

Estructuras condicionales.

1. Dados dos números enteros $n1$ y $n2$ determinar e imprimir por consola si el primero es múltiplo del segundo.

```

5      # Inicializar variables.
6      n1 = n2 = 0
7      estatus = "" # Guarda si el valor es multiplo o no.
8      # Entrada de datos.
9      n1 = int(input("De un numero entero N1: "))
10     n2 = int(input("De un numero entero N2: "))
11     # Proceso
12     # Evaluamos si el n1 es multiplo de n2
13     if (n1 % n2 == 0):
14         estatus = str(n1) + " es multiplo de " + str(n2)
15     else:
16         estatus = str(n1) + " no es multiplo " + str(n2)
17     print("\nSalida Resultados")
18     print(estatus)

```

Solución:

Línea 6: Inicializamos las variables $n1$ y $n2$ como valores enteros (int).

Línea 7: Inicializamos la variable estatus de tipo cadena (String).

Líneas 9-10: se procede a leer desde el teclado los valores enteros $n1$ y $n2$.

Línea 13: Determinamos mediante un **if** si el resto de la división entre $n1$ y $n2$ es igual a cero, de cumplirse el mensaje indicara que $n1$ es múltiplo de $n2$, de no cumplirse se indicara la contrario. Es de hacer notar que en las líneas 14 y 16 para poder concatenar los valores de $n1$ y $n2$ con la cadena debemos previamente convertir estos valores en una cadena para lograr esto usamos la función **str()**.

2. Escriba un programa que reciba dos números enteros como entrada. Determine e imprima por consola un mensaje por consola Signos Opuestos solo si uno de los enteros es positivo y el otro negativo y Signos Iguales si ambos enteros tienen el mismo signo.

```

7      # Inicializar variables
8      num1 = num2 = 0
9      mensaje = ""
10     # Entrada de datos
11     num1 = int(input("De un numero entero positivo o negativo: "))
12     num2 = int(input("De otro numero entero positivo o negativo: "))
13     # Proceso
14     if ((num1<0) and (num2 >=0)) or ((num1>=0) and (num2<0)):
15         mensaje = "Signos Opuestos"
16     else:
17         mensaje = "Signos Iguales"
18     #Salida de resultados
19     print("\n\t\tLos enteros son de ",mensaje)

```

Solución:

Una vez inicializada las variables $num1$, $num2$ y $mensaje$, tipo entero (int) las dos primeras y de tipo cadena (String) la última, procedemos en la línea 14 verificar si los valores leídos son de signos opuestos, simplemente comprobamos si uno de los valores es menor que cero y el otro es mayor o igual a cero, aplicando los operadores lógicos AND y OR, si se cumple la condición se indicara en el mensaje son de signos opuestos en caso contrario de signos iguales.

3. Dado tres números enteros, determinar e imprimir cuál de los tres es el menor

```

4  # Inicializar variables
5  n1 = n2 = n3 = 0 # Variables de los tres numeros.
6  menor = 0
7  # Entrada de datos
8  n1 = int(input("De el valor de n1 'Entero': "))
9  n2 = int(input("De el valor de n2 'Entero': "))
10 n3 = int(input("De el valor de n3 'Entero': "))
11 # Proceso
12 if ((n1<n2) and (n1<n3)):
13     menor = n1
14 elif ((n2<n1) and (n2<n3)):
15     menor = n2
16 elif ((n3<n1) and (n3<n2)):
17     menor = n3
18 # Salida
19 print("\nSalida Resultados")
20 print("El menor de los tres es: ",menor)

```

Solución:

Entre las líneas 5 y 6 procedemos a inicializar las variables n1, n2, n3 de tipo entera y la variable menor, también entera, la cual almacenara el menor de los tres valores leídos.

Entre las líneas 12 y 17 determinamos cual es el menor, comparando uno de los valores con los otros, dos si se cumple la condición guardamos en **menor** el menor valor encontrado. En línea 12 n1 contra n2 y n3, de no cumplirse, en la línea 14 n2 contra n1 y n3 y de no cumplirse en la línea 16 n3 contra n1 y n2. Finalmente, en la línea 20 mostramos el resultado.

