

UVA 3: Condicionales

22 de marzo de 2023

1. Resolver ecuación.

Escriba un programa que resuelva la ecuación $ax + b = 0$, para a y b ingresados como entrada por el usuario. Se debe validar si tiene solución, es decir, si $a \neq 0$.

2. Círculo.

Dado un círculo (coordenadas del centro y radio) y un punto cualquiera (coordenadas), determinar si el punto está dentro, fuera o justo en la circunferencia del círculo.

3. Ecuación de segundo grado.

Implemente un programa que resuelva la ecuación de segundo grado $ax^2 + bx + c = 0$, para a , b y c ingresados por el usuario. Guíese por los siguientes ejemplos:

a: 2	a: -2	a: 1
b: 4	b: 3	b: 1
c: 2	c: 2	c: 1
La solución es: -1.0	Solución 1: -0.5	No tiene soluciones reales.
	Solución 2: 2.0	

Nota: recuerde que el discriminante está dado por $b^2 - 4ac$

4. Horas extra.

Una empresa paga de manera diferenciada las horas extra de sus empleados, es decir, las que trabajan después de las 44 que forman su jornada. Existe un tope en el número de horas extra: la empresa paga únicamente las primeras 10. El monto a recibir por cada hora extra depende del tipo de empleado: los de tipo 1 reciben \$5.000 por hora; los de tipo 2 reciben \$8.000; y los de tipo 3 reciben \$10.000. Los empleados de tipo 4 o superior no tienen derecho al pago de horas extra.

Escriba un programa que reciba como entrada el tipo de empleado y la cantidad de horas que trabajó en la semana, y entregue como salida el monto que recibirá por concepto de horas extra. Guíese por los siguientes ejemplos:

Tipo de empleado: 2	Tipo de empleado: 3	Tipo de empleado: 5
Horas semana: 44	Horas semana: 46	Horas semana: 46
\$ 0 (no tiene extras)	A pagar: \$ 20000	\$ 0 (sin derecho)
Tipo de empleado: 2	Tipo de empleado: 2	
Horas semana: 46	Horas semana: 60	
A pagar: \$ 16000	A pagar: \$ 80000	

5. Número mayor.

Escriba un programa que pida al usuario ingresar 3 números enteros, y que luego imprima en pantalla el mayor de todos ellos. Su solución debe utilizar instrucciones `if` anidadas, es decir, instrucciones `if` dentro de otras instrucciones `if`.

```
Ingrese n1: 7
Ingrese n2: 10
Ingrese n3: 3
El mayor es: 10
```

Nota: La solución de este problema le servirá para el siguiente ejercicio.

6. Tipo de triángulo: ángulos.

Construya un programa que determine el tipo de triángulo según la relación de sus ángulos. Lo primero será solicitar al usuario que ingrese α_1 , α_2 y α_3 (puede asumir que siempre ingresará enteros positivos). Se debe verificar que es un triángulo válido, es decir, que se cumple $\alpha_1 + \alpha_2 + \alpha_3 = 180$. En caso contrario, se imprime el mensaje **Triángulo inválido**.

El segundo paso será encontrar el valor del ángulo mayor, el cual llamaremos c . A los dos ángulos restantes los llamaremos a y b (no importa cuál es cuál, sólo nos interesará cuánto vale la suma de ambos). Cuando $a + b = c$, el triángulo es rectángulo. Cuando $a + b < c$, el triángulo es obtusángulo. Finalmente, cuando $a + b > c$, el triángulo es acutángulo. A continuación se muestran ejemplos de ejecución del programa:

Ingrese el primer ángulo: 60	Ingrese el primer ángulo: 60
Ingrese el segundo ángulo: 30	Ingrese el segundo ángulo: 40
Ingrese el tercer ángulo: 90	Ingrese el tercer ángulo: 90
El triángulo es rectángulo	Triángulo inválido
Ingrese el primer ángulo: 70	Ingrese el primer ángulo: 120
Ingrese el segundo ángulo: 60	Ingrese el segundo ángulo: 40
Ingrese el tercer ángulo: 50	Ingrese el tercer ángulo: 20
El triángulo es acutángulo	El triángulo es obtusángulo

7. Tipo de triángulo: lados.

Los tres lados a , b y c de un triángulo deben satisfacer la desigualdad triangular: cada uno de los lados no puede ser más largo que la suma de los otros dos.

