

SOLUCIONES EJERCICIOS



2ª PARTE TEMA 1:

INSTRUCCIONES CONDICIONALES

Mª Ángeles Fdez de Sevilla Vellón

2ª PARTE TEMA1:

SOLUCIONES I.CONDICIONALES

Número 1. Este programa lee la calificación numérica de un ejercicio (de 0 a 10) y devuelve su calificación alfabética.

```
# Archivo: ejercicio1_20.py
# Autor: XXXXX XXXXXXXX (nombre y apellidos)
# Fecha: DD/MM/AAAA
# Descripción: Lee calificación numérica y devuelve la
# calificación alfabética.

# Programa principal
nota = float(input('Introduce la nota: '))
if (nota < 0) or (nota > 10):
    print('Error')
elif nota < 5:
    print ('Un', nota, 'equivale a un SUSPENSO')
elif nota <7:
    print ('Un', nota, 'equivale a un APROBADO')
elif nota<9:
    print ('Un', nota, 'equivale a un NOTABLE')
else :
    print ('Un', nota, 'equivale a un SOBRESALIENTE')
```

Número 2. Este programa lee dos caracteres y averigua si están ordenados por su código ascii. Además, si el primero de ellos es una cifra, comprueba si es impar y si es primo.

```
# Archivo: ejercicio1_21.py
# Autor: XXXXX XXXXXXXX (nombre y apellidos)
# Fecha: DD/MM/AAAA
# Descripción: Lee dos caracteres y averigua si están
# ordenados por su código ascii y si el primero es una
# cifra comprueba si es impar y primo.

# Programa principal
dato1 = str(input('Escribe un caracter: '))
dato2 = str(input('Escribe otro caracter: '))

if dato1 < dato2:
    print ('Están ordenados por su código ASCII:')
else:
    print ('No están ordenados por su código ASCII: ')
```

```
if (dato1 >='0') and (dato1 <='9'):
    print('El dato1 es un número')
    dat1 = ord(dato1)-ord('0')
    if (dat1 % 2)== 0 :
        print ('El número ', dat1, ' es PAR')
    else:
        print ('El número ', dat1, ' es IMPAR')
    if (((dat1 % 2)!= 0) and (dat1 != 9)) or (dat1 == 2):
        print ('El número es primo')
    else:
        print ('El número no es primo')
else:
    print('El dato1 no es un número')
```

Número 3. Este programa pide una fecha (día,mes,año) y calcula la edad correspondiente.

```
# Archivo: ejercicio1_22.py
# Autor: XXXXX XXXXXXXX (nombre y apellidos)
# Fecha: DD/MM/AAAA
# Descripción: Pide fecha y calcula la edad.

# Llamada a la función datetime para obtener hora del stma
from datetime import datetime

# Programa principal
a = datetime.now() # Obtiene fecha y hora actual
año = int(input('Dime el año: '))
mes = int(input('Dime el mes: '))
dia = int(input('Dime el día: '))
edad = a.year-año
if ((a.month<mes) or ((a.month==mes) and (a.day<dia))):
    edad = edad-1
print(edad)
```

Número 4. Este programa determina si un año es o no bisiesto.

NOTA: ej: 1996 es bisiesto, 2000 es bisiesto pero 1800 no lo es.

```
# Archivo: ejercicio1_23.py
# Autor: XXXXX XXXXXXXX (nombre y apellidos)
# Fecha: DD/MM/AAAA
# Descripción: Determina si un año es bisiesto o no.

# Programa principal
anio= int(input('Introduzca el año (AAAA) => '))

if (anio % 4) == 0:
    if (anio % 100 == 0) and (anio % 400 != 0):
```

```

        bisiesto = False
    else: bisiesto = True
else: bisiesto = False
print('El año ', anio)
if bisiesto:
    print(' es bisiesto')
else: print(' no es bisiesto')

```

Número 5. Este programa pide un mes y un año y calcula el número de días del mes.