4. Escribir un programa que tenga como entrada la hora y los minutos que una persona comenzó a hacer una llamada telefónica y la hora y los minutos en que termino de hacerla, determine el tiempo que duro la llamada en horas y minutos y el costo a pagar por la misma sabiendo que por los primeros 10 minutos se cobraran 50 Bs y por cada minuto adicional a los 10 primeros minutos se cobran a 30.50 Bs el minuto. Consideraciones: 1 hora son 60 minutos.

```

10 # Inicializar variables
11 hi = mi = 0 # Hora y minutos iniciales
12 hf = mf = 0 # Hora y minutos finales
13 tmi = tmf = 0 # Total minutos iniciales y finales.
14 dif = 0 # Diferencia tiempo inicial y final
15 hd = md = 0 # Horas que duro la llamada, minutos que duro la llamada
16 costo = 0.0 # Costo de la llamada
17 # Entrada de datos
18 hi = int(input("Hora de inicio llamada: "))
19 mi = int(input("Minutos de inicio: "))
20 hf = int(input("Hora de fin llamada: "))
21 mf = int(input("Minutos de fin: "))
22 # Proceso
23 # Tiempo de inicio lo llevamos todo a minutos
24 tmi = hi * 60 + mi
25 # Tiempo de fin lo llevamos todo a minutos
26 tmf = hf * 60 + mf

```

```

27 # Calculamos la duracion de la llamada
28 dif = tmf - tmi
29 # Llevamos la duracion a horas y minutos
30 hd = dif // 60
31 md = dif % 60
32 # Calculamos el costo de la llamada
33 if dif <= 10:
34     costo = 50
35 else:
36     costo = 50 + (dif - 50) * 30.5
37 # Salida
38 print("\nSalida de resultados")
39 print("La llamada duro ", hd, " horas y ", md, " minutos")
40 print("El costo de la misma fue de: ", costo, " Bs.")

```

Solución:

Entre las líneas 11 y 21 procedemos a inicializar las variables requeridas por el programa y a efectuar la lectura de los datos requeridos.

En las líneas 24 y 26 llevamos a minutos el tiempo se inicio la llamada y el tiempo en que termino la llamada con la finalidad de trabajar en una sola unidad de tiempo, en este caso a minutos.

En la línea 28 calculamos la diferencia entre el tiempo final y el inicial, con lo cual determinamos la duración de la llamada.

En las líneas 30 y 31 llevamos el total de minutos que duró la llamada a horas y minutos, por ejemplo, si la duración fue de 7234 minutos la división entera (//) seria 7234//60 → 2 horas y el resto (%) de la división entera seria 7234%60 → 34 por lo tanto la llamada duro 2 horas y 34 minutos.

Entre las líneas 33 y 36 determinamos el costo de la llamada, verificando si la duración fue menor o igual a los diez minutos, en este caso el costo será de 50 Bs, en caso contrario al costo por los primeros 10 minutos le agregamos el costo generado por los minutos adicionales.

Entre la línea 38 y 40 mostramos los resultados, observe que en la cadena de la línea 38 hemos incluido \n, que provoca un salto de línea antes de imprimir la cadena.

5. Escriba un programa que tenga como entrada el peso de una persona. Determine e imprima por consola el mensaje que corresponda según la siguiente tabla:

Peso entre	Mensaje
00-40	Flaco
41-60	Delgado
61-80	Rellenito
81 o más	Gordo

```

11 # Inicializar Variables
12 peso = 0.0
13 mensaje = ""
14 # Entrada de datos
15 peso = float(input("De el peso de la persona: "))
16 # Proceso
17 if peso <= 40:
18     mensaje="Flaco"
19 elif peso <= 60:
20     mensaje = "Delgado"
21 elif peso<=80:
22     mensaje = "Rellenito"
23 else:
24     mensaje = "Gordo"
25 # Salida resultados
26 print("\n\tDe acuerdo a su peso su condicion es: ",mensaje)

```

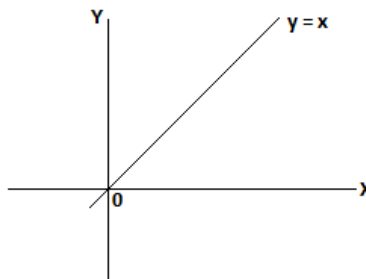
Solución:

En las líneas 11, 12 y 15 inicializamos las variables requeridas y procedemos a la lectura de los datos.

Entre las líneas 17 y 24 determinamos el estatus de la persona de acuerdo a su peso, comenzando en la 17 donde verificamos si el peso es menor o igual a los 40 Kilos de cumplirse el mensaje seria Flaco de no cumplirse la persona pesa mas de 40 kilos, en la 19 verificamos si la persona esta entre los 41 y los 60 kilos, si se cumple el mensaje seria Delgado, de no ser así la persona pesa mas de 60 kilos, en la línea 21 verificamos si el peso de la persona esta entre los 61 y 80 kilos, de ser así el mensaje seria Rellenito, de no ser así no quedaría otra alternativa que es Gordo.

