SWAPPING TWO NUMBERS

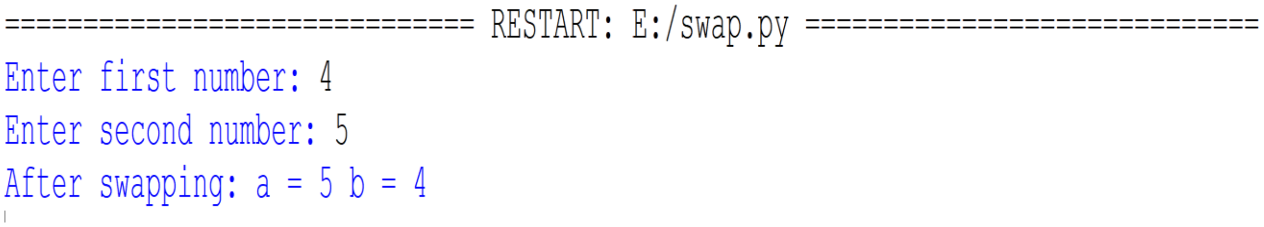
AIM:

ALGORITHM:

SOURCE CODE:

```
a = int(input("Enter first number: "))

b = int(input("Enter second number: "))

temp = a

a = b

b = temp

print("After swapping: a =", a, "b =", b)
```

OUTPUT:



```
============================== RESTART: E:/swap.py ==============================
Enter first number: 4
Enter second number: 5
After swapping: a = 5 b = 4
```

RESULT:

AREA OF RECTANGLE

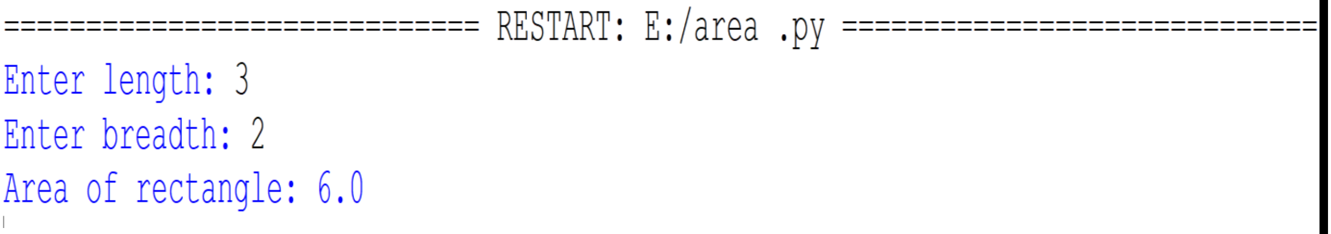
AIM:

ALGORITHM:

SOURCE CODE:

```
length = float(input("Enter length: "))

breadth = float(input("Enter breadth: "))

area = length * breadth

print("Area of rectangle:", area)
```

OUTPUT:



```
============================== RESTART: E:/area .py ==============================
Enter length: 3
Enter breadth: 2
Area of rectangle: 6.0
```

RESULT:

SQUARE ROOT OF A NUMBER

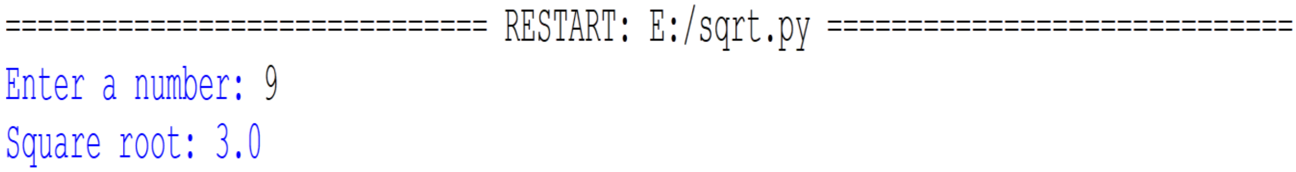
AIM:

ALGORITHM:

SOURCE CODE:

```
import math
num = float(input("Enter a number: "))
sqrt = math.sqrt(num)
print("Square root:", sqrt)
```

OUTPUT:



```
============================== RESTART: E:/sqrt.py =============================
Enter a number: 9
Square root: 3.0
```

RESULT:

AREA OF TRIANGLE

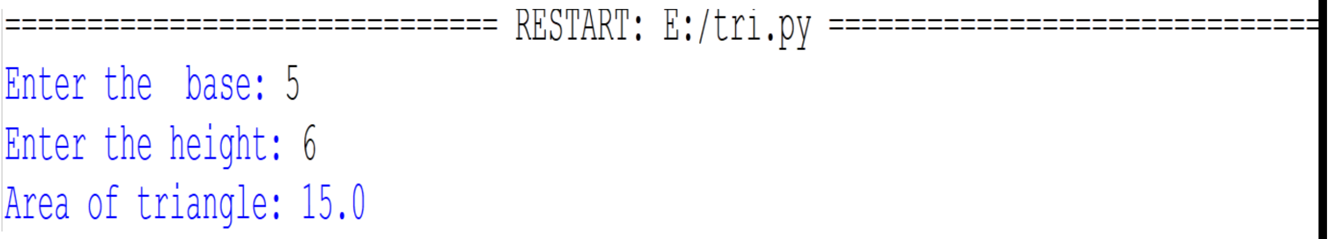
AIM:

ALGORITHM:

SOURCE CODE:

```
base = float(input("Enter the  base: "))
height = float(input("Enter the height: "))
area = 0.5 * base * height
print("Area of triangle:", area)
```

OUTPUT:



```
============================== RESTART: E:/tri.py ==============================
Enter the  base: 5
Enter the height: 6
Area of triangle: 15.0
```

RESULT:

KILOMETERS TO MILES

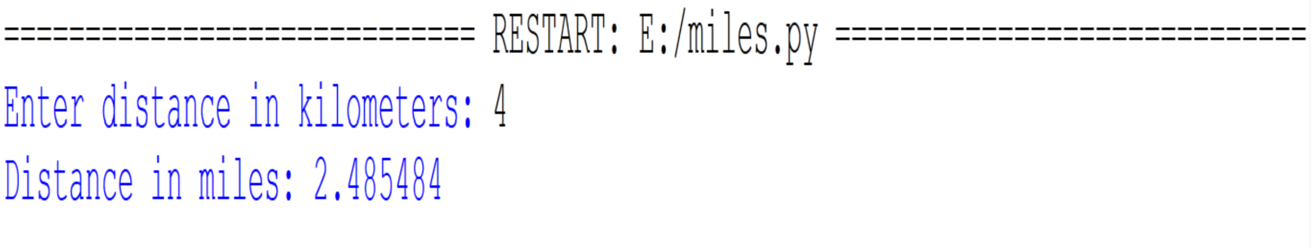
AIM:

ALGORITHM:

SOURCE CODE:

```
km = float(input("Enter distance in kilometers: "))
miles = km * 0.621371
print("Distance in miles:", miles)
```

OUTPUT:



```
============================= RESTART: E:/miles.py =============================
Enter distance in kilometers: 4
Distance in miles: 2.485484
```

RESULT:

CELCIUS TO FAHRENHEIT

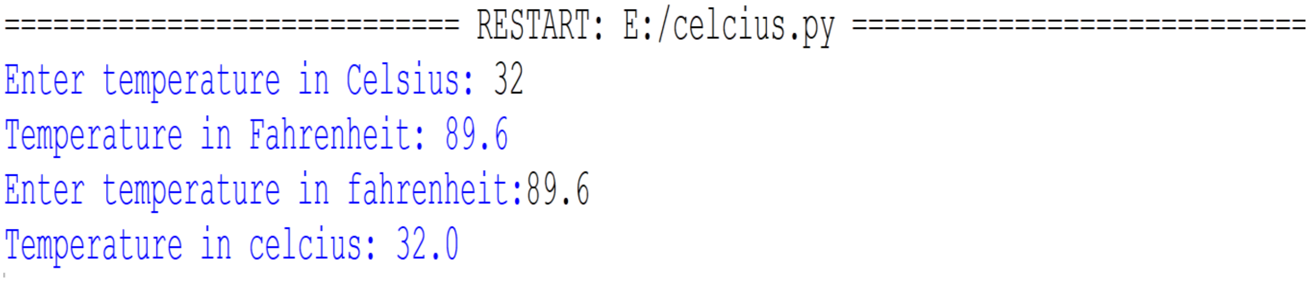
AIM:

ALGORITHM:

SOURCE CODE:

```
celsius = float(input("Enter temperature in Celsius: "))
fahrenheit = (celsius * 9/5) + 32
print("Temperature in Fahrenheit:", fahrenheit)
fahrenheit=float(input("Enter temperature in fahrenheit:"))
celcius=(fahrenheit-32)*5/9
print("Temperature in celcius:",celcius)
```

OUTPUT:



```
============================ RESTART: E:/celcius.py ============================
Enter temperature in Celsius: 32
Temperature in Fahrenheit: 89.6
Enter temperature in fahrenheit:89.6
Temperature in celcius: 32.0
```

RESULT:

SIMPLE AND COMPOUND INTEREST

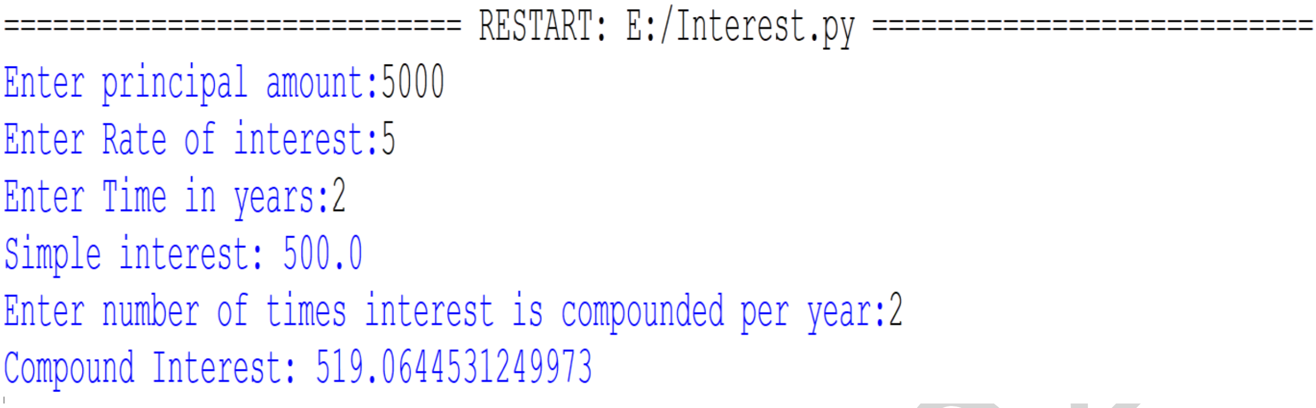
AIM:

ALGORITHM:

SOURCE CODE:

```
P=float(input("Enter principal amount:"))
R=float(input("Enter Rate of interest:"))
T=float(input("Enter Time in years:"))
SI=(P*R*T)/100
print("Simple interest:",SI)
N=float(input("Enter number of times interest is compounded per year:"))
A=P*(1+(R/(100*N)))**(N*T)
CI=A-P
print("Compound Interest:",CI)
```

OUTPUT:



```
============================= RESTART: E:/Interest.py ============================
Enter principal amount:5000
Enter Rate of interest:5
Enter Time in years:2
Simple interest: 500.0
Enter number of times interest is compounded per year:2
Compound Interest: 519.0644531249973
```

RESULT:

AREA OF CIRCLE

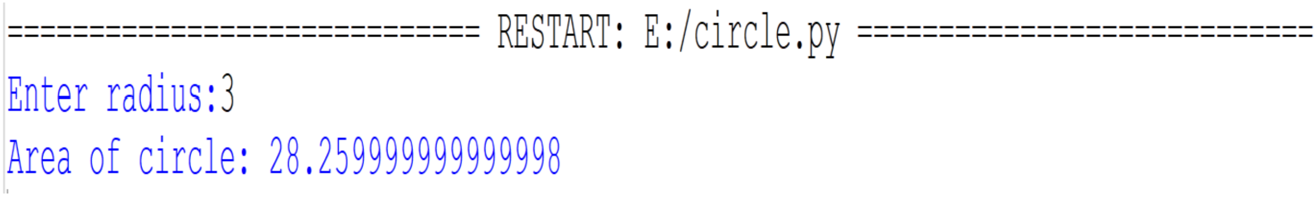
AIM:

ALGORITHM:

SOURCE CODE:

```
radius=float(input("Enter radius:"))
area= 3.14*radius*radius
print("Area of circle:",area)
```

OUTPUT:



```
============================== RESTART: E:/circle.py ==============================
Enter radius:3
Area of circle: 28.259999999999998
```

RESULT:

AVERAGE OF THREE NUMBERS

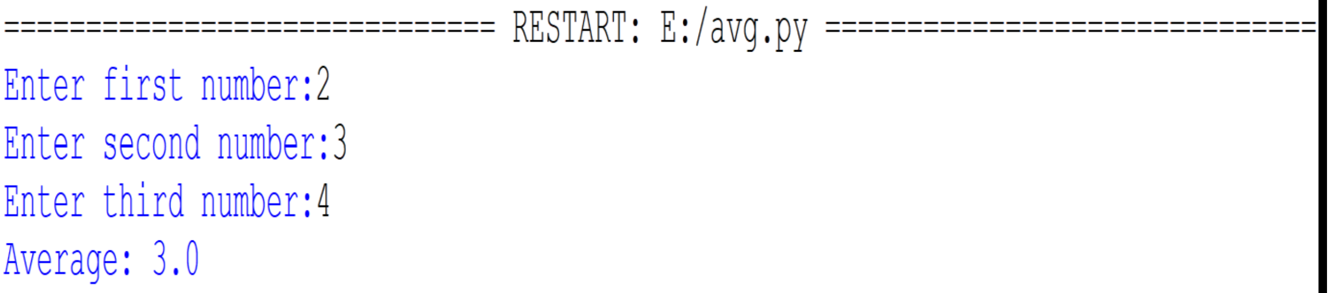
AIM:

ALGORITHM:

SOURCE CODE:

```
a=float(input("Enter first number:"))
b=float(input("Enter second number:"))
c=float(input("Enter third number:"))
average=(a+b+c)/3
print("Average:",average)
```

OUTPUT:



```
============================== RESTART: E:/avg.py ==============================
Enter first number:2
Enter second number:3
Enter third number:4
Average: 3.0
```

RESULT:

BODY MASS INDEX (BMI)

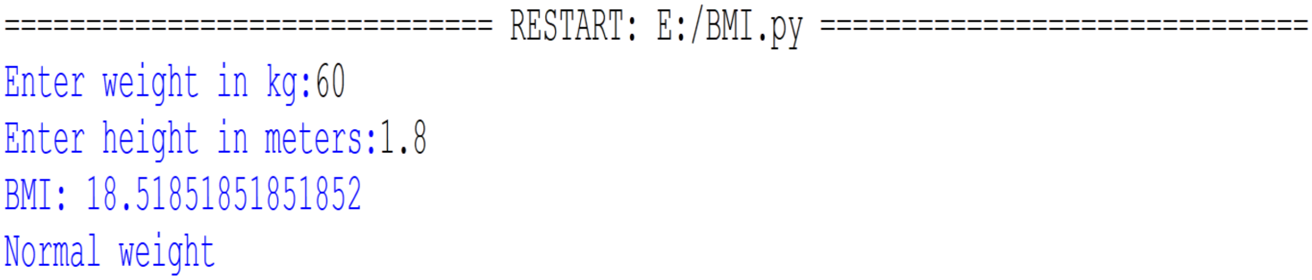
AIM:

ALGORITHM:

SOURCE CODE:

```
weight=float(input("Enter weight in kg:"))
height=float(input("Enter height in meters:"))
bmi=weight/(height*height)
print("BMI:",bmi)
if bmi<18.5:
   print("Under weight")
elif 18.5<=bmi<24.9:
   print("Normal weight")
elif 25<=bmi<29.9:
   print("Overweight")
else:
   print("Obesity")
```

OUTPUT:



```
============================== RESTART: E:/BMI.py ==============================
Enter weight in kg:60
Enter height in meters:1.8
BMI: 18.51851851851852
Normal weight
```

RESULT:

ARITHMETIC OPERATORS

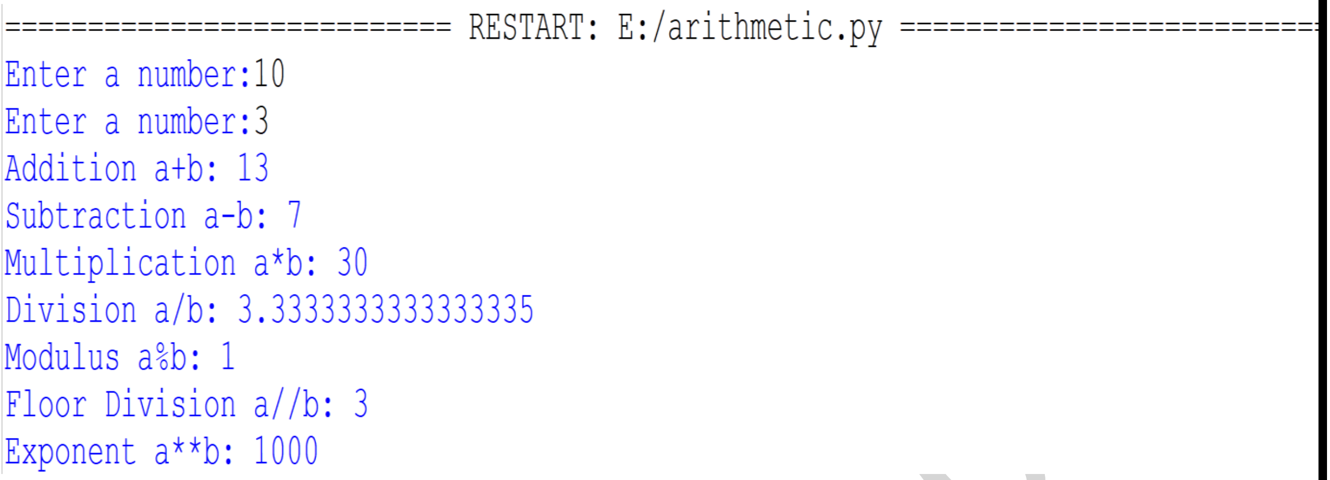
AIM:

ALGORITHM:

SOURCE CODE:

```
a=int(input("Enter a number:"))

b=int(input("Enter a number:"))

print("Addition a+b:", a + b)

print("Subtraction a-b:", a - b)

print("Multiplication a*b:", a * b)

print("Division a/b:", a / b)

print("Modulus a%b:", a % b)

print("Floor Division a//b:", a // b)

print("Exponent a**b:", a ** b)
```

