



SNS COLLEGE OF TECHNOLOGY

Coimbatore-35.

An Autonomous Institution

COURSE NAME : 19CST101 PROGRAMMING FOR PROBLEM SOLVING

I YEAR/ I SEMESTER

UNIT-I INTRODUCTION TO PROBLEM SOLVING TECHNIQUES

Topic: Illustrative Examples

Mrs. Devi G

Assistant Professor

Department of Computer Science and Engineering



To Find Area of A Rectangle

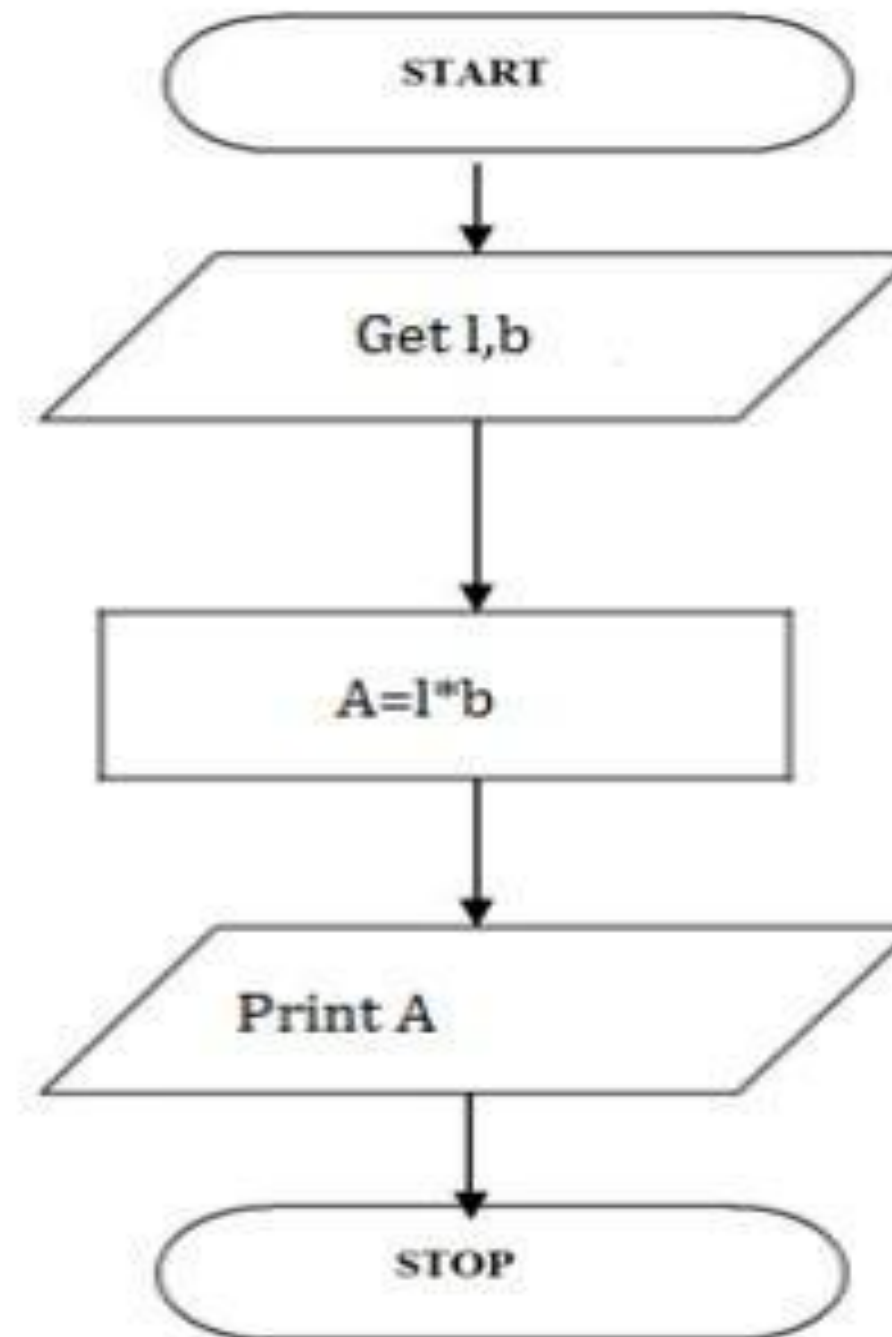


Algorithm

- Step 1: Start
- Step 2: get l,b values
- Step 3: Calculate $A=l*b$
- Step 4: Display A
- Step 5: Stop