En la línea 26 mostramos el mensaje, es de hacer notar que en la cadena del **print** usamos los caracteres especiales `\n` y `\t` el primero provoca un salto de línea y el segundo un desplazamiento del cursor 4 espacios en blanco a la derecha como tenemos doble `\t` el desplazamiento será de 8 blancos. Visualice esto al momento de ejecutar el programa.

6. Dada la recta de ecuación $y = x$, determine e imprima si un punto de coordenadas x e y esta sobre la recta, encima de la recta o por debajo de la recta.



```

6  # Inicializar variables
7  x = y = 0
8  # Entrada de datos
9  x = float(input("De el valor de x: "))
10 y = float(input("De el valor de y: "))
11 # Proceso
12 if x == y:
13     estatus = "SOBRE"
14 elif x > y:
15     estatus = "ENCIMA"
16 elif x < y:
17     estatus = "POR DEBAJO"
18 print("Salida Resultados")
19 print("El punto esta ", estatus, " la recta")

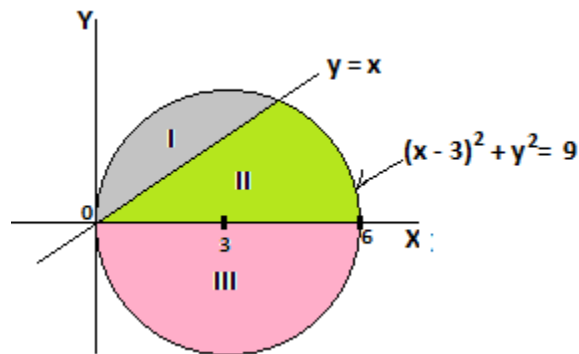
```

Solución:

Entre las líneas 7 y 10 inicializamos las variables y leemos las coordenadas del punto x e y .

Entre las líneas 12 y 16 verificamos la posición del punto con respecto a la línea si las coordenadas son iguales el punto esta sobre la línea, si no cumple verificamos si $x > y$, en este caso el punto esta por encima de la recta, de no ser así verificamos si $x < y$ entonces el punto esta por debajo de la recta.

7. Escriba un programa que lea las coordenadas de un punto. Determine e imprima por consola si el punto está en el área I o en el área II o en el área III o no está en ninguna de las áreas, el punto no puede estar ni en el borde de la circunferencia ni sobre la recta ni el en eje x. Ver figura



```

5      # Inicializar variables.
6      x = y = 0.0 # Coordenadas del punto.
7      radio = 0.0
8      area = ""
9      # Entrada de datos.
10     x = float(input("De el valor de x: "))
11     y = float(input("De el valor de y: "))
12     # Proceso
13     radio = (x - 3) ** 2 + y ** 2
14     if radio < 9:
15         if y > x:
16             area = "Esta en el area I"
17         elif y < x and y > 0:
18             area = "Esta en el area II"
19         elif y < 0:
20             area = "Esta en el area III"
21         else:
22             area = "No Esta en ninguna de las area"
23     else:
24         area = "No Esta en ninguna de las area"
25     print("\nSalida de Resultados")
26     print_(area)

```

Solución:

Entre las líneas 6 y 11 inicializamos las variables requeridas y leemos las coordenadas del punto que queremos ubicar en la gráfica.

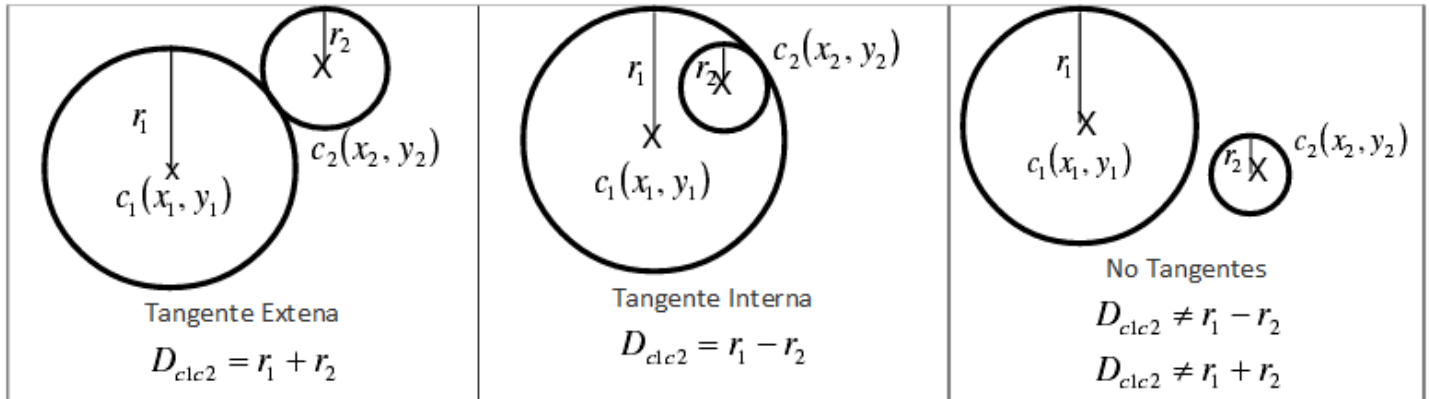
En la línea 13 calculamos el radio generado por las coordenadas del punto leído de acuerdo a la ecuación de la circunferencia dada.

