

Actividad 2

Daniel Esteban Alvarez Fuentes

daalvarezf@unal.edu.co

David Gil Rúa

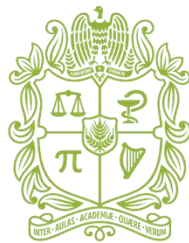
dgilr@unal.edu.co

Universidad Nacional de Colombia

Facultad de Minas

Programación Orientada a Objetos

Profesor: Walter Arboleda



UNIVERSIDAD
NACIONAL
DE COLOMBIA

Medellín, Colombia

14 de marzo de 2024

Parte 1:

Capítulo 3: Estructura secuencial

18. Se tiene la siguiente información de un empleado:

- · código del empleado,
- · nombres,
- · número de horas trabajadas al mes,
- · valor hora trabajada,
- · porcentaje de retención en la fuente.

Haga un algoritmo que muestre: código, nombres, salario bruto y salario neto.

Se tiene la siguiente información de un empleado:

>· código del empleado

>· nombres

>· número de horas trabajadas al mes

>· valor hora trabajada

>· porcentaje de retención en la fuente

Haga un algoritmo que muestre: código, nombres, salario bruto y salario neto.

```
def mostrar_datos(cod, nom, htm, ht, rf):  
    sb = htm * ht  
    sn = sb - (sb * rf)  
    print(f"El código del empleado es: {cod}")  
    print(f"El nombre del empleado es: {nom}")  
    print(f"El salario bruto es: $ {sb}")  
    print(f"El salario neto es: $ {sn}")
```

```
mostrar_datos(1, "Juan", 100, 40, 0.1)
```

El código del empleado es: 1

El nombre del empleado es: Juan

El salario bruto es: \$ 4000

El salario neto es: \$ 3600.0

19. Dado el valor del lado en un triángulo equilátero, haga un algoritmo que obtenga el perímetro, el valor de la altura y el área del triángulo.

```
from math import sqrt
```

```
import numpy as np
def informacion_triangulo(lado):
    altura = (sqrt(3) / 2) * lado
    perimetro = lado * 3
    area = (lado * altura) / 2
    print(f"La altura es: {altura} cm")
    print(f"El perímetro es: {perimetro} cm")
    print(f"El área es: {area} cm^2")
```

```
informacion_triangulo(8)
```

```
La altura es: 6.928203230275509 cm
```

```
El perímetro es: 24 cm
```

```
El área es: 27.712812921102035 cm^2
```

21. Dados los tres lados de un triángulo, haga un algoritmo que encuentre: perímetro, semiperímetro y el área del triángulo.

```
def informacion_triangulo2(lado1, lado2, lado3):
    perimetro = lado1 + lado2 + lado3
    semiperimetro = perimetro / 2
    area = sqrt(semiperimetro * (semiperimetro - lado1) * (semiperimetro - lado2) * (semiperimetro - lado3))
    print(f"El perímetro es: {perimetro} cm")
    print(f"El semiperímetro es: {semiperimetro} cm")
    print(f"El área es: {area} cm^2")
```

```
informacion_triangulo2(2,3,4)
```

```
El perímetro es: 9 cm
```

```
El semiperímetro es: 4.5 cm
```

```
El área es: 2.9047375096555625 cm^2
```

Capítulo 4: Estructura decisión lógica

Ejercicios resueltos:

7. Hacer un algoritmo que, dados dos valores numéricos A y B, escriba un mensaje diciendo si A es mayor, menor o igual a B.

```
def mayor_menor(a,b):
    if a > b:
        print(f"{a} es mayor que {b}")
    elif a < b:
        print(f"{a} es menor que {b}")
    else:
        print(f"{a} es igual a {b}")
```

```
mayor_menor(7,5)
```

```
7 es mayor que 5
```

10. Cierta universidad para liquidar el pago de matrícula de un estudiante le exige los siguientes datos:

>• Número de inscripción

>• Nombres

>• Patrimonio.

>• Estrato social.

La universidad cobra un valor constante para cada estudiante de \$50.000. Si el patrimonio es mayor que \$2'000.000 y el estrato superior a 3, se le incrementa un porcentaje del 3% sobre el patrimonio. Hacer un algoritmo que muestre:

>• Número de inscripción.

>• Nombres.