OUTPUT:



```
=========================== RESTART: E:/arithmetic.py ==========================
Enter a number:10
Enter a number:3
Addition a+b: 13
Subtraction a-b: 7
Multiplication a*b: 30
Division a/b: 3.3333333333333335
Modulus a%b: 1
Floor Division a//b: 3
Exponent a**b: 1000
```

RESULT:

COMPARISON OPERATORS

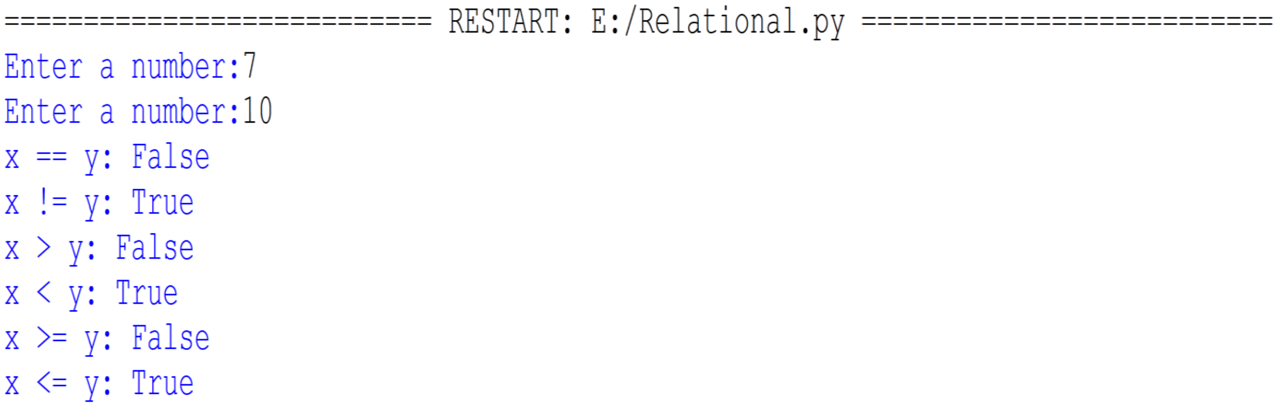
AIM:

ALGORITHM:

SOURCE CODE:

```
x=int(input("Enter a number:"))
y=int(input("Enter a number:"))
print("x == y:", x == y)
print("x != y:", x != y)
print("x > y:", x > y)
print("x < y:", x < y)
print("x >= y:", x >= y)
print("x <= y:", x <= y)
```

OUTPUT:



```
=========================== RESTART: E:/Relational.py ==========================
Enter a number:7
Enter a number:10
x == y: False
x != y: True
x > y: False
x < y: True
x >= y: False
x <= y: True
```

RESULT:

LOGICAL OPERATORS

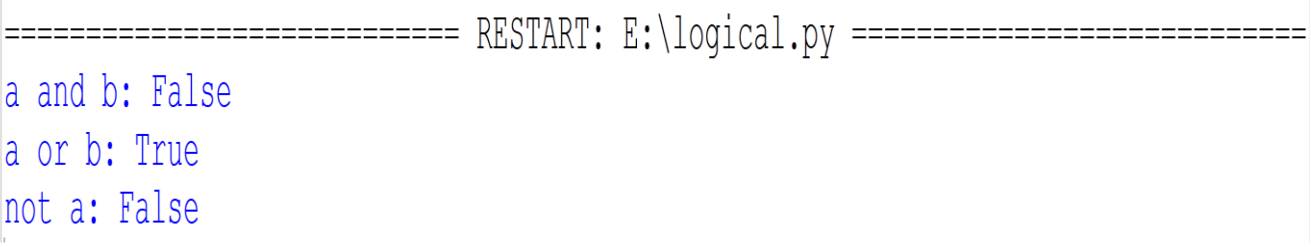
AIM:

ALGORITHM:

SOURCE CODE:

```
a = True
b = False
print("a and b:", a and b)
print("a or b:", a or b)
print("not a:", not a)
```

OUTPUT:



```
============================ RESTART: E:\logical.py ============================
a and b: False
a or b: True
not a: False
```

RESULT:

IDENTITY OPERATORS

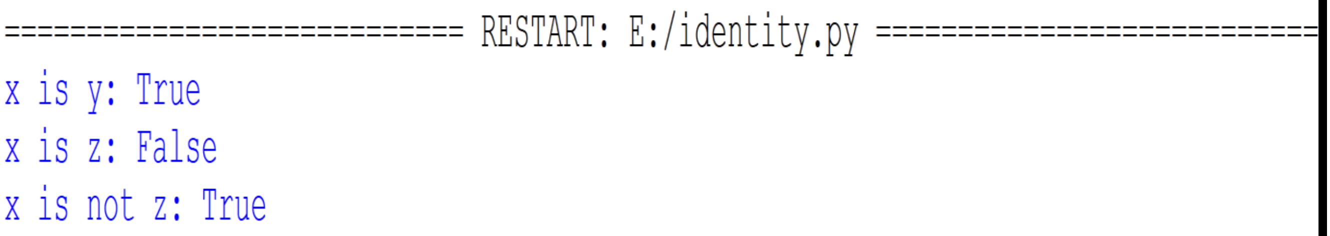
AIM:

ALGORITHM:

SOURCE CODE:

```
x = [1, 2, 3]
y = x
z = [1, 2, 3]
print("x is y:", x is y)
print("x is z:", x is z)
print("x is not z:", x is not z)
```

OUTPUT:



```
============================ RESTART: E:/identity.py ===========================
x is y: True
x is z: False
x is not z: True
```

RESULT:

MEMBERSHIP OPERATORS

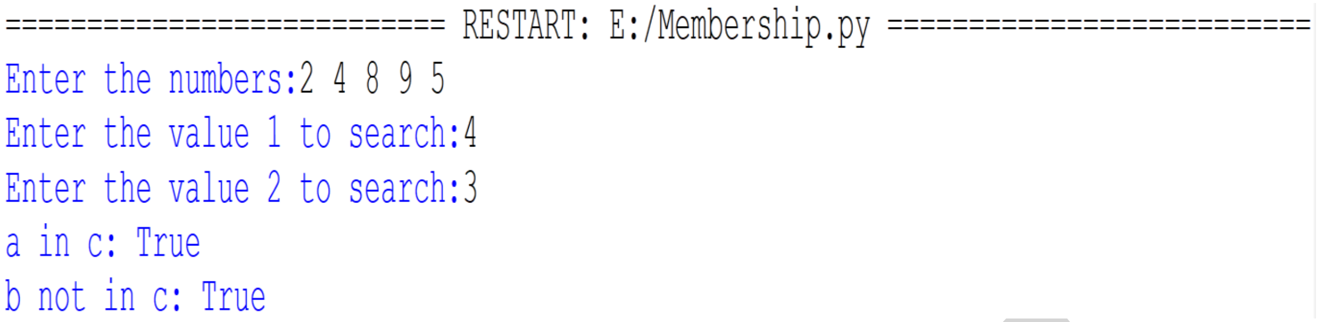
AIM:

ALGORITHM:

SOURCE CODE:

```
c=list(map(int,input("Enter the numbers:").split()))

a=int(input("Enter the value 1 to search:"))

b=int(input("Enter the value 2 to search:"))

print("a in c:",a in c)

print("b not in c:",b not in c)
```

OUTPUT:



```
============================ RESTART: E:/Membership.py ==========================
Enter the numbers:2 4 8 9 5
Enter the value 1 to search:4
Enter the value 2 to search:3
a in c: True
b not in c: True
```

RESULT:

BITWISE OPERATORS

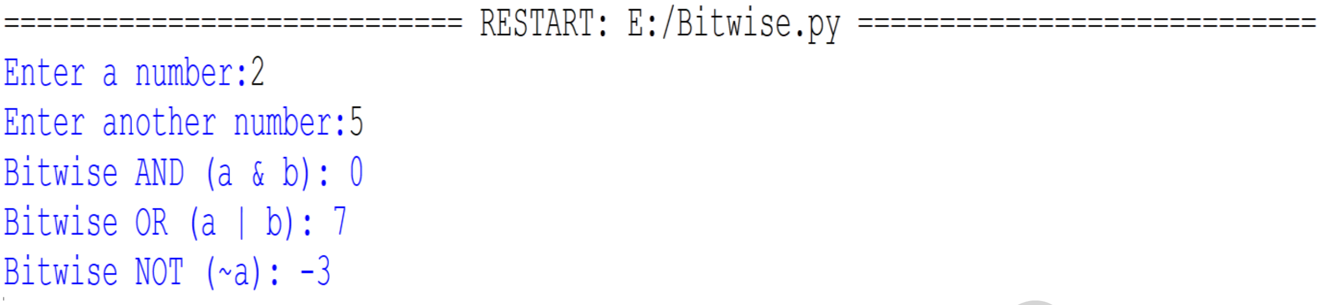
AIM:

ALGORITHM:

SOURCE CODE:

```
a=int(input("Enter a number:"))
b=int(input("Enter another number:"))
print("Bitwise AND (a & b):", a & b)
print("Bitwise OR (a | b):", a | b)
print("Bitwise NOT (~a):", ~a)
```

OUTPUT:



```
============================ RESTART: E:/Bitwise.py ============================
Enter a number:2
Enter another number:5
Bitwise AND (a & b): 0
Bitwise OR (a | b): 7
Bitwise NOT (~a): -3
```

RESULT:

OPERATOR PRECEDENCE

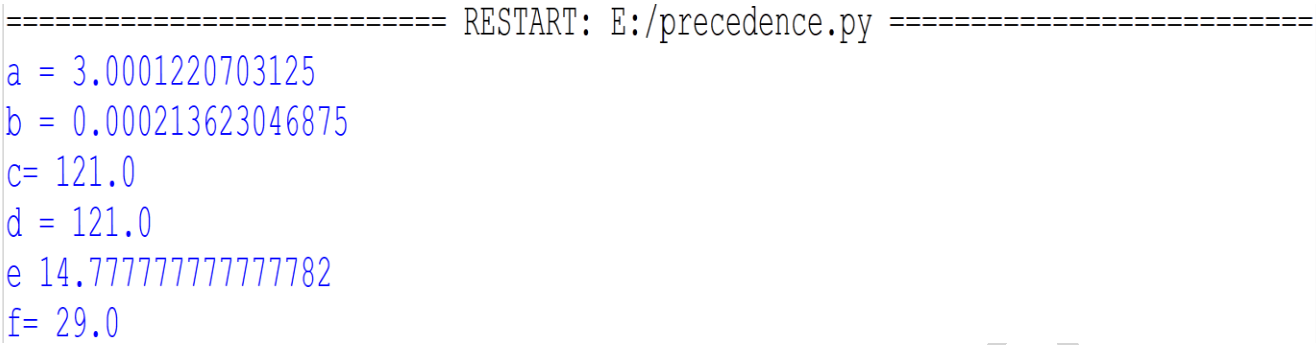
AIM:

ALGORITHM:

SOURCE CODE:

```
a = 3 + 4 * 2 / (1 - 5) ** 2 ** 3

print("a =", a)

b = (3 + 4) * 2 / (1 - 5) ** 2 ** 3

print("b =", b)

c = 100 % 3 + 5 ** (2 + 1) - 10 / 2

print("c=", c)

d = (100 % 3) + 5 ** (2 + 1) - (10 / 2)

print("d =", d)

e = ((4 + 5 * 2) / 3) ** 2 - 7

print("e", e)

f = (((4 + 5) * 2) / 3) ** 2 - 7

print("f=",f)
```

OUTPUT:



```
============================ RESTART: E:/precedence.py ==========================
a = 3.0001220703125
b = 0.000213623046875
c= 121.0
d = 121.0
e 14.777777777777782
f= 29.0
```

RESULT:

ELIGIBLE TO VOTE OR NOT

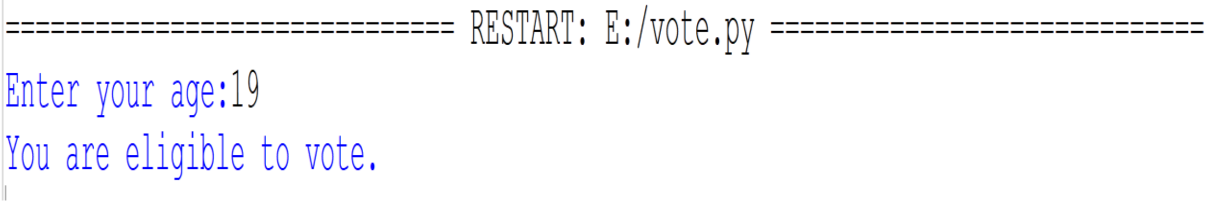
AIM:

ALGORITHM:

SOURCE CODE:

```
age=int(input("Enter your age:"))
if age>=18:
   print("You are eligible to vote.")
else:
   print("You are not eligible to vote.")
```

OUTPUT:



```
============================== RESTART: E:/vote.py =============================
Enter your age:19
You are eligible to vote.
```

RESULT:

SENIOR CITIZEN OR NOT

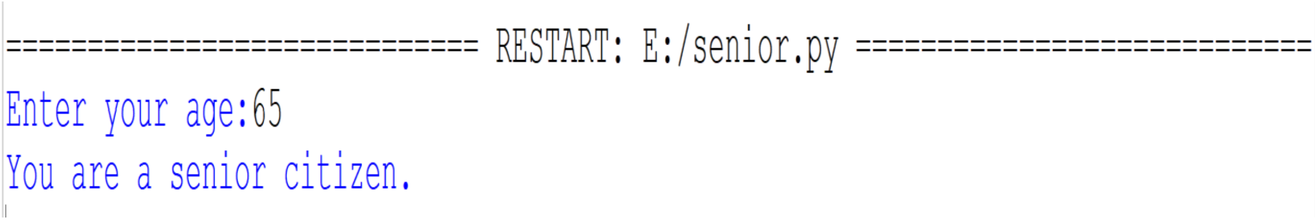
AIM:

ALGORITHM:

SOURCE CODE:

```
age=int(input("Enter your age:"))
if age>=60:
   print("You are a senior citizen.")
else:
   print("You are not a senior citizen.")
```

OUTPUT:



```
============================= RESTART: E:/senior.py =============================
Enter your age:65
You are a senior citizen.
```

RESULT:

BASED ON AGE DISPLAY DIFFERENT MESSAGES

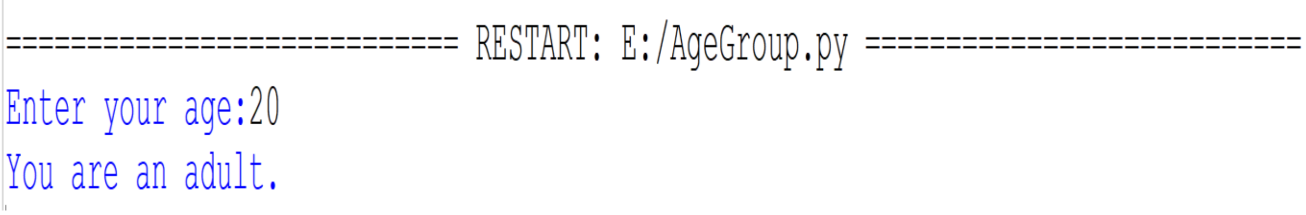
AIM:

ALGORITHM:

SOURCE CODE:

```
age=int(input("Enter your age:"))
if age<13:
   print("You are a child.")
elif 13<= age <=19:
   print("You are a teenager.")
elif 20<= age <=59:
   print("You are an adult.")
else:
   print("You are a senior citizen.")
```

OUTPUT:



```
============================ RESTART: E:/AgeGroup.py ===========================
Enter your age:20
You are an adult.
```

RESULT:

INTEGER INTO VARIOUS CATEGORIES

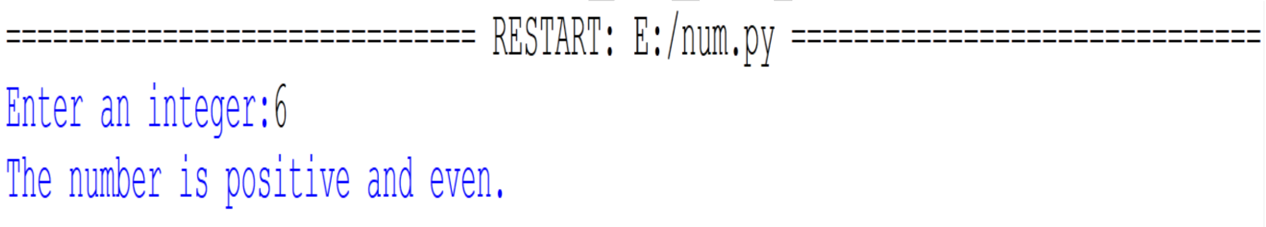
AIM:

ALGORITHM:

SOURCE CODE:

```
try:
    num=int(input("Enter an integer:"))
    if num>0 and num%2==0:
        print("The number is positive and even.")
    elif num>0 and num%2!=0:
        print("The number is positive and odd.")
    elif num<0 and num%2==0:
        print("The number is negative and even.")
    elif num<0 and num%2!=0:
        print("The number is negative and odd.")
    elif num==0:
        print("The number is zero.")
except ValueError:
    print("Invalid input!Please enter a valid integer.")
```

OUTPUT:



```
============================== RESTART: E:/num.py ==============================
Enter an integer:6
The number is positive and even.
```

RESULT:

SUM OF POSITIVE NUMBERS UNTILL '0' IS ENTERED

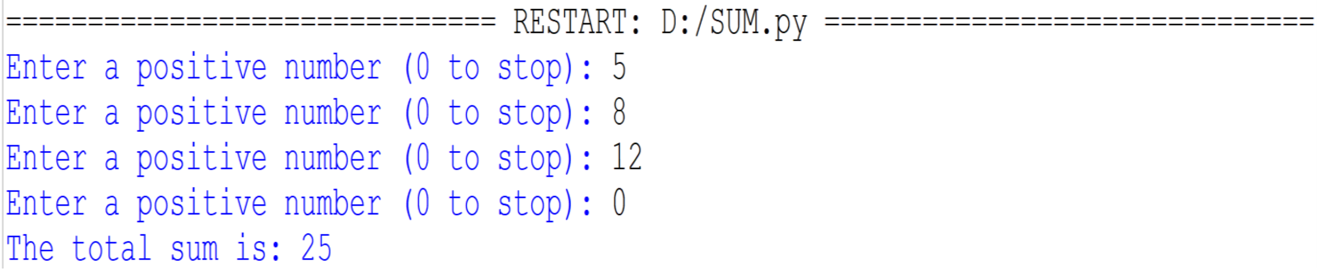
AIM:

ALGORITHM:

SOURCE CODE:

```
total = 0
num = int(input("Enter a positive number (0 to stop): "))
while num != 0:
   if num > 0:
      total += num
   else:
      print("Please enter only positive numbers.")
   num = int(input("Enter a positive number (0 to stop): "))
print("The total sum is:", total)
```

OUTPUT:



```
============================== RESTART: D:/SUM.py ==============================
Enter a positive number (0 to stop): 5
Enter a positive number (0 to stop): 8
Enter a positive number (0 to stop): 12
Enter a positive number (0 to stop): 0
The total sum is: 25
```

RESULT:

MULTIPLICATION TABLE

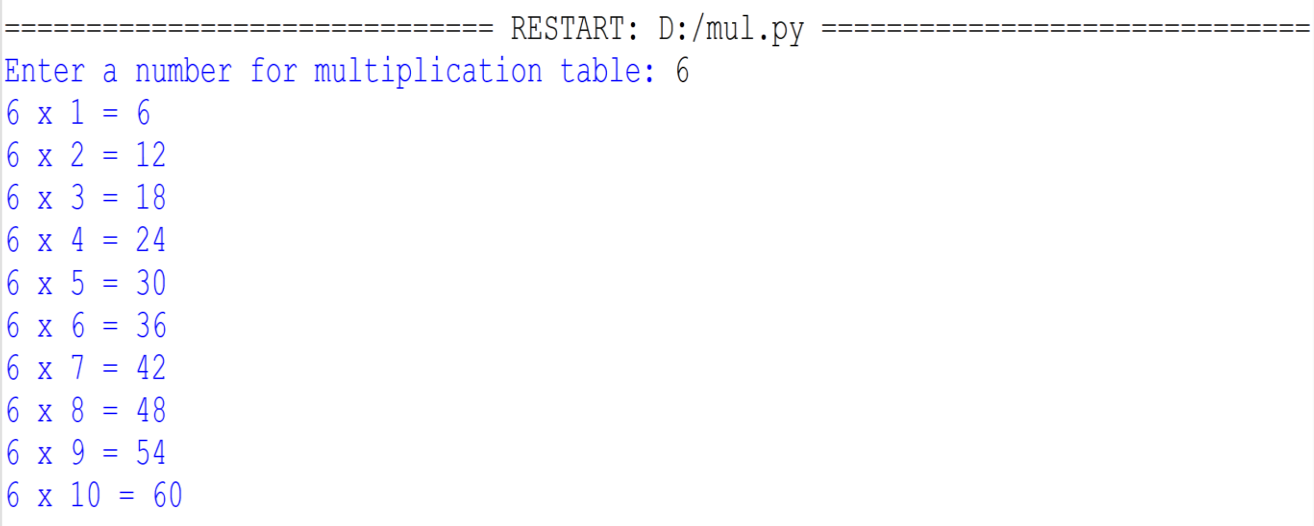
AIM:

ALGORITHM:

SOURCE CODE:

```
n = int(input("Enter a number for multiplication table: "))
i = 1
while i <= 10:
   print(n, "x", i, "=", n * i)
   i += 1
```

OUTPUT:



```
============================== RESTART: D:/mul.py ==============================
Enter a number for multiplication table: 6
6 x 1 = 6
6 x 2 = 12
6 x 3 = 18
6 x 4 = 24
6 x 5 = 30
6 x 6 = 36
6 x 7 = 42
6 x 8 = 48
6 x 9 = 54
6 x 10 = 60
```

RESULT:

SUM OF FIRST ‘N’ NATURAL NUMBERS

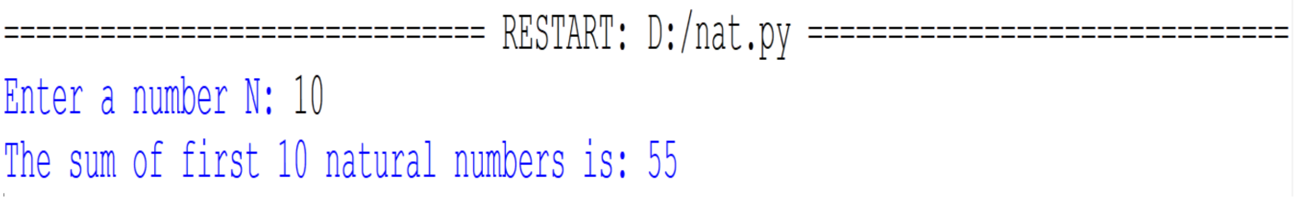
AIM:

ALGORITHM:

SOURCE CODE:

```
N = int(input("Enter a number N: "))
i = 1
total = 0
while i <= N:
   total += i
   i += 1
print("The sum of first", N, "natural numbers is:", total)
```

OUTPUT:



```
============================== RESTART: D:/nat.py ==============================
Enter a number N: 10
The sum of first 10 natural numbers is: 55
```

RESULT:

COUNTS THE NUMBER OF VOWELS FOR A STRING

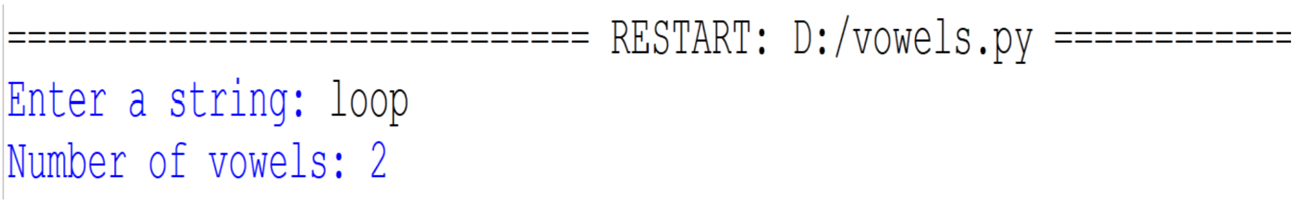
AIM:

ALGORITHM:

SOURCE CODE:

```
string = input("Enter a string: ")
vowels = "aeiouAEIOU"
count = 0
for ch in string:
    if ch in vowels:
        count += 1
print("Number of vowels:", count)
```

OUTPUT:



```
============================== RESTART: D:/vowels.py ============
Enter a string: loop
Number of vowels: 2
```