NOTA: Aunque como podemos comprobar Python interpreta la prioridad de los operandos, en general para que el código sea más claro es conveniente escribir paréntesis.

```

# Archivo: ejercicio1_24.py
# Autor: XXXXX XXXXXXXX (nombre y apellidos)
# Fecha: DD/MM/AAAA
# Descripción: Pide mes y año y calcula el número de días
# del mes.

# Programa principal
a = int(input('Ingresa año\n'))
m = int(input('Ingresa mes\n'))
if (m<1) or (m>12):
    print('; Valor fuera de rango !')
else:
    if m in(4,6,9,11):
        num_dias = 30
    elif m ==2:
        #forma más compacta de determinar si un año es bisiesto
        if((a % 4 == 0)and((a % 100!=0)or (a % 400 == 0))):
            num_dias = 29
        else: num_dias = 28
    else: num_dias = 31
print('El mes ',m,' del año ', a, ' tiene ', num_dias, ' días')

```

Número 6. Este programa solicita las coordenadas de un punto en el plano y lo ubica.

```

# Archivo: ejercicio1_25.py
# Autor: XXXXX XXXXXXXX (nombre y apellidos)
# Fecha: DD/MM/AAAA
# Descripción: Solicita las coordenadas de un punto y lo
# ubica.

# Programa principal
print('Introduzca las coordenadas del punto (X Y) => ');
x = int(input('Introduce coordenada x: '))

```

```
y = int(input('Introduce coordenada y: '))
print('El punto está en: ');
if x*y == 0: # el punto está en un eje o es el origen
    if x==0:
        if y>0:
            print('la parte positiva del eje Y')
        elif y<0:
            print('la parte negativa del eje Y')

        else: print('el origen de coordenadas')
    else:
        if x>0:
            print('la parte positiva del eje X')
        else:
            print('la parte negativa del eje X')
else:
    if x*y>0: # x e y tienen mismo signo
        if x>0:
            print('el primer cuadrante')
        else:
            print('el tercer cuadrante')
    else: # x e y tienen distinto signo
        if x>0:
            print('el cuarto cuadrante')
        else:
            print('el segundo cuadrante');
```

Número 7. Este programa pide las coordenadas de tres puntos en el plano y averigua cuales están más próximos.

```
# Archivo: ejercicio1_26.py
# Autor: XXXXX XXXXXXXX (nombre y apellidos)
# Fecha: DD/MM/AAAA
# Descripción: Pide las coordenadas de 3 puntos en el plano
# y averigua cuales están más próximos.

# Llamada a la librería math entera
from math import *

# Programa principal
x1 = int(input('Introduce coordenada x1: '))
y1 = int(input('Introduce coordenada y1: '))
x2 = int(input('Introduce coordenada x2: '))
y2 = int(input('Introduce coordenada y2: '))
x3 = int(input('Introduce coordenada x3: '))
y3 = int(input('Introduce coordenada y3: '))

dist12 = sqrt((x1-x2)**2 + (y1-y2)**2)
dist13 = sqrt((x1-x3)**2 + (y1-y3)**2)
```

```

dist23 = sqrt((x2-x3)**2 + (y2-y3)**2)

if (dist12 > dist13) or (dist12 > dist23):

    if dist13 > dist23:
        print ('El 2° y el 3° están mas cerca')
    else:
        print ('El 1° y el 3° están mas cerca')

else:

    print ('El 1° y el 2° están mas cerca')