Escriba un programa que reciba como entrada los tres lados de un triángulo e indique:

- si acaso el triángulo es inválido.
- si no lo es, qué tipo de triángulo es (escaleno, isósceles o equilátero).

```
Ingrese a: 3.9
Ingrese b: 6.0
Ingrese c: 1.2
No es un triangulo valido.
```

```
Ingrese a: 3.0
Ingrese b: 5.0
Ingrese c: 4.0
El triangulo es escaleno.
```

```
Ingrese a: 2.0
Ingrese b: 1.9
Ingrese c: 2.0
El triangulo es isoceles.
```

```
Ingrese a: 3.14
Ingrese b: 3.14
Ingrese c: 3.14
El triangulo es equilatero.
```

8. Paracaidista.

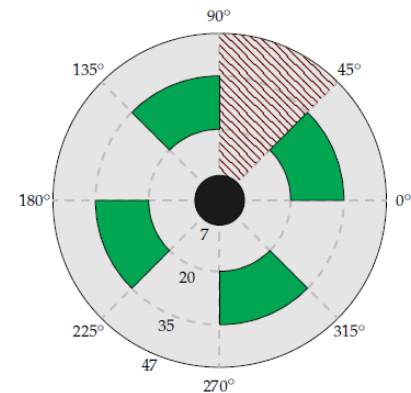
[25 %] Para celebrar el aniversario de la fundación de Kreisdorf, la alcaldesa ha organizado un espectáculo aéreo, que finalizará con el aterrizaje de un paracaidista en la plaza del pueblo.

La plaza es circular, y su radio es de 47 metros. En su centro hay una pileta (●) de 7 metros de radio. Además, hay cuatro áreas verdes (■), en las posiciones indicadas en la figura; los radios interno y externo de las áreas verdes son, respectivamente, 20 y 35 metros. El resto de la superficie es cemento.

El público que irá a presenciar el espectáculo se ubicará en el sector de la plaza indicado en la figura (▨).

El paracaidista ha encargado el desarrollo de un programa que le permita planificar su trayectoria para evitar caer en un lugar peligroso. El punto de aterrizaje está definido por dos valores: su distancia al centro de la plaza y el ángulo con respecto a las coordenadas indicadas en la figura.

Escriba un programa que, a partir de la distancia y el ángulo de aterrizaje, le indique al paracaidista donde caerá. El programa debe imprimir PILETA, AREA VERDE, PUBLICO, CEMENTO o FUERA DE LA PLAZA. Si el paracaidista aterriza justo en la frontera entre dos regiones, elija cualquiera de ellas.