RESULT:

SUM OF ALL NUMBERS IN THE GIVEN LIST

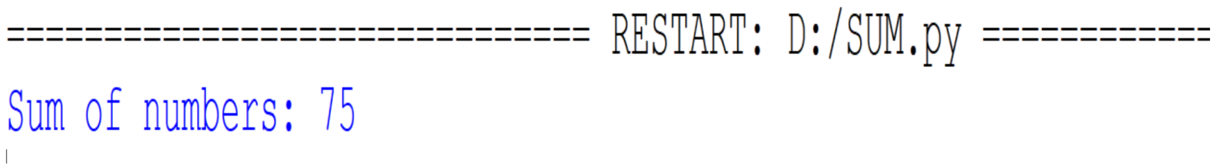
AIM:

ALGORITHM:

SOURCE CODE:

```
numbers = [5, 10, 15, 20, 25]
total = 0
for num in numbers:
      total += num
print("Sum of numbers:", total)
```

OUTPUT:



```
============================== RESTART: D:/SUM.py ============
Sum of numbers: 75
```

RESULT:

MULTIPLICATION TABLE OF ‘n’ FROM 11 TO 20

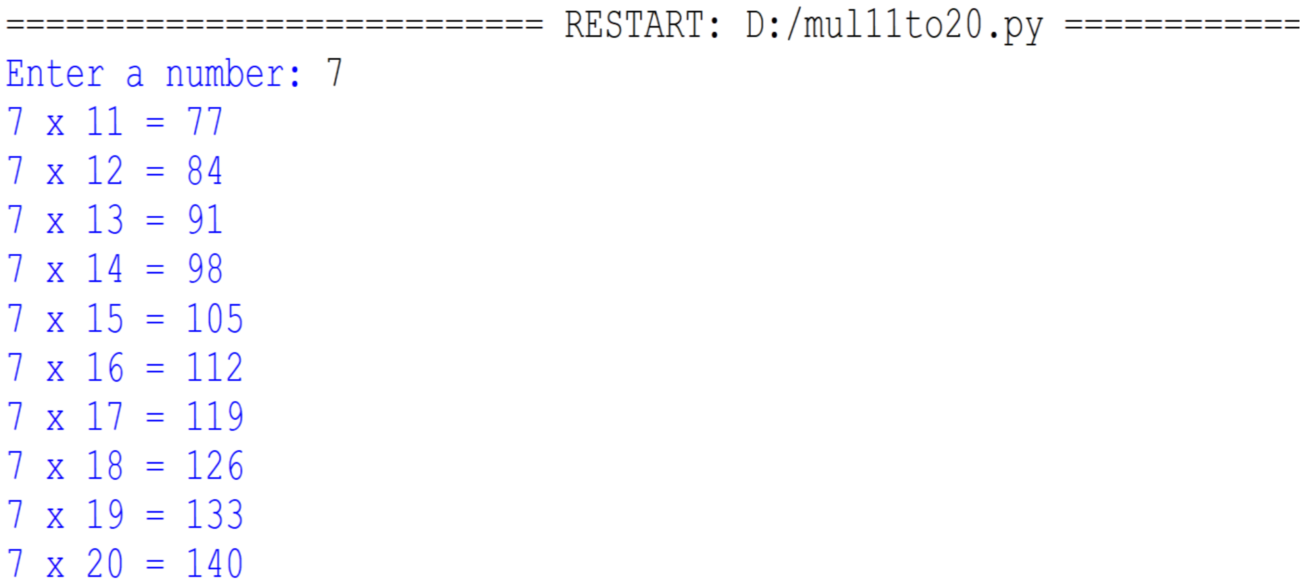
AIM:

ALGORITHM:

SOURCE CODE:

```
n = int(input("Enter a number: "))
for i in range(11, 21):
        print(n, "x", i, "=", n * i)
```

OUTPUT:



```
========================== RESTART: D:/mul11to20.py ============
Enter a number: 7
7 x 11 = 77
7 x 12 = 84
7 x 13 = 91
7 x 14 = 98
7 x 15 = 105
7 x 16 = 112
7 x 17 = 119
7 x 18 = 126
7 x 19 = 133
7 x 20 = 140
```

RESULT:

SKIP NUMBERS DIVISIBLE BY 5, STOP IF NEGATIVE, SUM OTHERS

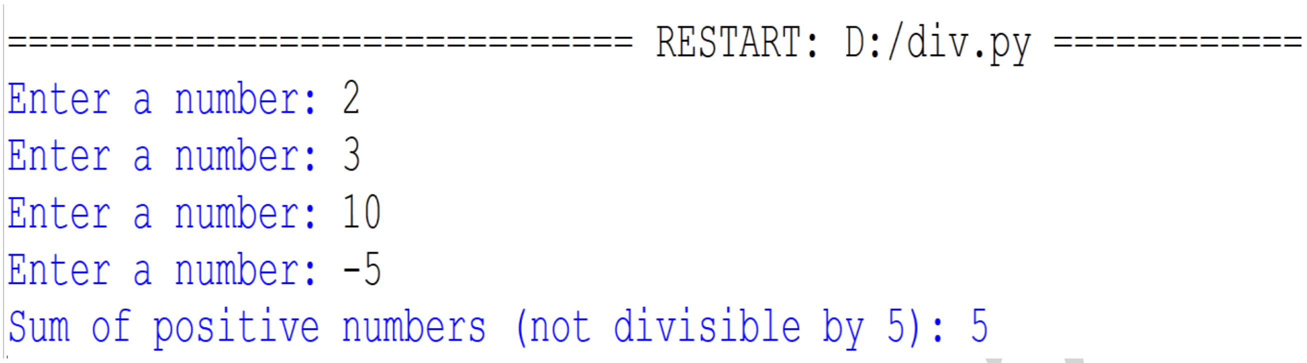
AIM:

ALGORITHM:

SOURCE CODE:

```
total = 0
while True:
   num = int(input("Enter a number: "))
    if num < 0:
        break
    if num % 5 == 0:
        continue
   total += num
print("Sum of positive numbers (not divisible by 5):", total)
```

OUTPUT:



```
============================== RESTART: D:/div.py ============
Enter a number: 2
Enter a number: 3
Enter a number: 10
Enter a number: -5
Sum of positive numbers (not divisible by 5): 5
```

RESULT: