**1.AIM:** Write an algorithm, flowchart and pseudocode to find area of a rectangle.

**AlGORITHM:**

Step 1: Start

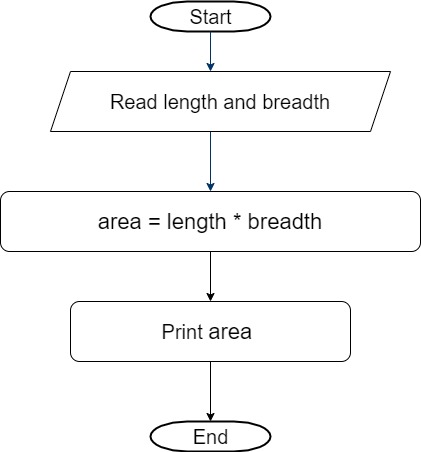
Step 2: Input length and breadth

Step 3: area = length \* breadth

Step 4: print area

Step 5: stop

**FLOWCHART:**



**PSEUDOCODE:**

START

INPUT L , B

READ L

READ B

Area = L \* B

Print area

STOP

**2.AIM:** Write an algorithm, flowchart and pseudocode for Calculating area and circumference of circle.

**ALGORITHM:**

Step 1: Start

Step 2: Input radius

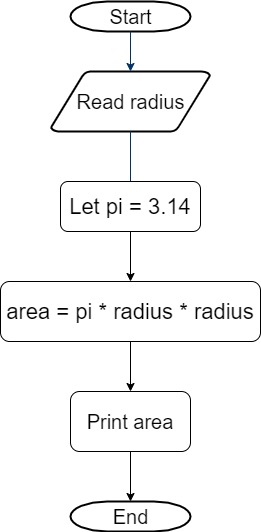
Step 3: let pi = 3.14

Step 4: area = pi \* radius \* radius

Step 6: print area

Step 7: stop

**FLOWCHART:**



**PSEUDOCODE:**

START

INPUT radius

LET pi = 3.14

area = pi \* radius \* radius

PRINT area

STOP

**3.AIM:**  Write an algorithm flowchart and pseudocode for Calculating simple interest

**Algorithm:**

Step 1:Start

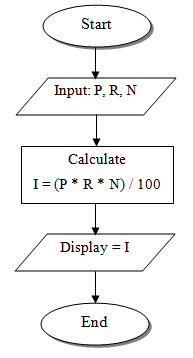
Step 2:Read Principal Amount, Rate and Time

Step 3:Calculate Interest using formula SI= ((amount\*rate\*time)/100)

Step 4:Print Simple Interest

Step 5:Stop

**FLOWCHART:**



**PSEUDOCODE:**

1. Start the program.

2. Read Principal amount, rate of interest and time period.

3. Calculate interest by using the formula.    Simple interest=(Principal amount x rate of interest x time period)/100

4. Print the simple interest.

5. Stop the program.

**4.AIM: WRITE AN ALGORITHM FOR CALCULATING ENGINEERING CUTOFF**

**ALGORITHM:**

Step 1: Start

Step2: get P,C,M value

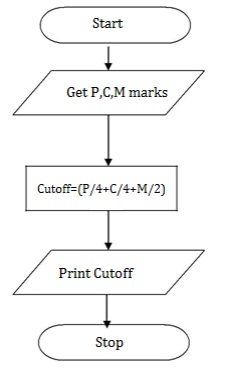
Step3:calculate

Cutoff= (P/4+C/4+M/2)

Step 4: Display Cutoff

Step 5: Stop

**FLOWCHART:**



**PSEUDOCODE:**

BEGIN

READ P,C,M

CALCULATE

Cutoff= (P/4+C/4+M/2)