```

Número 8. Este programa pide una hora en notación de 24 horas y la devuelve en notación de 12 horas.

```

# Archivo: ejercicio1_27.py
# Autor: XXXXX XXXXXXXX (nombre y apellidos)
# Fecha: DD/MM/AAAA
# Descripción: Pide una hora en notación 24h y la devuelve
# en notación 12h.

# Constantes
am = 'AM'      # (* antes de mediodía *)
pm = 'PM'      # (* después de mediodía *)

# Programa principal
hora = int(input('Introduzca una hora: '))
minutos = int(input('Introduzca minutos: '))

if ((hora<0)or(hora>24))or((minutos<0)or(minutos>60)):
    print('La hora introducida no es correcta! ')
else:
    if hora >= 12:
        hora = hora - 12
        mediodia = pm
    else:
        mediodia = am

print('En notación 12 horas es:',hora,':',minutos,' ',
mediodia)

```

Número 9. Este programa emite facturas de una empresa de alquiler de automóviles.

```

# Archivo: ejercicio1_28.py
# Autor: XXXXX XXXXXXXX (nombre y apellidos)
# Fecha: DD/MM/AAAA
# Descripción: Programa que emite facturas de una empresa

```

```
# de alquiler de coches.

# Constantes
Limite_inf = 300
Limite_sup = 1000
Cantidad_fija = 120#cantidad fija para kms<= que limite_inf
Precio1 = 5# cnts/km para kms entre limite_inf y limite_sup
Precio2 = 3 # cnts/km para kms que sobrepasen limite_sup

# Programa principal
cuentakms_inicial=int(input('Introduzcauentakms inicial: '))
cuentakms_final= int(input('Introduzca cuentakms final: '))

kilometros = cuentakms_final - cuentakms_inicial

if kilometros <= 0:
    print('Los datos introducidos no son correctos ')
else:
    if kilometros <= Limite_inf:
        suplemento = 0
    else:
        if kilometros <= Limite_sup:
            suplemento = (kilometros - Limite_inf)*Precio1
        else:
            suplemento = (Limite_sup - Limite_inf)*Precio1
            + (kilometros - Limite_sup)*Precio2

print('El importe total es: ', 100*(Cantidad_fija +
suplemento) , ' euros')
```

Número 10. Este programa calcula las raíces de una ecuación de segundo grado.

$$ax^2+bx+c=0$$

```
# Archivo: ejercicio1_29.py
# Autor: XXXXX XXXXXXXX (nombre y apellidos)
# Fecha: DD/MM/AAAA
# Descripción: Calcula las raíces de una ecuación de 2º
# grado.

# Llamada a la librería math entera
from math import *

# Programa principal
a = int(input('Introduzca el coeficiente a: '))
b = int(input('Introduzca el coeficiente b: '))
c = int(input('Introduzca el coeficiente c: '))
```

```

print('La solución de la ecuación ', a,'x^2',' + ', b,'x',' '
+ ', c, ' = 0 es: ')

if a == 0:
    if b == 0:
        if c == 0:
            print('Cualquier número es solución ')
        else:
            print('No tiene solución ')
    else:
        print('Una raíz real única:', -c/b)
else:
    disc = b**2 - 4*a*c
    if disc == 0:
        print('Una raíz real doble: ', -b/(2*a))
    else:
        if disc > 0:
            x1 = (-b+sqrt(disc))/(2*a)
            x2 = (-b-sqrt(disc))/(2*a)
            print('Dos raíces reales: ', x1, ' y', x2)
        else:
            preal = -b/(2*a)
            pimag = sqrt(-disc)/(2*a)
print('Dos raíces complejas conjugadas: ', preal,' (+/-) ',
pimag,'i')

```

Número 11. Este programa lee un año en cifras arábicas (entre 1 y 2000) y lo convierte a cifras romanas.

```

# Archivo: ejercicio1_30.py
# Autor: XXXXX XXXXXXXX (nombre y apellidos)
# Fecha: DD/MM/AAAA
# Descripción: Lee un año en cifras arábicas y lo convierte
# a cifras romanas.

# Constantes
Anio_min = 1
Anio_max = 2000

# Programa principal
anio = int(input('Introduzca el año: '))

if (anio < Anio_min) or (anio > Anio_max):
    print('; Año fuera de rango !')
else:
    millar = anio // 1000;
    anio_cent = anio % 1000
    centena = anio_cent // 100
    anio_dece = anio_cent % 100
    decena = anio_dece // 10