>• Pago de matrícula.

```
def precio_matricula(num, nom, pat, estrato):  
    precio_matricula = 50000  
    if pat > 2000000 and estrato > 3:  
        precio_matricula = precio_matricula + 0.03*pat  
    print(f"El número de matrícula es: {num}")  
    print(f"El nombre del estudiante es: {nom}")  
    print(f"El precio de la matrícula que debe pagar el estudiante es: $ {precio_matricula}")
```

```
precio_matricula(1,"David",3000000,4)
```

```
El número de matrícula es: 1
```

```
El nombre del estudiante es: David
```

```
El precio de la matrícula que debe pagar el estudiante es: $ 140000.0
```

11. Escribir un algoritmo que acepte tres números enteros diferentes y muestre el mayor de ellos.

```
def mas_grande_3(a,b,c):  
    if a > b and a > c:  
        print(f"{a} es el número más grande entre {a,b,c}")  
    elif b > c:  
        print(f"{b} es el número más grande entre {a,b,c}")  
    else:  
        print(f"{c} es el número más grande entre {a,b,c}")
```

```
mas_grande_3(3,5,7)
```

7 es el número más grande entre (3, 5, 7)

12. Determinar la cantidad de dinero recibida por un trabajador por concepto de las horas semanales trabajadas en una empresa, sabiendo que cuando las horas de trabajo exceden de 40, el resto se considera horas extras y se pagan al doble de una hora normal, cuando no exceden de 8; si las horas extras exceden de 8, se pagan las primeras 8 al doble de lo que se paga una hora normal y el resto al triple. Del trabajador se conocen los siguientes datos: nombres, número de horas trabajadas en la semana y valor recibido por una hora normal de trabajo.

```
def dinero_recibido(horas_trabajadas, nombre, precio_hora):
    if horas_trabajadas > 40:
        horas_extras = horas_trabajadas - 40
        if horas_extras > 8:
            salario = (40 * precio_hora) + (8 * precio_hora * 2) + ((horas_extras - 8) * precio_hora * 3)
        else:
            salario = (40 * precio_hora) + (horas_extras * precio_hora * 2)
    else:
        salario = horas_trabajadas * precio_hora
    print(f"El trabajador {nombre} devengó un salario de: $ {salario}")

dinero_recibido(50, "Juan", 4000)
```

El trabajador Juan devengó un salario de: \$ 248000

13. Un almacén efectúa una promoción en la cual se hace un descuento sobre el valor de la compra total, según el color de la bolita que el cliente saque al pagar en caja. Si la bolita es blanca no se le hará descuento alguno, si es verde se le hará un 10% de descuento, si es amarilla un 25%, si es azul un 50% y si es roja un 100%. Hacer un algoritmo para determinar la cantidad final que un cliente deberá pagar por su compra. Se sabe que sólo hay bolitas de los colores mencionados.

```
def cantidad_pagar(precio):
    num_rand = np.random.randint(0,5)
    colores = ["rojo", "verde", "azul", "amarillo", "blanco"]
    descuentos = {"blanco": 0, "verde": 0.1, "amarillo": 0.25, "azul": 0.5, "rojo": 1}
    total_pagar = precio - (precio * descuentos[colores[num_rand]])
    print(f"Sacaste la bolita {colores[num_rand]} el precio a pagar es: $ {total_pagar} pesos")

cantidad_pagar(10000)
```

Sacaste la bolita amarillo el precio a pagar es: \$ 7500.0 pesos

14. Una empresa con tres departamentos tiene establecido un plan de incentivos para sus vendedores. Al final del período, a cada departamento se le pide el importe global de las ventas. A los departamentos que excedan el

33% de las ventas totales se les adiciona al salario de los vendedores un porcentaje equivalente al 20% del salario mensual. Las nóminas de los tres departamentos son iguales. Si se tienen los siguientes datos:

- > • Ventas del departamento 1
- > • Ventas del departamento 2
- > • Ventas del departamento 3
- > • Salario de los vendedores de cada departamento