Distancia: 25.4
Angulo: 192.9
AREA VERDE

Distancia: 29.1
Angulo: 333.0
CEMENTO

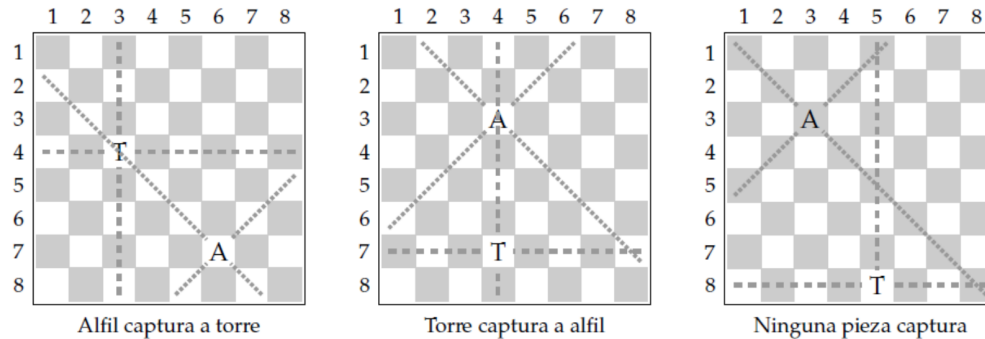
Distancia: 40
Angulo: 60
PUBLICO

Dado el enunciado anterior, complete el siguiente código, de manera que se produzcan las salidas correctas al programa planteado:

```
1 d = float(input('Distancia: '))
2 a = float(input('Ángulo: '))
3
4 if _____:
5     print("FUERA DE LA PLAZA")
6 else:
7     if _____:
8         print('PILETA')
9     elif _____:
10        print("PUBLICO")
11    elif _____:
12        print("AREA VERDE")
13    else:
14        print("CEMENTO")
```

9. Ajedrez

Un tablero de ajedrez es una grilla de ocho filas y ocho columnas, numeradas de 1 a 8. Dos de las piezas del ajedrez son el alfil y la torre. El alfil se desplaza en diagonal, mientras que la torre se desplaza horizontal o verticalmente. Una pieza puede ser capturada por otra si está en una casilla a la cual la otra puede desplazarse, tal como se muestra en el siguiente ejemplo:



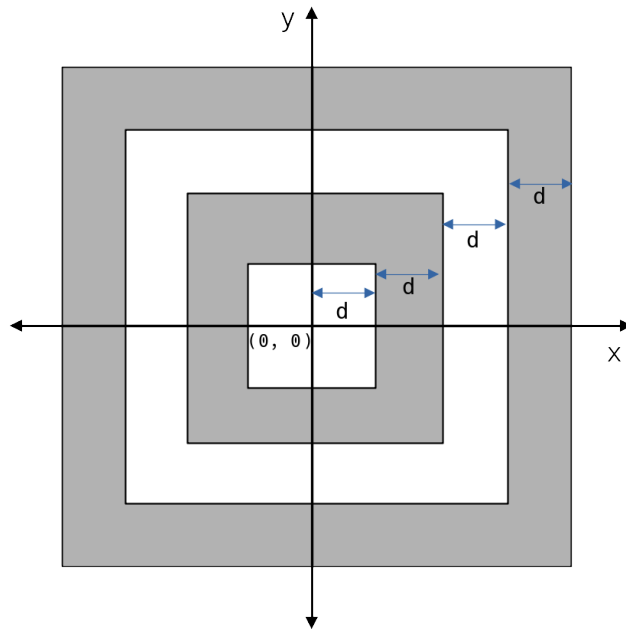
Escriba un programa que reciba como entrada las posiciones de un alfil y de una torre en el tablero, e indique cuál pieza captura a la otra (suponga que todos los datos ingresados son válidos). A continuación se muestra un ejemplo de la ejecución deseada:

Fila alfil: 7 Columna alfil: 6 Fila torre: 4 Columna torre: 3 Alfil captura	Fila alfil: 3 Columna alfil: 4 Fila torre: 7 Columna torre: 4 Torre captura	Fila alfil: 3 Columna alfil: 3 Fila torre: 8 Columna torre: 5 Ninguna captura
---	---	---

10. Videojuego.

En un juego de video hay que dispararle a un tablero de tiro al blanco que se encuentra dado por cuatro áreas cuadradas concéntricas separadas por una distancia d (igual para todas las áreas). Suponga que el tiro al blanco se encuentra en un plano cartesiano de dos dimensiones y cuyo centro $(0, 0)$ es el centro mismo del tiro al blanco.

Si las áreas concéntricas se enumeran desde las más interna hasta la más externa de 1 a 4, realice un programa en Python que retorne el número del área en la cual ha caído el disparo. El usuario debe ingresar la distancia d y el punto (x, y) .



Ingrese distancia: 5
 Ingrese x: 3
 Ingrese y: 4
 Área 1

Ingrese distancia: 1
 Ingrese x: 6
 Ingrese y: 6
 Fuera del área

Ingrese distancia: 4
 Ingrese x: -5
 Ingrese y: 10
 Área 3