Pseudo Code

```
BEGIN  
READ l,b  
CALCULATE  $A=l*b$   
DISPLAY A  
END
```





Calculating Area and Circumference of Circle



Algorithm

Step 1: Start

Step 2: get r value

Step 3: Calculate $A=3.14*r*r$

Step 4: Calculate $C=2*3.14*r$

Step 5: Display A,C

Step 6: Stop

Pseudo Code

BEGIN

READ r

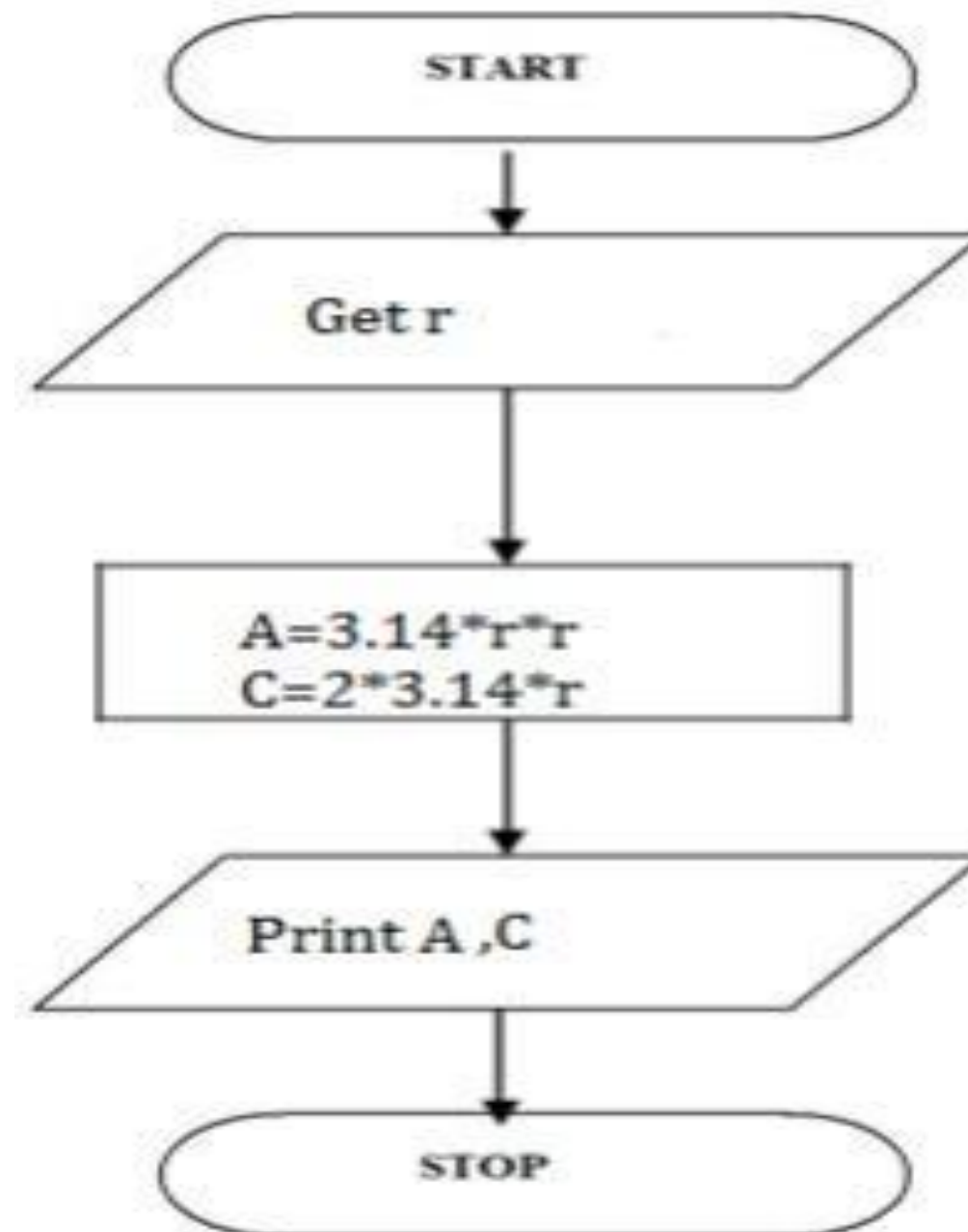
CALCULATE A and C

$A=3.14*r*r$

$C=2*3.14*r$

DISPLAY A

END





CALCULATING SIMPLE INTEREST

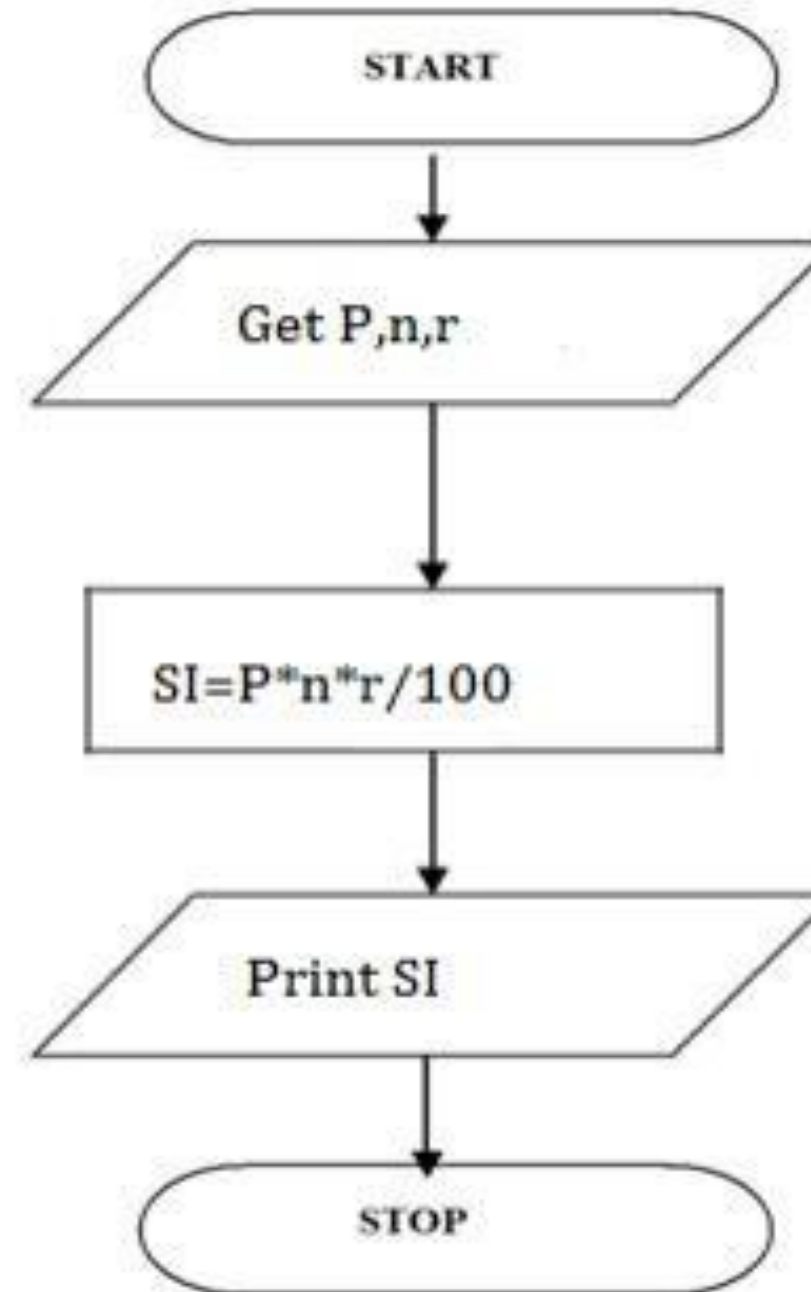


Algorithm

- Step 1: Start
- Step 2: get P, n, r value
- Step 3: Calculate $SI = (p * n * r) / 100$
- Step 4: Display S
- Step 5: Stop

