

โปรแกรมข้างล่างนี้อ่านวันเดือนปีสองวัน แสดงราศีของวันทั้งสอง และแสดงจำนวนวันระหว่างสองวันนี้

```

mname = ["Jan", "Feb", "Mar", "Apr",
         "May", "Jun", "Jul", "Aug",
         "Sep", "Oct", "Nov", "Dec"]
date1 = input().split()
d1 = int(date1[0])
m1 = mname.index(date1[1][:3]) + 1
y1 = int(date1[2])
# หาราศี
if d1 >= 22 and m1==3 or d1 <=21 and m1==4 : z1 = "Aries"
elif d1 >= 22 and m1==4 or d1 <=21 and m1==5 : z1 = "Taurus"
elif d1 >= 22 and m1==5 or d1 <=21 and m1==6 : z1 = "Gemini"
elif d1 >= 22 and m1==6 or d1 <=21 and m1==7 : z1 = "Cancer"
elif d1 >= 22 and m1==7 or d1 <=21 and m1==8 : z1 = "Leo"
elif d1 >= 22 and m1==8 or d1 <=21 and m1==9 : z1 = "Virgo"
elif d1 >= 22 and m1==9 or d1 <=21 and m1==10 : z1 = "Libra"
elif d1 >= 22 and m1==10 or d1 <=21 and m1==11 : z1 = "Scorpio"
elif d1 >= 22 and m1==11 or d1 <=21 and m1==12 : z1 = "Sagittarius"
elif d1 >= 22 and m1==12 or d1 <=20 and m1==1 : z1 = "Capricorn"
elif d1 >= 21 and m1==1 or d1 <=20 and m1==2 : z1 = "Aquarius"
elif d1 >= 21 and m1==2 or d1 <=21 and m1==3 : z1 = "Pisces"

date2 = input().split()
d2 = int(date2[0])
m2 = mname.index(date2[1][:3]) + 1
y2 = int(date2[2])

if d2 >= 22 and m2==3 or d2 <=21 and m2==4 : z2 = "Aries"
elif d2 >= 22 and m2==4 or d2 <=21 and m2==5 : z2 = "Taurus"
elif d2 >= 22 and m2==5 or d2 <=21 and m2==6 : z2 = "Gemini"
elif d2 >= 22 and m2==6 or d2 <=21 and m2==7 : z2 = "Cancer"
elif d2 >= 22 and m2==7 or d2 <=21 and m2==8 : z2 = "Leo"
elif d2 >= 22 and m2==8 or d2 <=21 and m2==9 : z2 = "Virgo"
elif d2 >= 22 and m2==9 or d2 <=21 and m2==10 : z2 = "Libra"
elif d2 >= 22 and m2==10 or d2 <=21 and m2==11 : z2 = "Scorpio"
elif d2 >= 22 and m2==11 or d2 <=21 and m2==12 : z2 = "Sagittarius"
elif d2 >= 22 and m2==12 or d2 <=20 and m2==1 : z2 = "Capricorn"
elif d2 >= 21 and m2==1 or d2 <=20 and m2==2 : z2 = "Aquarius"
elif d2 >= 21 and m2==2 or d2 <=21 and m2==3 : z2 = "Pisces"

days_in_feb1 = 28
if y1 % 400 == 0 or y1 % 100 != 0 and y1%4 == 0 :
    days_in_feb1 = 29

days_in_feb2 = 28
if y2 % 400 == 0 or y2 % 100 != 0 and y2%4 == 0 :
    days_in_feb2 = 29

days_in_m1 = 31
if m1==4 or m1==6 or m1==9 or m1==11 :
    days_in_m1 = 30
elif m1==2 :
    days_in_m1 = days_in_feb1

# เริ่มหาจำนวนวันตั้งแต่ d1,m1,y1 ถึง d2,m2,y2 วิธีที่เขียนอาจผิดพลาดเล็กน้อย ไม่ต้องกังวล
days = 0
if m1 < 12 : days += 31
if m1 < 11 : days += 30
if m1 < 10 : days += 31
if m1 < 9 : days += 30
if m1 < 8 : days += 31
if m1 < 7 : days += 31
if m1 < 6 : days += 30
if m1 < 5 : days += 31
if m1 < 4 : days += 30
if m1 < 3 : days += 31
if m1 < 2 : days += days_in_feb1

if m2 > 1 : days += 31
if m2 > 2 : days += days_in_feb2
if m2 > 3 : days += 31

```

```

if m2 > 4 : days += 30
if m2 > 5 : days += 31
if m2 > 6 : days += 30
if m2 > 7 : days += 31
if m2 > 8 : days += 31
if m2 > 9 : days += 30
if m2 > 10 : days += 31
if m2 > 11 : days += 30

days += (days_in_m1 - d1 + 1) + int((y2 - y1 - 1)*365.25) + (d2 - 1)
print(z1, z2)
print(days)

```

โปรแกรมข้างบนนี้มี code ซ้ำกันพอควร จึงปรับใหม่โดยใช้ฟังก์ชันตามโครงของโปรแกรมข้างล่างนี้

```

def read_date(): # อ่านวันเดือนปีคั่นด้วยช่องว่าง เดือนเป็นชื่อเดือน คินลิสต์ เลขวัน เดือน ปี

def zodiac(d,m): # คินชื่อราศีของวัน d เดือน m

def days_in_feb(y): # คินจำนวนวันของเดือนกุมภาพันธ์ในปี y

def days_in_month(m,y): # คินจำนวนวันของเดือน m ในปี y

def days_in_between(d1,m1,y1,d2,m2,y2):
    # คินจำนวนวันตั้งแต่วันเดือนปี d1,m1,y1 ถึง d2,m2,y2

def main() :
    d1,m1,y1 = read_date()
    d2,m2,y2 = read_date()
    # แสดง ราศีของ d1,m1,y1 กับ ของ d2,m2,y2 บรรทัดเดียวกัน คั่นด้วยช่องว่าง
    # แสดงจำนวนวันตั้งแต่ d1,m1,y1 ถึง d2,m2,y2

exec(input().strip()) # ต้องมีบรรทัดนี้เมื่อส่งไป grader

```

ข้อมูลนำเข้า

คำสั่งในการทดสอบฟังก์ชันที่เขียน

ข้อมูลส่งออก

ผลที่ได้จากคำสั่งที่ป้อนเป็นข้อมูลนำเข้า

ตัวอย่าง

input	output (ทางจอภาพ)
<code>print(read_date())</code> 1 Jan 2017	[1, 1, 2017]
<code>print(zodiac(2,9))</code>	Virgo
<code>print(days_in_feb(2016))</code>	29
<code>print(days_in_month(2,2017))</code>	28
<code>print(days_in_between(1,1,2016, 1,1,2017))</code>	366
<code>main()</code> 1 Jan 2015 2 Aug 2017	Capricorn Leo 943