```

```
unidad = anio_dece % 10
print('El año en números romanos es: ', end='')

if millar == 1:
    print('M', end='')
else millar == 2:
    print('MM', end='')

if centena == 1:
    print('C', end='')
elif centena == 2:
    print('CC', end='')
elif centena == 3:
    print('CCC', end='')
elif centena == 4:
    print('CD', end='')
elif centena == 5:
    print('D', end='')
elif centena == 6:
    print('DC', end='')
elif centena == 7:
    print('DCC', end='')
elif centena == 8:
    print('DCCC', end='')
else centena == 9:
    print('CM', end='')

if decena == 1:
    print('X', end='')
elif decena == 2:
    print('XX', end='')
elif decena == 3:
    print('XXX', end='')
elif decena == 4:
    print('XL', end='')
elif decena == 5:
    print('L', end='')
elif decena == 6:
    print('LX', end='')
elif decena == 7:
    print('LXX', end='')
elif decena == 8:
    print('LXXX', end='')
else decena == 9:
    print('XC', end='')

if unidad == 1:
    print('I', end='')
elif unidad == 2:
    print('II', end='')
```

```

elif unidad == 3:
    print('III', end='')
elif unidad == 4:
    print('IV', end='')
elif unidad == 5:
    print('V', end='')
elif unidad == 6:
    print('VI', end='')
elif unidad == 7:
    print('VII', end='')
elif unidad == 8:
    print('VIII', end='')
else unidad == 9:
    print('IX', end='')

```

Número 12. Este programa calcula el ángulo que forma la parte positiva del eje de las abscisas con el vector que une el origen de coordenadas con un punto dado.

```

# Archivo: ejercicio1_31.py
# Autor: XXXXX XXXXXXXX (nombre y apellidos)
# Fecha: DD/MM/AAAA
# Descripción: Calcula el ángulo formado entre la parte
# positiva del eje de abscisas con el vector que une el
# origen de coordenadas con un punto dado.

# Llamada a la librería math entera
from math import *

# Programa principal
x = int(input('Introduce coordenada x: '))
y = int(input('Introduce coordenada y: '))

if x == 0:
    if y > 0:
        angulo = pi/2
    else:
        if y < 0:
            angulo = 3*pi/2
        else:
            angulo = 0
else:
    # La función arctangente devuelve valor entre pi/2 y -pi/2
    #hemos de convertir su valor entre 0 y 2 pi

```

```
angulo = atan(y/x)
if x < 0:
    angulo = angulo + pi
else:
    if y < 0:
        angulo = angulo + 2*pi
print('El ángulo correspondiente es: ', angulo, 'radianes')
```

Número 13. Este programa pide dos números por teclado y una operación y devuelva el resultado de la misma.

```
# Archivo: ejercicio1_32.py
# Autor: XXXXX XXXXXXXX (nombre y apellidos)
# Fecha: DD/MM/AAAA
# Descripción: Pide dos números por teclado y una operación
# y devuelve el resultado de la misma.

# Programa principal
dato1 = int(input('Introduce un numero: '))
dato2 = int(input('Introduce otro numero: '))

operacion = str(input('Escribe "s" si quieres sumar, "r" si
quieres restar, "m" si quieres multiplicar y "d" si quieres
dividir: '))

if operacion == 's':
    print(dato1, '+ ', dato2, '= ', dato1 + dato2)
elif operacion == 'r':
    print(dato1, '- ', dato2, '= ', dato1 - dato2)
elif operacion == 'm':
    print(dato1, '* ', dato2, '= ', dato1 * dato2)
elif operacion == 'd':
    print(dato1, '/ ', dato2, '= ', dato1 / dato2)
else:
    print('Error')
```