Pseudo Code

- BEGIN
- READ P, n, r
- CALCULATE S
- $SI = (p * n * r) / 100$
- DISPLAY SI
- END





CALCULATING ENGINEERING CUTOFF

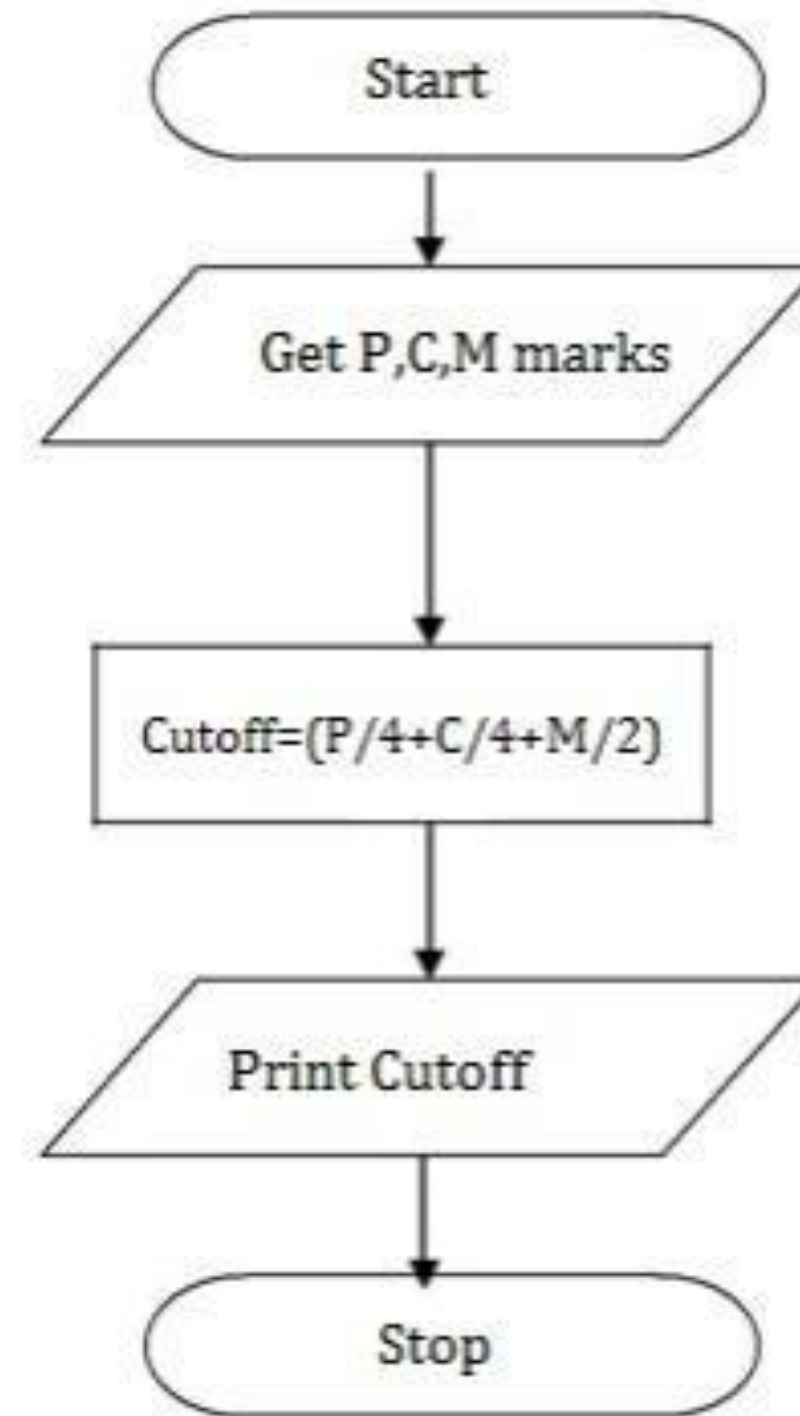


Algorithm

- Step 1: Start
- Step 2: get P,C,M value
- Step 3: calculate Cutoff= $(P/4+C/4+M/2)$
- Step 4: Display Cutoff
- Step 5: Stop

Pseudo Code

- BEGIN
- READ P,C,M
- CALCULATE
- Cutoff= $(P/4+C/4+M/2)$
- DISPLAY Cutoff
- END





