Flowchart-1: Maximum of 2 Numbers.

num1>num2

Max(num1, num2)

#calling max function and storing the result in max

Input num1 as int

Input num2 as int

Create a function named max(int num1, int num2)

Declare the variables num1, num2 and maxnum

Print num2 is max

Print num1 is max

Algorithm-1: Maximum of 2 Numbers.

1. Start
2. Declare the variables num1, num2 and maxnum as int.
3. Create the function named max(int num1, int num2)
4. Take the inputs num1 and num2 from user.
5. Call the max function.
6. If (num1>num2), goto 8.
7. num2 is maximum.
8. num1 is maximum.
9. End

Algorithm-2: Roots of quadratic equation.

1. Start
2. Declare the int variables a, b, and c & float variables d, x1 and x2.
3. Take a, b and c values from user as input.
4. Calculate the value of d using the formula d=(b\*b)-(4\*a\*c).
5. If (d>=0), goto 7.
6. Print roots are imaginary., goto 9.
7. Calculate x1=(-b+√d)/(2\*a) and x2=(-b-√d)/(2\*a).
8. Print the x1 and x2 values.
9. Stop.

Flowchart-2: Roots of quadratic equation.

d>=0

Declare the variables a, b, c as int and x1, x2, d as float

Input a, b and c as int

d=(b\*b)-(4\*a\*c)

Roots are imaginary

x1=(-b+√d)/(2\*a)

x2=(-b-√d)/(2\*a)

Display roots x1 and x2

Flowchart-3: Area of a circle.

Algorithm-3: Area of a circle.

Display the area of circle a

Calculate the area

a=22.7\*r\*r

Input the value of radius r in float

Declare the variable a and r as float

1. Start.
2. Declare the float variables

a and r.

1. Take the input r from user.
2. Calculate the area using

formula a=22.7\*r\*r.

1. Print the area a.
2. End.

Flowchart-4: Sum of 2 numbers.

Display sum as int

Calculate

sum=num1+num2

Input the value of num1 and num2

Declare the variables num1, num2 and sum as int

Algorithm-4: Sum of 2 numbers.

1. Start
2. Declare the variables

num1, num2 and sum

as int.

1. Take the user input

num1 and num2.

1. Calculate the sum

using sum=num1+num2.

1. Print sum value.
2. End.

Flowchart-5: Display sum of numbers 1 to 10 using do-while loop (exit controlled loop).

Initialize i=1 and sum=0

Declare the variables i and sum as int

Print sum

i<=10

Increment the value of i

i++

sum=sum+i

Algorithm-5: Display sum of numbers 1 to 10 using do-while loop (exit controlled loop).

1. Start
2. Declare the variables i and sum as int.
3. Initialize variable i=1 and sum=0.
4. Perform sum=sum+i
5. Increment the i value (i++).
6. Do while (i<=10), goto 4.
7. Print sum.
8. End loop.

Algorithm-6: Display multiplication table of n using for loop (entry controlled loop).

1. Start
2. Declare variables i and n.
3. Initialize i=1.
4. Print (n)\*(i)=(n\*i)
5. Increment value of i (i++).
6. For (i<=10), goto 4.
7. Stop loop.

Flowchart-6: Display multiplication table of n using for loop (entry controlled loop).

Declare the variables n and i as int

Input n (the number whose multiplication table the user requires)

Initialize the variable

i=1

Increment the value of i

i++

Print

(n)\*(i)=(n\*i)

i>=10