In [6]:
```
#chef
for i in range(int(input())):
    a,b,a1,b1,a2,b2 = map(int,input().split())
    if (a==a1 or a==b1) and (b==-a1 or b==b2):
        print(1)
    elif (a==a2 or a==b2) and (b==a2 or b==b2):
        print(2)
    else:
        print(0)
```

```
0
0
1
1
```

In [24]:
```
#create set with defficult level of problem

for t in range (int(input())):
    a = list(map(int,input().split()))
    b = set(a)
    if len(a)==1:
        print(0)
    elif len(a)==2 and b.count(b[0])!=2:
        print(1)
    else:
        print(2)
```

```
2
2
```

In [19]:
```
#Develop a python code to check given two dates are equal with overload operaters

from datetime import date
x = date(2013,2,1)
y = date(2013,2,2)

print (x > y)
print (y > x)
```

```
False
True
```

In [35]:
```
dis1 = float(input("Enter first Distence as KM.MM :"))
dis2 = float(input("Enter second Distence as KM.MM :"))

char = input("Enter the operation would you like to perfom(+,-,*,/) :")

result = 0
if char == '+':
    result = dis1 + dis2
elif char == '-':
    result = dis1 - dis2
elif char == '*':
    result = dis1 * dis2
elif char == '/':
    result = dis1 / dis2
else:
    print("Please enter the above charaters only")

print(dis1,char,dis2, ':',result,"Km")
```

```
2.758 + 2.758 : 5.516 Km
```

In [6]:
```
from operator import length_hint


class Box:
  def __init__(self,Length,Breadth,Depth):
    self.Length = Length
    self.Breadth = Breadth
    self.Depth = Depth
  def display(self):
    print("Length: ",self.Length)
    print("Breadth: ",self.Breadth)
    print("Depth :",self.Depth)
    volume = (self.Length*self.Breadth*self.Depth)
    print("Volume of the given cube is :",volume)

class WeightBox(Box):
  def __init__(self,Length,Breadth,Depth,Weight):
    Box.__init__(self,Length,Breadth,Depth)
    self.Weight = Weight
  def display(self):
    Box.display(self)
    print("Weight: ",self.Weight)

class Colour(WeightBox):
  def __init__(self,Length,Breadth,Depth,Weight,colour):
    WeightBox.__init__(self,Length,Breadth,Depth,Weight)
    self.colour=colour
  def display(self):
    print("Length: ",self.Length)
    print("Breadth: ",self.Breadth)
    print("Depth: ",self.Depth)
    volume = (self.Length*self.Breadth*self.Depth)
    print("Volume of the given cube is :",volume)
    print("Weight: ",self.Weight)
    print("Colour: ",self.colour)



e = Colour(4,5,6,"2KG","Red")
e.display()
```

```
Length:  4
Breadth:  5
Depth:  6
Volume of the given cube is : 120
Weight:  2KG
Colour:  Red
```