TO CHECK GREATEST OF TWO NUMBERS

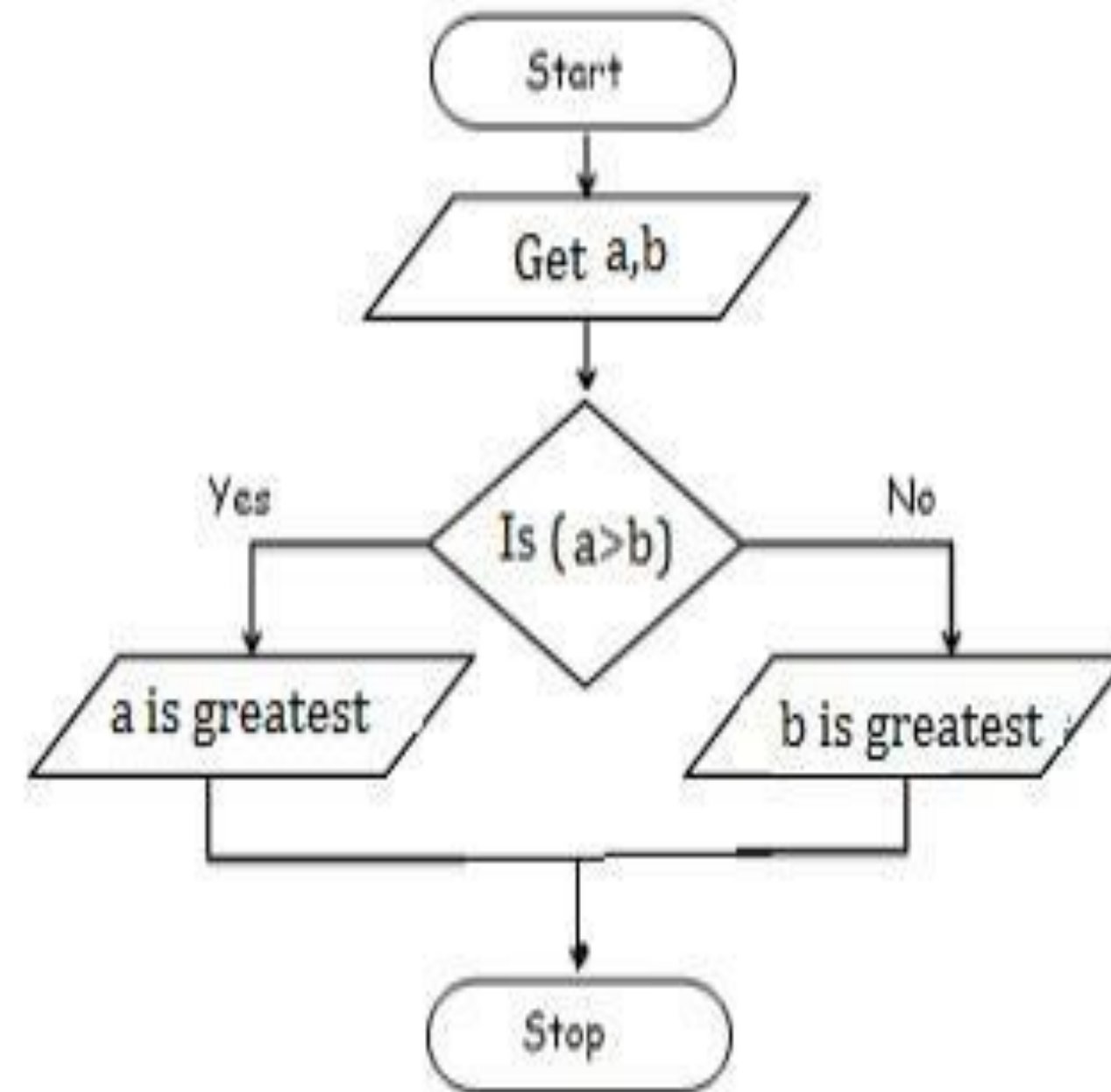


Algorithm

- Step 1: Start
- Step 2: get a,b value
- Step 3: check if($a > b$) print a is greater
- Step 4: else b is greater
- Step 5: Stop

Pseudo Code

- BEGIN
- READ a,b
- IF ($a > b$) THEN
- DISPLAY a is greater
- ELSE
- DISPLAY b is greater
- END IF
- END





TO CHECK LEAP YEAR OR NOT

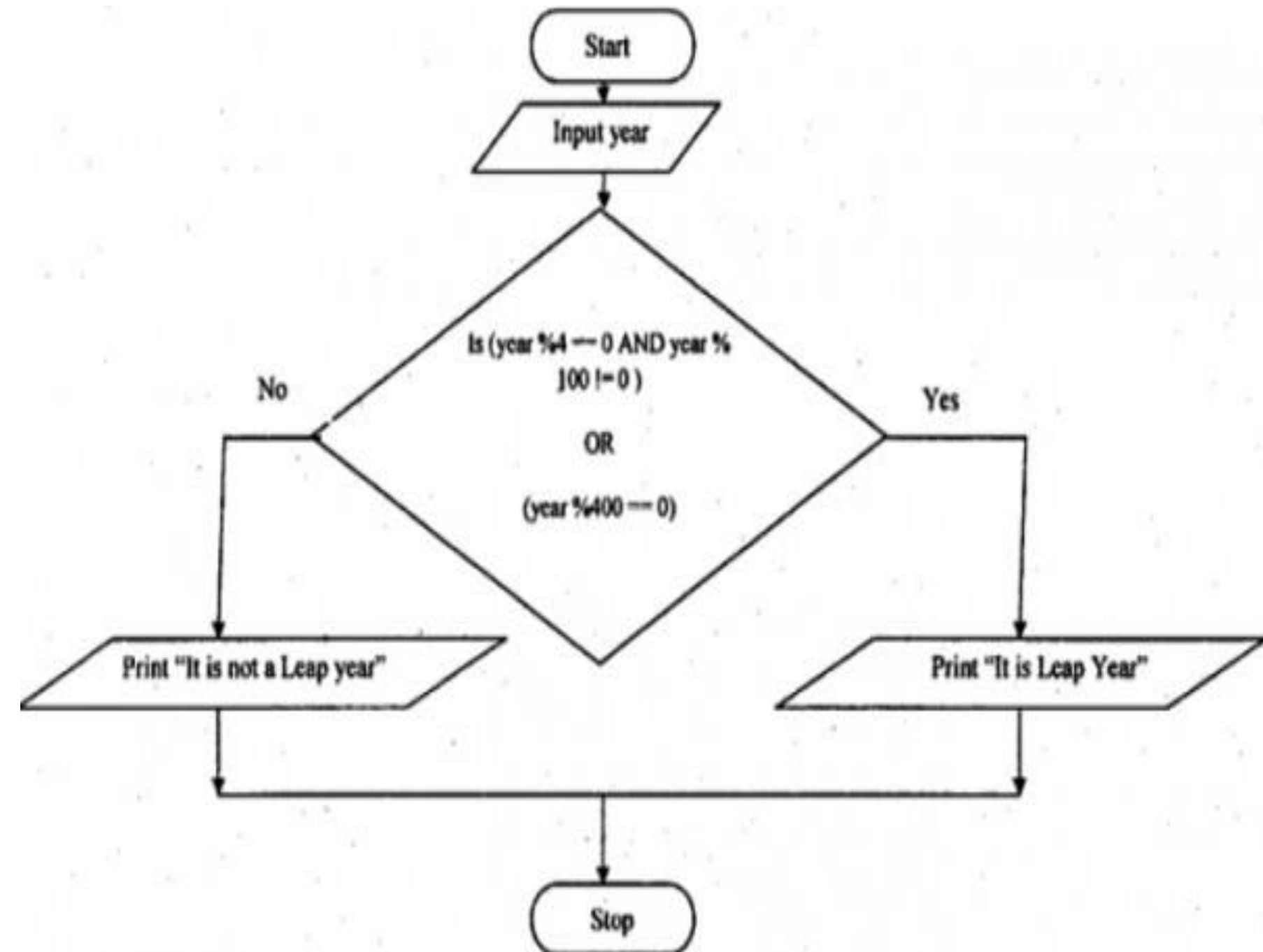


Algorithm

- Step 1: Start
- Step 2: get y
- Step 3: if($y \% 4 == 0$) print leap year
- Step 4: else print not leap year
- Step 5: Stop

Pseudo Code

- BEGIN
- READ y
- IF ($y \% 4 == 0$) THEN
- DISPLAY leap year
- ELSE
- DISPLAY not leap year
- END IF
- END