DISPLAY Cutoff

END

**5. AIM: TO CHECK GREATEST OF TWO NUMBERS**

**ALGORITHM:**

Step 1: Start

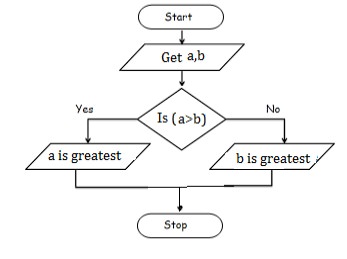
Step 2: get a,b value

Step 3: check if(a>b) print a is greater

Step 4: else b is greater

Step 5: Stop

**FLOWCHART:**



**PSEUDOCODE:**

BEGIN

READ a,b

IF (a>b) THEN

DISPLAY a is greater

ELSE

DISPLAY b is greater

END IF

END

**6. AIM: TO CHECK POSITIVE OR NEGATIVE NUMBER**

**ALGORITHM:**

**Step 1:**Start

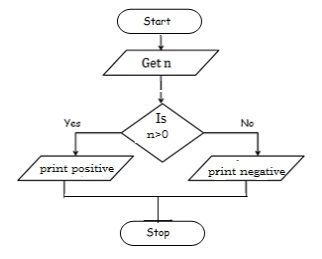
**Step 2:**get num

**Step 3:**check if(num>0) print a is positive

**Step 4:**else num is negative

**Step 5:**Stop

**FLOWCHART:**



**PSEUDOCODE:**

BEGIN

READ num

IF (num>0) THEN

DISPLAY num is positive

ELSE

DISPLAY num is negative

END IF

END

**7. AIM: TO CHECK ODD OR EVEN NUMBER**

**ALGORITHM:**

Step 1: Start

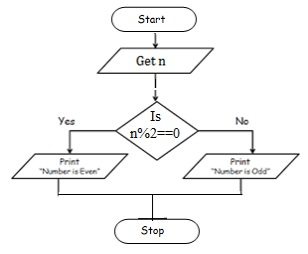
Step 2: get num

Step 3: check if(num%2==0) print num is even

Step 4: else num is odd

Step 5: Stop

**FLOWCHART:**



**PSEUDOCODE:**

BEGIN

READ num

IF (num%2==0) THEN

DISPLAY num is even

ELSE

DISPLAY num is odd

END IF

END

**8.AIM: TO CHECK GREATEST OF THREE NUMBERS**

**ALGORITHM:**

Step1: Start

Step2: Get A, B, C

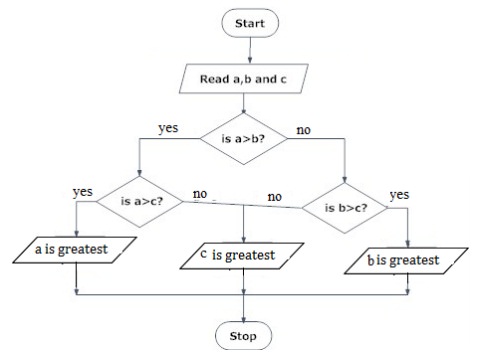
Step3: if(A>B) goto Step4 else goto step5

Step4: If(A>C) print A else print C

Step5: If(B>C) print B else print C

Step6: Stop

**FLOWCHART:**



**PSEUDOCODE:**

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

**9.AIM: Write an algorithm to check whether given number is +ve, -ve or zero.**

**ALGORITHM:**

**Step 1:**Start

**Step 2:**Get n value.

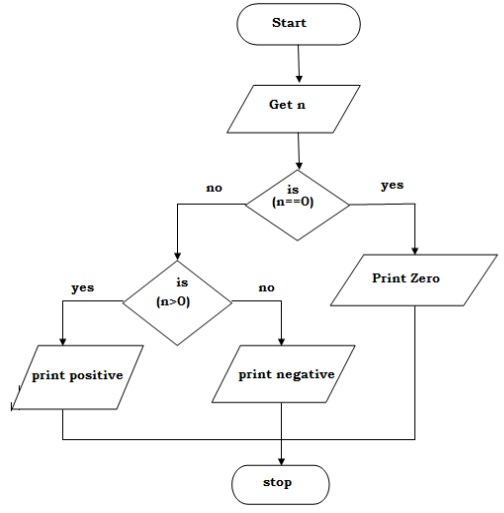
**Step 3:**if (n ==0) print “Given number is Zero” Else goto step4

