $\boldsymbol{x2} = (-b - d)/2a$

Example 9

- **Algorithm**:
- Step 1: Input a, b, c
- Step 2: $d \leftarrow \text{sqrt}(b \times b - 4 \times a \times c)$
- Step 3: $x1 \leftarrow (-b + d) / (2 \times a)$
- Step 4: $x2 \leftarrow (-b - d) / (2 \times a)$
- Step 5: Print $x1, x2$

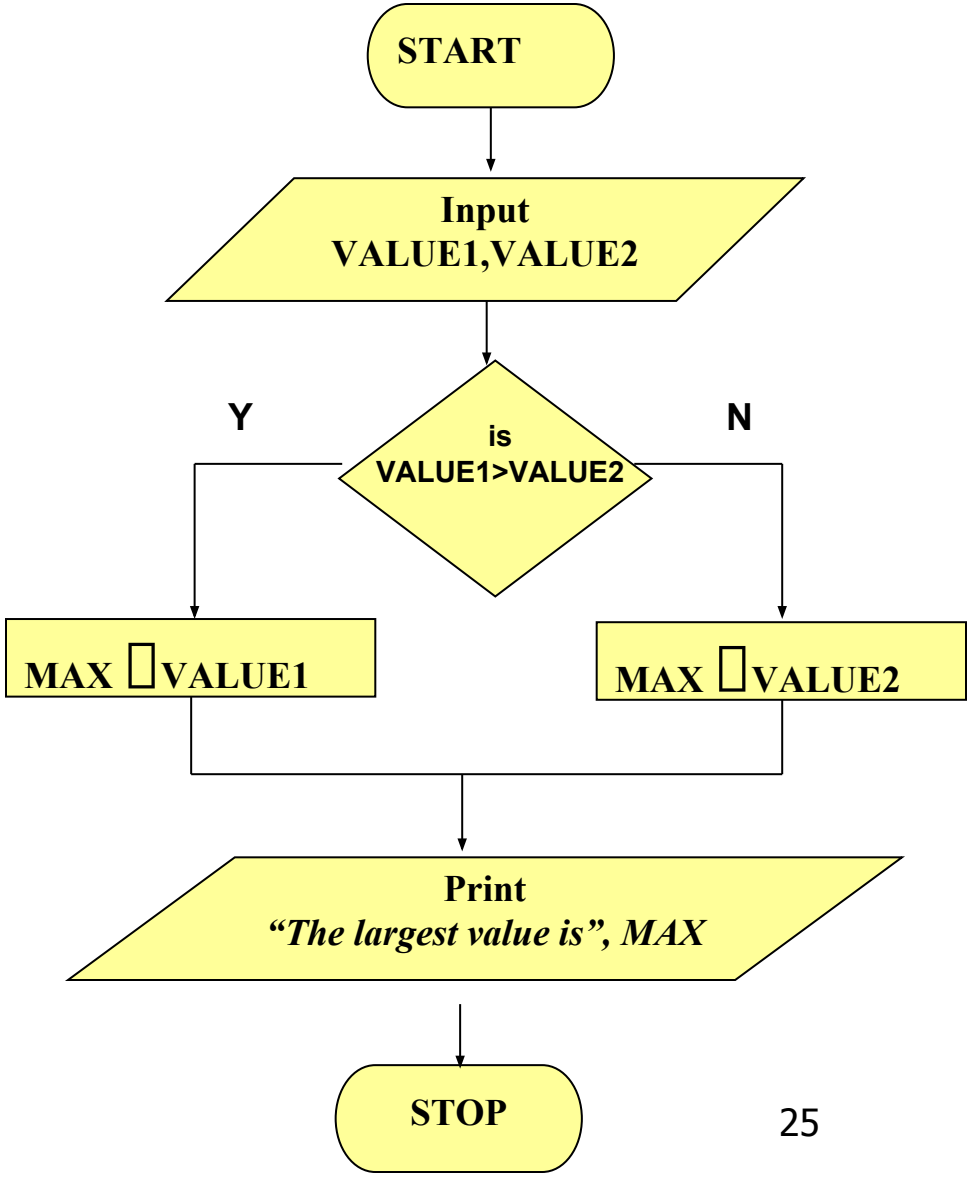


Example 10

- Write an algorithm that reads two values, determines the largest value and prints the largest value with an identifying message.

Example 10

ALGORITHM
Step 1: *Input* VALUE1, VALUE2
Step 2: *if (*VALUE1 > VALUE2)
then
MAX VALUE1
else
MAX VALUE2
endif
Step 3: *Print "The largest value is", MAX*



Example 11

Write and algorithm and draw a flowchart to

a) read an employee name (NAME), overtime hours worked (OVERTIME), hours absent (ABSENT) and

b) determine the bonus payment (PAYMENT).

Example 11

Bonus Schedule	
OVERTIME – (2/3)*ABSENT	Bonus Paid
>40 hours >30 but ≤ 40 hours >20 but ≤ 30 hours >10 but ≤ 20 hours ≤ 10 hours	$50 $40 $30 $20 $10

Step 1: *Input* NAME,OVERTIME,ABSENT
Step 2: *if* (OVERTIME–(2/3)*ABSENT > 40) *then*
PAYMENT 50
*else if (*OVERTIME–(2/3)*ABSENT > 30) *then*
PAYMENT 40
*else if (*OVERTIME–(2/3)*ABSENT > 20) *then*
PAYMENT 30
*else if (*OVERTIME–(2/3)*ABSENT > 10) *then*
PAYMENT 20
else
PAYMENT 10
endif
Step 3: *Print* “Bonus for”, NAME “is $”, PAYMENT

Example 11

- **Flowchart: Draw the flowchart of the above algorithm?**

Example 12

Express an algorithm to get two numbers from the user (dividend and divisor), testing to make sure that the divisor number is not zero, and displaying their quotient using a *flowchart.*

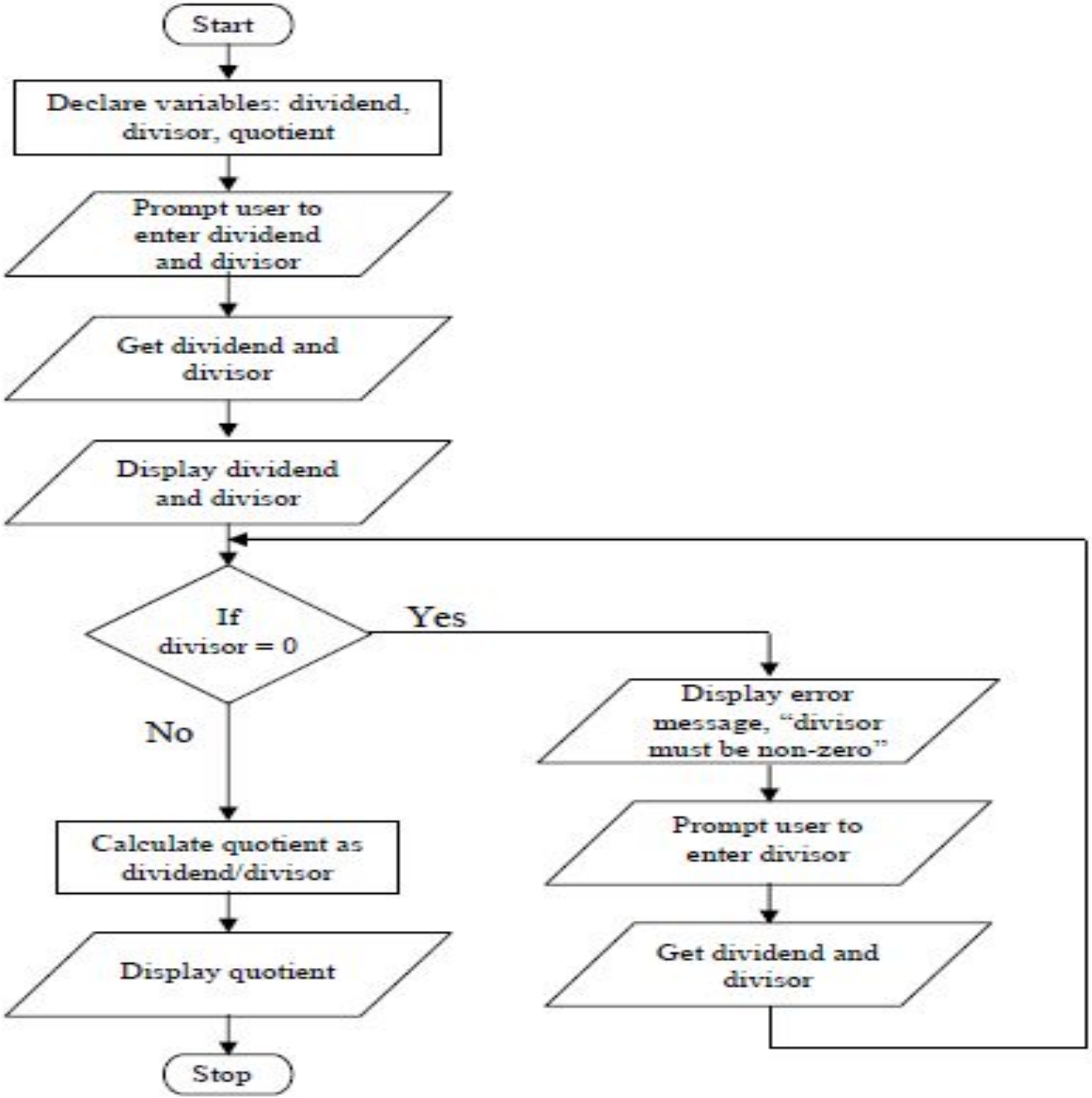
Example 12 Answer

- **Step 1** – Declare variables – dividend, divisor, quotient
- **Step 2** – Prompt user to get dividend
- **Step 3 –** Store values in dividend variable
- **Step 4** – Prompt user to get divisor
- **Step 5 –** Store value in divisor variable
- **Step 6 –** Display dividend and divisor
- **Step 7 -** Loop

    Selection: If divisor is equal to zero

        Display error message, "divisor must be non-zero" and

- go back to step 4
- **Step 8 -** Calculate quotient as dividend/divisor
- **Step 9 -** Display quotient


Start
Declare variables: dividend, divisor, quotient
Prompt user to enter dividend and divisor
Get dividend and divisor
Display dividend and divisor
If divisor = 0
Yes
No
Display error message, "divisor must be non-zero"
Prompt user to enter divisor
Get dividend and divisor
Calculate quotient as dividend/divisor
Display quotient
Stop