TO CHECK POSITIVE OR NEGATIVE NUMBER

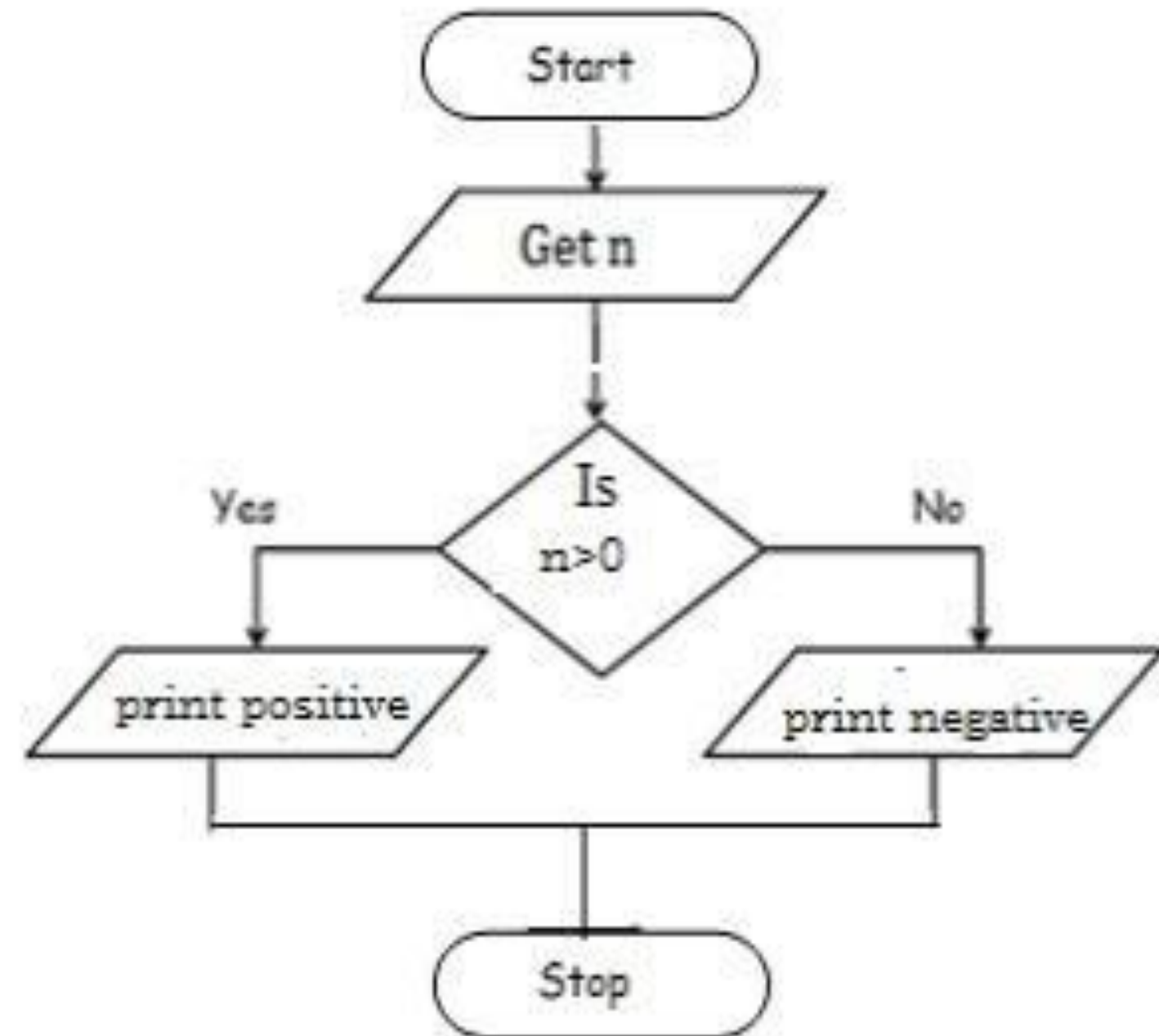


Algorithm

- Step 1: Start
- Step 2: get num
- Step 3: check if($\text{num} > 0$) print a is positive
- Step 4: else num is negative
- Step 5: Stop

Pseudo Code

- BEGIN
- READ num
- IF ($\text{num} > 0$) THEN
- DISPLAY num is positive
- ELSE
- DISPLAY num is negative
- END IF
- END