**Step 4:**if (n > 0) then Print “Given number is +ve”

**Step 5:**else Print “Given number is-ve”

**Step 6:**Stop

**FLOWCHART:**



**PSEUDOCODE:**

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

**10.AIM: WRITE AN ALGORITHM TO PRINT ALL NATURAL NUMBERS UP TO N**

**ALGORITHM:**

**Step 1:**Start

**Step 2:**get n value.

**Step 3:**initialize i=1

**Step 4:**if (i<=n) go to step 5 else go to step 8

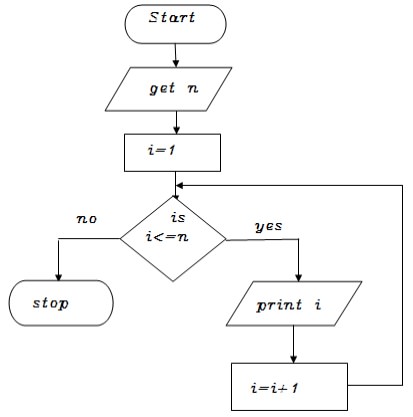
**Step 5:**Print i value

**step 6 :**increment i value by 1

**Step 7:**go to step 4

**Step 8:**Stop

**FLOWCHART:**



**PSEUDOCODE:**

BEGIN

GET n

INITIALIZE i=1

WHILE(i<=n) DO

PRINT i

i=i+1

ENDWHILE

END

**11.AIM: WRITE AN ALGORITHM 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<=n) goto step 5 else goto step 8

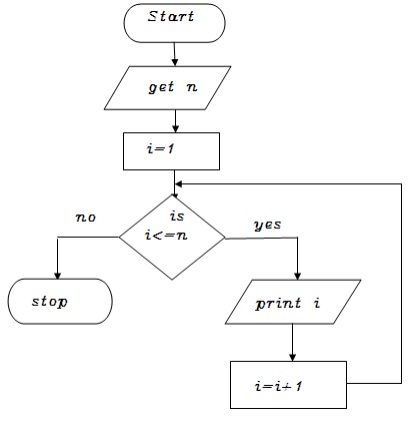
step 5: print i value

step 6: increment i value by 2

step 7: goto step 4

step 8: stop

**FLOWCHART:**



**PSEUDOCODE:**

BEGIN

GET n

INITIALIZE i=1

WHILE(i<=n) DO

           PRINT i

           i=i+2

ENDWHILE

END

**12. AIM: WRITE AN ALGORITHM TO PRINT N EVEN NUMBERS**

**ALGORITHM:**

Step 1: start

step 2: get n value

step 3: set initial value i=2

step 4: check if(i<=n) goto step 5 else goto step8

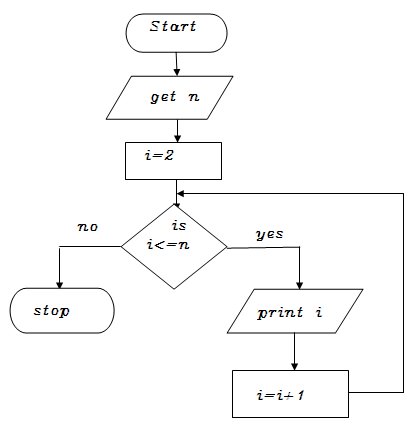
step 5: print i value

step 6: increment i value by 2

step 7: goto step 4

step 8: stop

**FLOWCHART:**



**PSEUDOCODE:**

BEGIN

GET n

INITIALIZE i=2

WHILE(i<=n) DO

           PRINT i

           i=i+2

ENDWHILE

END

**13. AIM: WRITE AN ALGORITHM TO PRINT SQUARES OF A NUMBER**

Step 1: start

step 2: get n value

step 3: set initial value i=1

step 4: check i value if(i<=n) goto step 5 else goto step8

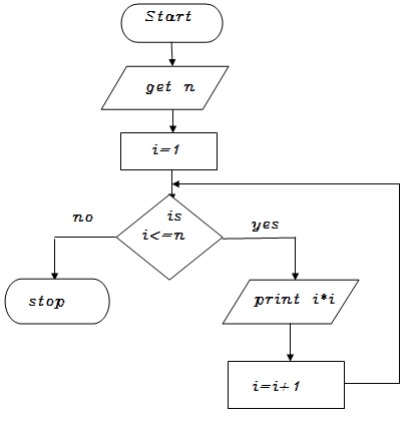
step 5: print i\*i value

step 6: increment i value by 1

step 7: goto step 4

step 8: stop

**FLOWCHART:**



**PSEUDOCODE:**

BEGIN

GET n

INITIALIZE i=1

WHILE(i<=n) DO

           PRINT i\*i

           i=i+2

ENDWHILE

END

**14. AIM: WRITE AN ALGORITHM TO PRINT TO PRINT CUBES 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<=n) goto step 5 else goto step8

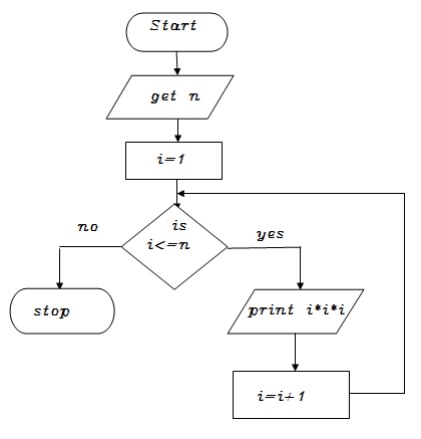
step 5: print i\*i \*i value

step 6: increment i value by 1

step 7: goto step 4

step 8: stop

**FLOWCHART:**



**PSEUDOCODE:**

BEGIN

GET n

INITIALIZE i=1

WHILE(i<=n) DO

           PRINT i\*i\*i

           i=i+2

ENDWHILE

END

**15.AIM: WRITE AN ALGORITHM TO FIND SUM OF A GIVEN NUMBER**

Step 1: start

step 2: get n value

step 3: set initial value i=1, sum=0

Step 4: check i value if(i<=n) goto step 5 else goto step8

step 5: calculate sum=sum+i

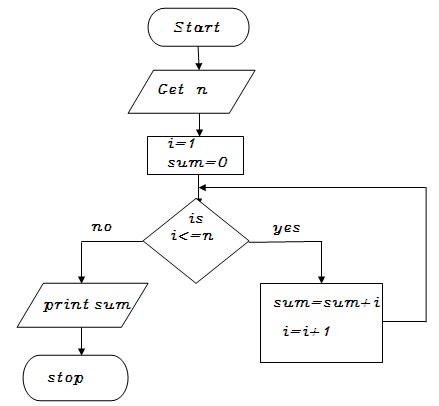
step 6: increment i value by 1

step 7: goto step 4

step 8: print sum value

step 9: stop

**FLOWCHART:**



**PSEUDOCODE:**

BEGIN

GET n

INITIALIZE i=1,sum=0

WHILE(i<=n) DO

           sum=sum+i

           i=i+1

ENDWHILE

PRINT sum

END

**16.AIM: WRITE AN ALGORITHM TO FIND FACTORIAL OF A GIVEN NUMBER**

Step 1: start

step 2: get n value

step 3: set initial value i=1, fact=1

Step 4: check i value if(i<=n) goto step 5 else goto step8

step 5: calculate fact=fact\*i

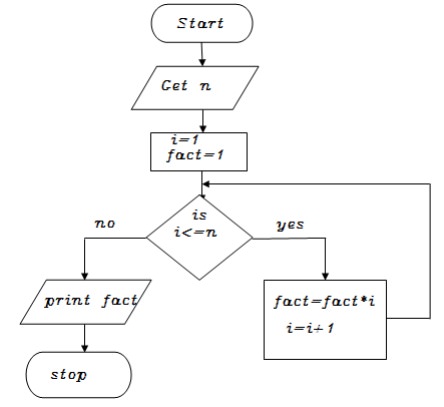
step 6: increment i value by 1

step 7: goto step 4

step 8: print fact value

step 9: stop

**FLOWCHART:**



**PSEUDOCODE:**

BEGIN

GET n

INITIALIZE i=1,fact=1

WHILE(i<=n) DO

           fact=fact\*i

           i=i+1

ENDWHILE

PRINT fact

END