En la línea 14 verificamos si la circunferencia cuyo radio calculamos previamente este dentro de la circunferencia dada, si se cumple la condición el punto esta dentro de la circunferencia de ecuación $(x - 3)^2 + y^2 = 9$, de no cumplirse esto el punto no está en ninguna de las áreas.

Ahora si el punto está dentro de la circunferencia hay que determinar en qué área de la misma está el punto. Esto lo determinamos entre las líneas 15 y 22, si $x > y$, esta en el área I, de no cumplirse evaluamos $y < x$ and $y > 0$, si se cumple esta en el área II, de no cumplirse evaluamos $y < 0$, si se cumple el punto esta en el área III, de no cumplirse ninguna de las anteriores el punto estaría sobre la recta $y = x$ o sobre el eje x.

Entre las líneas 25 y 26 mostramos los resultados.

8. Escriba un programa que tenga como entrada las coordenadas de los centros de dos circunferencias y sus respectivos radios. Determine e imprima un mensaje que indique si su relación tangencial es: Tangencial Externa, Tangencial interna, No son tangenciales, Son iguales.



Recuerde: que la distancia entre dos puntos se determina como: $Dist = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$

Para la solución requerimos importar la funciones **fabs()** (Valor absoluto) y **sqrt()** (Raíz cuadrada).

```

7   from math import fabs, sqrt
8
9   # Inicializar variables.
10  r1 = r2 = 0.0 # Radios de las circunferencias.
11  x1 = x2 = y1 = y2 = 0.0 # Coordenadas centros circunferencias.
12  r = x = y = 0.0
13  dist = 0.0 # Distancia centros circunferencias.
14  mensaje = ""
15  # Entrada de los datos.
16  # Datos circunferencia 1:
17  r1 = float(input("De le radio de la circunferencia 1: "))
18  x1 = float(input("De el valor de x: "))
19  y1 = float(input("De el valor de y: "))
20  # Datos de la circunferencia 2:
21  r2 = float(input("De le radio de la circunferencia 2: "))
22  x2 = float(input("De el valor de x: "))
23  y2 = float(input("De el valor de y: "))
24  # Calculamos la distancia entre los radios
25  dist = sqrt((x2 - x1) ** 2 + (y2 - y1) ** 2)
26  print(dist)
27  # Determinamos su relacion Tangencial
28  if (dist == 0):
29      mensaje = "Son Iguales"
30  elif (dist == fabs(r1 - r2)):
31      mensaje = "Tangencial Interna"
32  elif (dist == r1 + r2):
33      mensaje = "Tangencial Externa"
34  else:
35      mensaje = "No son Tangenciales"
36  # Salida resultados
37  print("\n\tLa relacion entre las dos circunferencia es: ", mensaje)

```

Solución:

Línea 7: nos permite importa las funciones matemáticas **fabs** y **sqrt**, de la forma en que lo hemos hecho no tenemos que anteponer el nombre de la librería **math** al nombre de la función.

Entre las líneas 10 y 23 inicializamos las variables requeridas y leemos la data necesaria para el programa.

En la línea 24 calculamos la distancia entre los puntos que definen los centros de las circunferencias.

Entre las líneas 28 y 35 determinamos la relación tangencial que hay entre las dos circunferencias. Si la distancia entre los radios es igual a cero implica que las dos circunferencias tienen los radios iguales, de no cumplirse esta condición evaluamos si la distancia es igual al valor absoluto de la diferencia entre los radios de las dos circunferencias la relación es tangencial interna (ver la gráfica), de no cumplirse esta condición evaluamos si la distancia es igual a la suma de los radios si se cumple la relación es tangencial externa, en caso de no cumplirse ninguna de las anteriores no hay ninguna relación tangencial.

Línea 37 muestra el resultado de evaluar las circunferencias.

Datos de prueba:

Radio 1	X ₁	Y ₁	Radio 2	X ₂	Y ₂	Relación
4.0	0.0	0.0	2.0	0.0	2.0	Externa
2.0	0.0	0.0	5.0	7.0	0.0	Interna
6.0	1.0	2.0	4.5	7.0	2.0	Ninguna
2.0	1.0	1.0	2.0	1.0	1.0	Iguales