TO CHECK ODD OR EVEN NUMBER

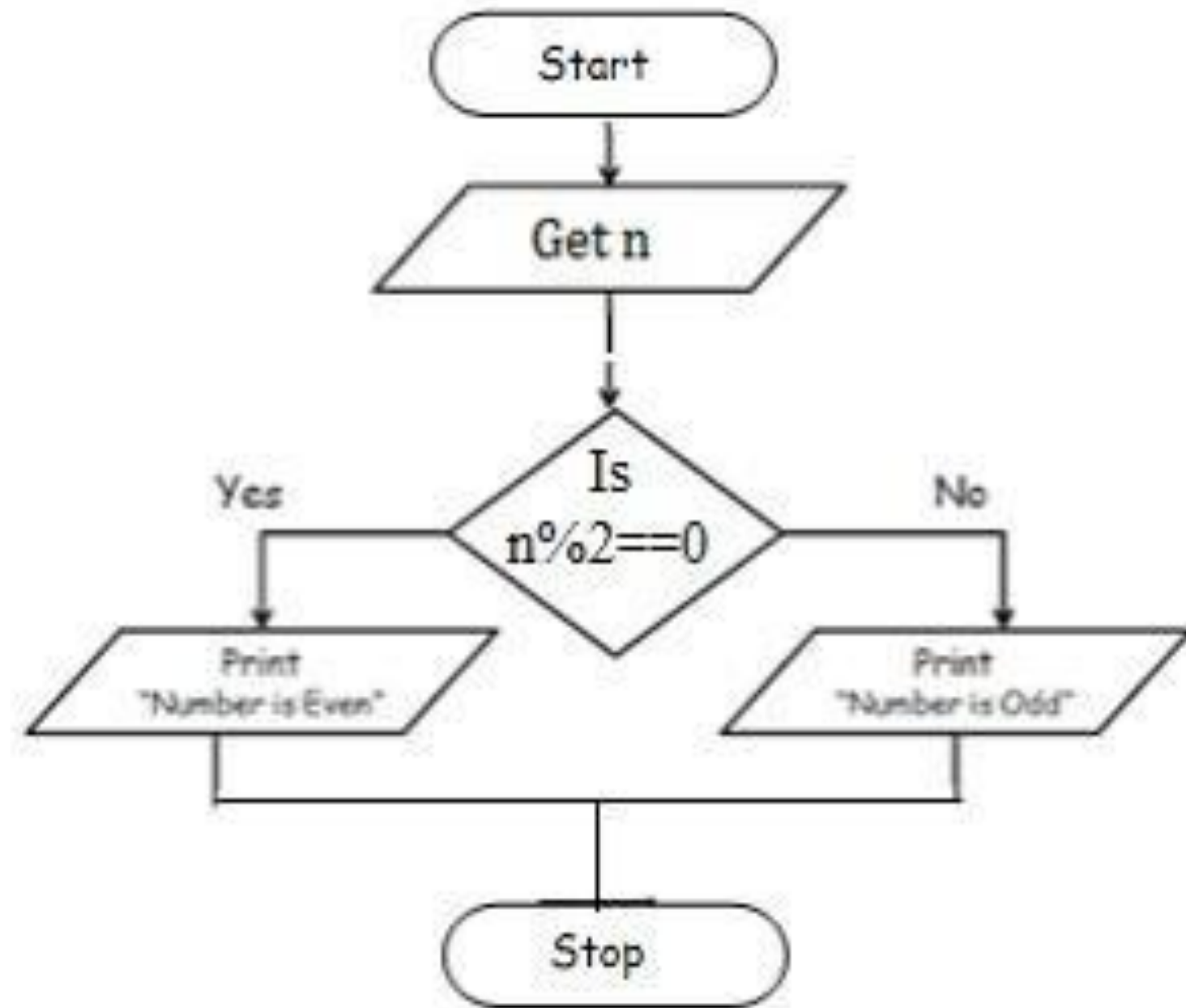


Algorithm

- Step 1: Start
- Step 2: get num
- Step 3: check if($\text{num} \% 2 == 0$) print num is even
- Step 4: else num is odd
- Step 5: Stop

Pseudo Code

- BEGIN
- READ num
- IF ($\text{num} \% 2 == 0$) THEN
- DISPLAY num is even
- ELSE
- DISPLAY num is odd
- END IF
- END





TO CHECK GREATEST OF THREE NUMBERS

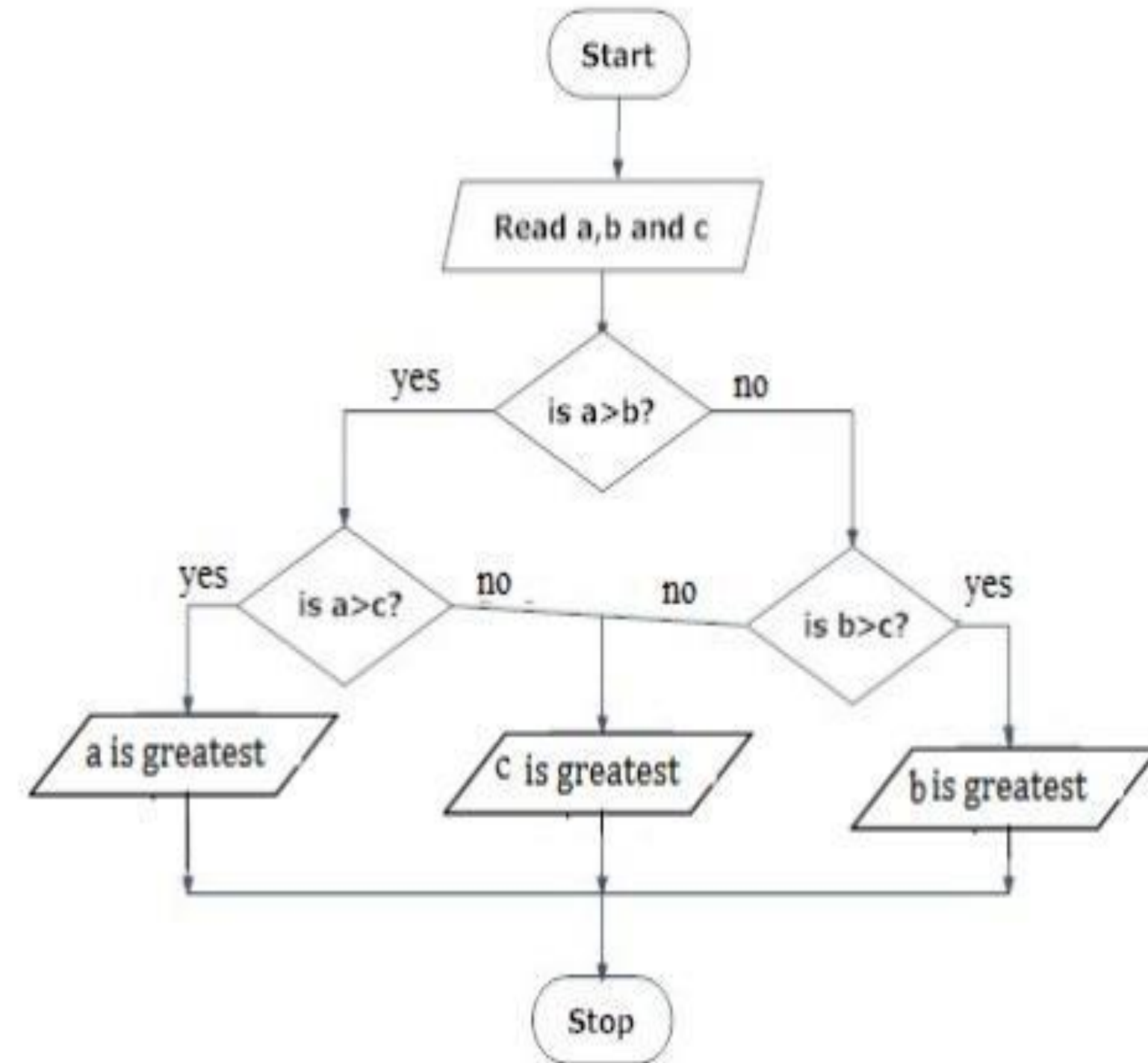


Algorithm

- Step 1: Start
- Step 2: Get A, B, C
- Step 3: if(A>B) goto Step4 else goto step5
- Step 4: If(A>C) print A else print C
- Step 5: If(B>C) print B else print C
- Step 6: Stop

Pseudo Code

- BEGIN
- READ a, b, c
- IF (a>b) THEN
- IF(a>c) THEN
- DISPLAY a is greater
- ELSE
- DISPLAY c is greater
- END IF
- ELSE
- IF(b>c) THEN
- DISPLAY b is greater
- ELSE
- DISPLAY c is greater
- END IF
- END





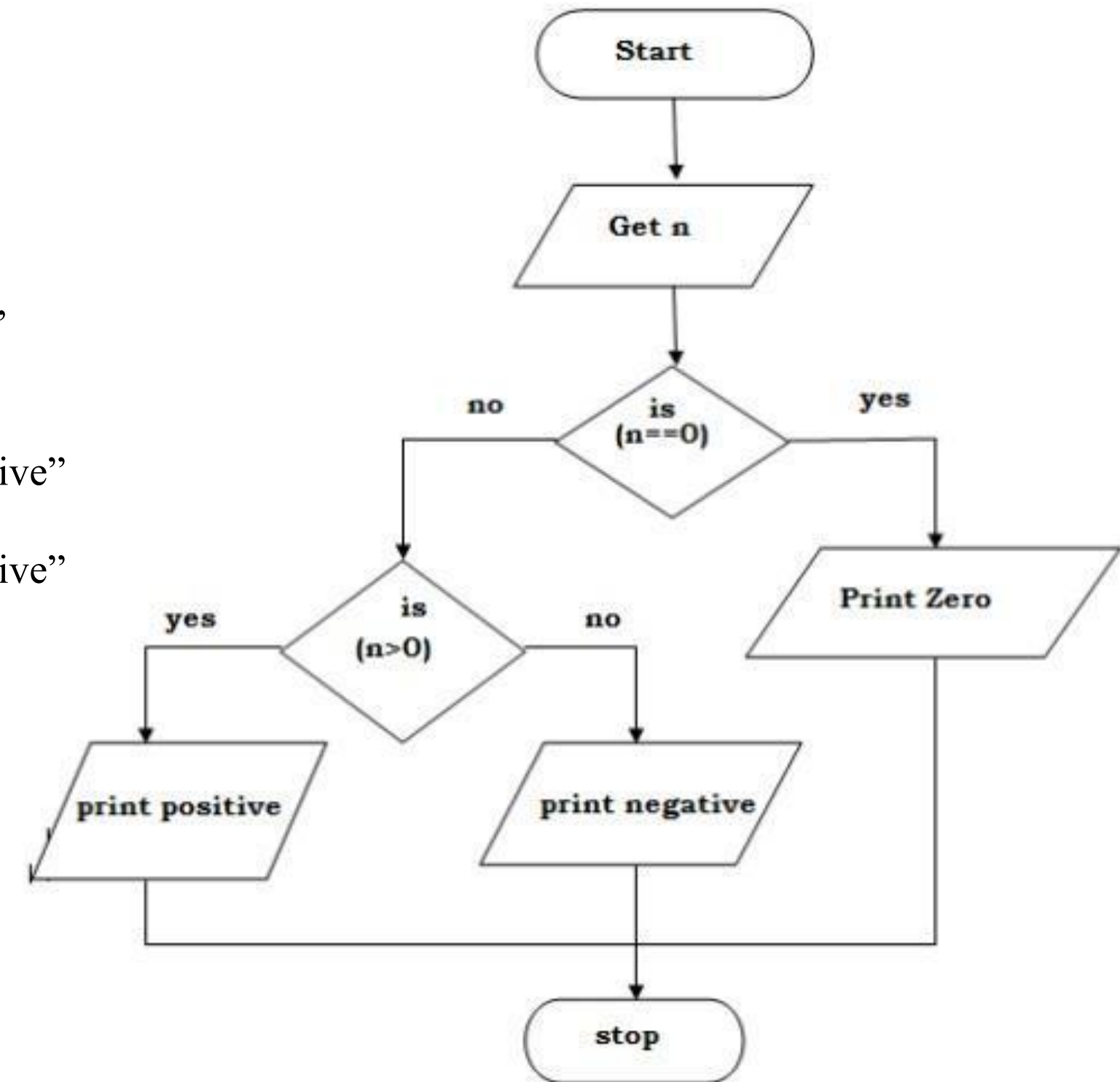
CHECK WHETHER GIVEN NUMBER IS +VE, -VE OR ZERO.

Algorithm

- Step 1: Start
- Step 2: Get n value.
- Step 3: if (n ==0) print “Given number is Zero” Else goto step4
- Step 4: if (n > 0) then Print “Given number is +ve”
- Step 5: else Print “Given number is -ve”
- Step 6: Stop

Pseudo Code

- BEGIN
- GET n
- IF(n==0) THEN
- DISPLAY “ n is zero”
- ELSE
- IF(n>0) THEN
- DISPLAY “n is positive”
- ELSE
- DISPLAY “n is positive”
- END IF
- END IF
- END





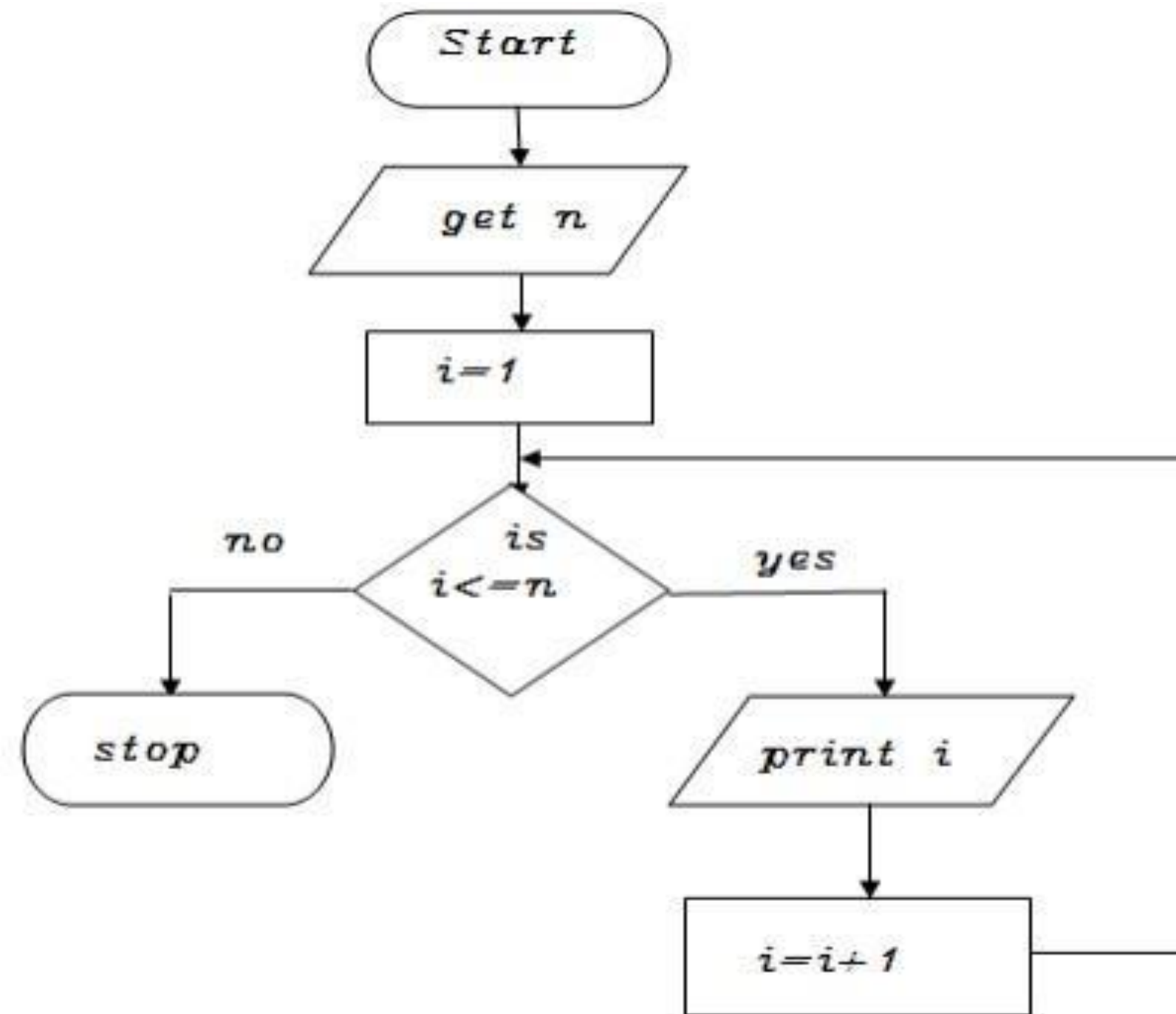
TO PRINT N ODD NUMBERS



Algorithm

- Step 1: start
- step 2: get n value
- step 3: set initial value $i=1$
- step 4: check if($i \leq n$) goto step 5 else goto step 8
- step 5: print i value
- step 6: increment i value by 2
- step 7: goto step 4
- step 8: stop

- BEGIN
- GET n
- INITIALIZE $i=1$
- WHILE($i \leq n$) DO
 - PRINT i
 - $i=i+2$
- ENDWHILE
- END





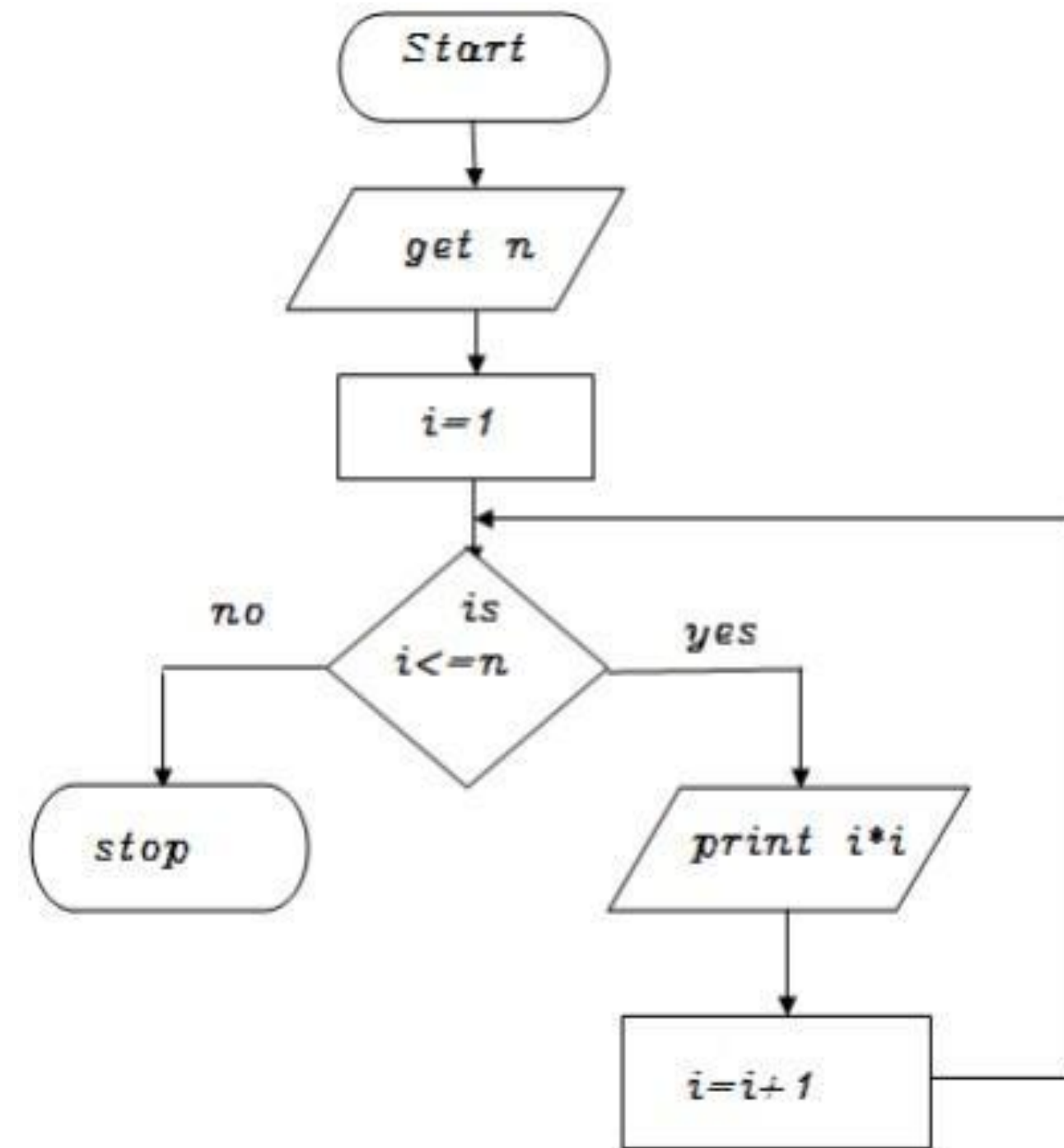
TO PRINT SQUARES OF A NUMBER



Algorithm

- step 1: start
- step 2: get n value
- step 3: set initial value $i=1$
- step 4: check i value if($i \leq n$) goto step 5 else goto step 8
- step 5: print $i*i$ value
- step 6: increment i value by 1
- step 7: goto step 4
- step 8: stop

- BEGIN
- GET n
- INITIALIZE $i=1$
- WHILE($i \leq n$) DO
 - PRINT $i*i$
 - $i=i+1$
- ENDWHILE
- END





Thank
you

rawpixel