Hacer un algoritmo que determine cuánto recibirán los vendedores de cada departamento al finalizar el período.

```
def salario_departamentos(v1,v2,v3,sal):  
    total = v1 + v2 + v3  
    porcentaje1 = v1 / total  
    porcentaje2 = v2 / total  
    porcentaje3 = v3 / total  
  
    if porcentaje1 > 0.33:  
        sal1 = sal + (sal * 0.2)  
    else:  
        sal1 = sal  
  
    if porcentaje2 > 0.33:  
        sal2 = sal + (sal * 0.2)  
    else:  
        sal2 = sal  
  
    if porcentaje3 > 0.33:  
        sal3 = sal + (sal * 0.2)  
    else:  
        sal3 = sal  
  
    print(f"El salario del departamento 1 es: $ {sal1}")  
    print(f"El salario del departamento 2 es: $ {sal2}")  
    print(f"El salario del departamento 3 es: $ {sal3}")  
  
salario_departamentos(3,4,5,1000000)
```

```
El salario del departamento 1 es: $ 1000000  
El salario del departamento 2 es: $ 1200000.0  
El salario del departamento 3 es: $ 1200000.0
```

15. Se tienen cuatro esferas (A, B, C, D) de las cuales se sabe que tres son de igual peso y una diferente. Elaborar un algoritmo que determine cuál es la esfera diferente y si es de mayor o menor peso.

```
def det_pes_dif(peso1, peso2, peso3, peso4):  
    arr = [peso1, peso2, peso3, peso4]  
    dict = {0:'A', 1:'B', 2:'C', 3:'D'}  
    for i in range(3):
```

```

if arr[i] != arr[i+1]:
    es_mayor = "Tiene mayor peso" if arr[i] < arr[i+1] else "Tiene menor peso"
    print(f"El peso diferente es la esfera: {dict[i+1]} y {es_mayor}")
    break

```

```
det_pes_dif(3,1,3,3)
```

```
El peso diferente es la esfera: B y Tiene menor peso
```

Ejercicios propuestos:

22. Elaborar un algoritmo que entre el nombre de un empleado, su salario básico por hora y el número de horas trabajadas en el mes; escriba su nombre y salario mensual si éste es mayor de \$450.000, de lo contrario escriba sólo el nombre.

```

nombre = input("Ingrese su nombre: ")
precio_hora = float(input("Ingrese la ganancia por hora: "))
horas = float(input("Ingrese las horas trabajadas: "))

```

```
salario_mes = precio_hora * horas
```

```

if salario_mes > 450000:
    print(f"El salario de {nombre} es: {salario_mes}")
else:
    print(f"El trabajador se llama {nombre}")

```

```
El salario de Daniel es: 2500000.0
```

23. Dados los valores A, B y C que son los parámetros de una ecuación de segundo grado, elaborar un algoritmo para hallar las posibles soluciones de dicha ecuación.

```

def solucion_ecn_segundo_grado(a,b,c):
    discriminante = b**2 - 4*a*c
    if discriminante > 0:
        x1 = (-b + sqrt(discriminante)) / (2*a)
        x2 = (-b - sqrt(discriminante)) / (2*a)
        print(f"Las soluciones son: {x1} y {x2}")

```

```

elif discriminante == 0:
    x = -b / (2*a)
    print(f"La solución es: {x}")

```

```

else:
    print("No tiene solución real")

```

```
solucion_ecn_segundo_grado(1,3,2)
```

Las soluciones son: -1.0 y -2.0

24. Se tienen tres esferas (A,B,C) de diferente peso, elaborar un algoritmo que determine cuál es la esfera de mayor peso.

```
def peso_esferas(a,b,c):  
    if a > b and a > c:  
        print(f"La esfera A es la más pesada")  
    elif b > c:  
        print(f"La esfera B es la más pesada")  
    else:  
        print(f"La esfera C es la más pesada")  
  
peso_esferas(3,5,7)  
  
La esfera C es la más pesada
```

Parte 2:

2.4 Páginas 87-95 llamado **Enunciado: clases sobre figuras geométricas**

• **Codificación:** Python ejercicio de las páginas 87-95 llamado “**clases sobre figuras geométricas**” del libro Ejercicios de programación orientada a objetos con Java y UML de Leonardo Bermón Angarita

Se requiere un programa que modele varias figuras geométricas: el círculo, el rectángulo, el cuadrado y el triángulo rectángulo.

- > El círculo tiene como atributo su radio en centímetros.
- > El rectángulo, su base y altura en centímetros.
- > El cuadrado, la longitud de sus lados en centímetros.
- > El triángulo, su base y altura en centímetros.

Se requieren métodos para determinar el área y el perímetro de cada figura geométrica. Además, para el triángulo rectángulo se requiere:

- > Un método que calcule la hipotenusa del rectángulo.
- > Un método para determinar qué tipo de triángulo es
 - Equilátero: todos sus lados son iguales.
 - Isósceles: tiene dos lados iguales.
 - Escaleno: todos sus lados son diferentes.

Se debe desarrollar una clase de prueba con un método main para crear las cuatro figuras y probar los métodos respectivos.

Enunciado: clases sobre figuras geométricas

- **Diagrama de clases:** realizar el diagrama de clases en Draw.io

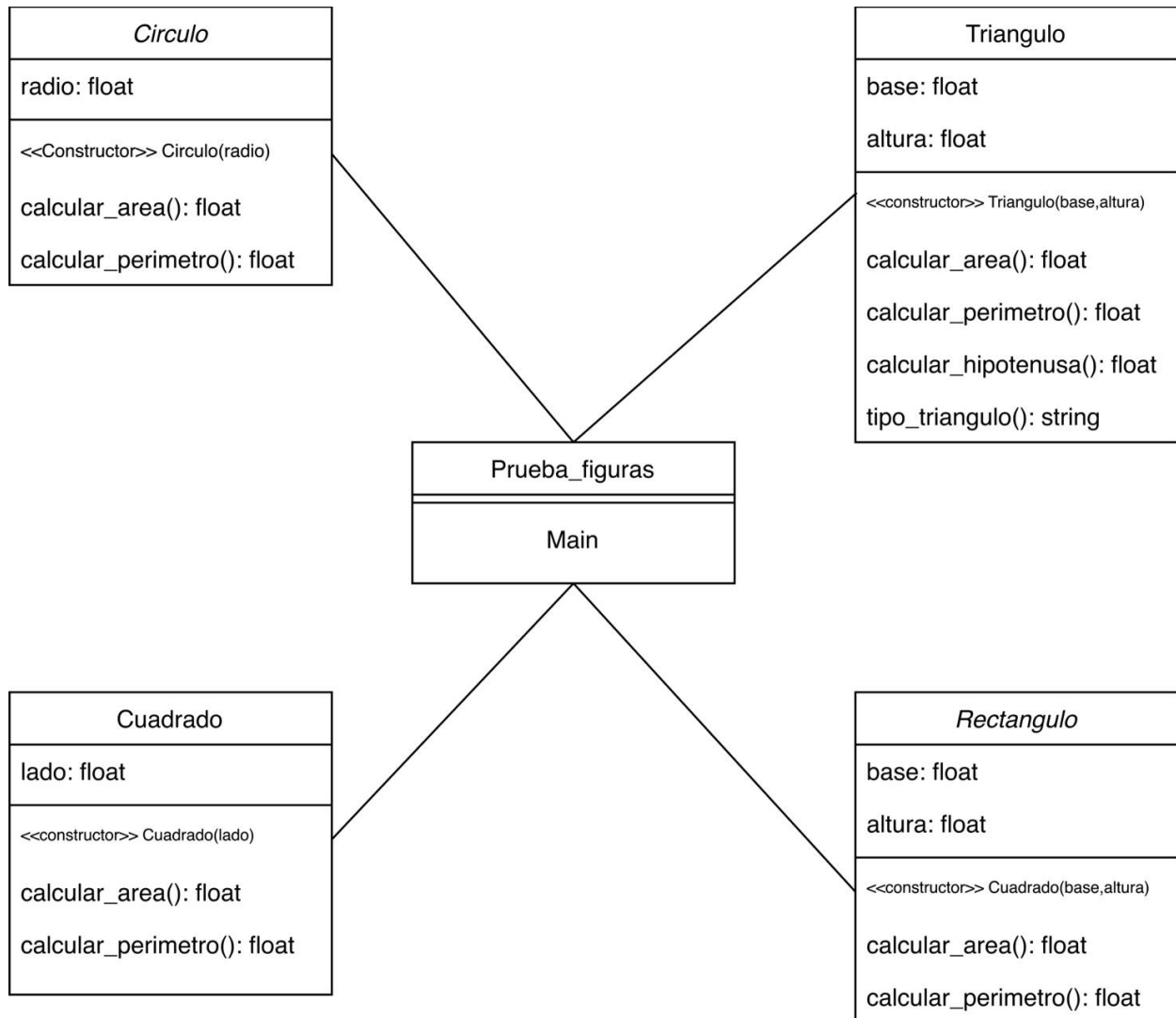


Figura 1. Diagrama de clases.

```
from math import pi, pow, sqrt

class Circulo:

    def __init__(self, radio: float) -> None:
        self.radio = radio

    def calcular_area(self) -> float:
```

```

    return pi*pow(self.radio,2)

def calcular_perimetro(self) -> float:
    return 2*pi*self.radio

class Cuadrado:

    def __init__(self, lado: float) -> None:
        self.lado = lado

    def calcular_area(self) -> float:
        return self.lado * self.lado

    def calcular_perimetro(self) -> float:
        return 4*self.lado

class Rectangulo:

    def __init__(self, base: float, altura: float) -> None:
        self.base = base
        self.altura = altura

    def calcular_area(self) -> float:
        return self.base * self.altura

    def calcular_perimetro(self) -> float:
        return 2*self.base + 2*self.altura

class Triangulo:

    def __init__(self, base: float, altura: float) -> None:
        self.base = base
        self.altura = altura

    def calcular_area(self) -> float:
        return (self.base*self.altura)/2

    def calcular_hipotenusa(self) -> float:
        return sqrt(pow(self.base,2) + pow(self.altura,2))

    def calcular_perimetro(self) -> float:
        return self.base + self.altura + self.calcular_hipotenusa()

    def tipo_triangulo(self) -> str:
        if (self.altura == self.base) and (self.base == self.calcular_hipotenusa()) and (self.altura ==
self.calcular_hipotenusa()):
            return 'El triangulo es equilatero'

```

```

    elif (self.altura != self.base) and (self.base != self.calcular_hipotenusa()) and (self.altura !=
self.calcular_hipotenusa()):
        return 'El triangulo es escaleno'

    else:
        return 'El triangulo es isocetes'

a1 = float(input("Ingrese el radio del circulo en cm: "))
b1 = float(input("Ingrese el lado del cuadrado en cm: "))
c1 = float(input("Ingrese la base del rectangulo en cm: "))
d1 = float(input("Ingrese la altura del rectangulo en cm: "))
e1 = float(input("Ingrese la base del triangulo en cm: "))
f1 = float(input("Ingrese la altura del triangulo en cm: "))

def main(a2,b2,c2,d2,e2,f2):
    c = Circulo(a2)
    print(f'El area del circulo es: {c.calcular_area()} cm^2')
    print(f'El perimetro del circulo es: {c.calcular_perimetro()} cm')
    print('\n')

    r = Rectangulo(b2,c2)
    print(f'El area del rectangulo es: {r.calcular_area()} cm^2')
    print(f'El perimetro del rectangulo es: {r.calcular_perimetro()} cm')
    print('\n')

    cu = Cuadrado(d2)
    print(f'El area del cuadrado es: {cu.calcular_area()} cm^2')
    print(f'El perimetro del cuadrado es: {cu.calcular_perimetro()} cm')
    print('\n')

    t = Triangulo(e2,f2)
    print(f'El area del triangulo es: {t.calcular_area()} cm')
    print(f'El perimetro del triangulo es: {t.calcular_perimetro()} cm')
    print(f'la hipotenusa del triagunlo es de {t.calcular_hipotenusa()} cm')
    print(f'{t.tipo_triangulo()}')

if __name__ == "__main__":
    main(a1,b1,c1,d1,e1,f1)

```

```

El area del circulo es: 28.274333882308138 cm^2
El perimetro del circulo es: 18.84955592153876 cm

```

```

El area del rectangulo es: 8.0 cm^2
El perimetro del rectangulo es: 12.0 cm

```

```

El area del cuadrado es: 9.0 cm^2
El perimetro del cuadrado es: 12.0 cm

```

El area del triangulo es: 6.0 cm
El perimetro del triangulo es: 12.0 cm
la hipotenusa del triagunlo es de 5.0 cm
El triangulo es escaleno

URL de los ejercicios: https://github.com/dgilrua/POO/blob/main/Actividad_2.ipynb