

# Ejercicios Programación en Python

---

David Camilo Cortes Salazar

Análisis y Visualización de Datos

Talento Tech

Bogotá D.C.

2024

# Ejercicios Programación en Python

---

David Camilo Cortes Salazar

Docente: Carlos Palacios

Análisis y Visualización de Datos

Talento Tech

Bogotá D.C.

2024

# Problema 1

Elaborar un algoritmo que halle el perímetro y el área de cinco figuras geométricas (diferentes a las hechas en clase).

Algoritmo:

```
import math

# INICIO

lado = float(input("Ingrese el tamaño del lado de la figura: "))

# PROCESOS

# Pentágono
perimetro_pentagono = 5 * lado
apotema_pentagono = lado / ( 2 * math.tan(math.pi / 5))
area_pentagono = perimetro_pentagono * apotema_pentagono / 2

# Hexágono
perimetro_hexagono = 6 * lado
apotema_hexagono = lado / ( 2 * math.tan(math.pi / 6))
area_hexagono = perimetro_hexagono * apotema_hexagono / 2

# Heptágono
perimetro_heptagono = 7 * lado
apotema_heptagono = lado / ( 2 * math.tan(math.pi / 7))
area_heptagono = perimetro_heptagono * apotema_heptagono / 2

# Octágono
perimetro_octagono = 8 * lado
apotema_octagono = lado / ( 2 * math.tan(math.pi / 8))
area_octagono = perimetro_octagono * apotema_octagono / 2

# Eneágono
perimetro_eneagono = 9 * lado
apotema_eneagono = lado / ( 2 * math.tan(math.pi / 9))
area_eneagono = perimetro_eneagono * apotema_eneagono / 2

#SALIDA

print(f"Area del Pentágono: {round(area_pentagono,2)}")
print(f"Area del Hexágono: {round(area_hexagono,2)}")
print(f"Area del Heptágono: {round(area_heptagono,2)}")
print(f"Area del Octágono: {round(area_octagono,2)}")
print(f"Area del Eneágono: {round(area_eneagono,2)}")
```

## Prueba de escritorio:

```
$ python problema_1.py  
Ingrese el tamaño del lado de la figura: 4  
Area del Pentágono: 27.53  
Area del Hexágono: 41.57  
Area del Heptágono: 58.14  
Area del Octágono: 77.25  
Area del Eneágono: 98.91
```

## Problema 2

Se dese calcular la distancia recorrida (m) por un móvil que tiene velocidad constante (m/s) durante un tiempo T (sg), considerar que es (Movimiento Rectilíneo Uniforme).

### Algoritmo

```
# INICIO

velocidad = float(input("Ingrese la velocidad (m/s): "))
tiempo = float(input("Ingrese el tiempo (sg): "))

# PROCESOS

distancia = velocidad * tiempo

#SALIDA

print(f"Distancia recorrida: {round(distancia,1)} metros")
```

### Prueba de escritorio:

```
$ python problema_2.py
Ingrese la velocidad (m/s): 10
Ingrese la tiempo (sg): 5
Distancia recorrida: 50.0 metros
```

## Problema 3

---

Elaborar un algoritmo que obtenga e imprima el valor de  $y$  a partir de la ecuación  $y = 3x^2 + 7x - 15$

### Algoritmo

```
# INICIO

x = float(input("Ingrese el valor de x: "))

# PROCESOS

y = 3 * (x ** 2) + 7 * x - 15

#SALIDA

print(f"El valor de y es: {y}")
```

### Prueba de escritorio:

```
$ python problema_3.py
Ingrese el valor de x: 4
El valor de y es: 61.0
```

## Problema 4

Elabore un algoritmo para obtener el promedio simple de un estudiante a partir de sus tres notas parciales.

### Algoritmo

```
# INICIO

nota1 = float(input("Ingrese la primera nota: "))
nota2 = float(input("Ingrese la segunda nota: "))
nota3 = float(input("Ingrese la tercera nota: "))

# PROCESOS

promedio = (nota1 + nota2 + nota3) / 3

#SALIDA

print(f"El promedio es: {round(promedio, 1)}")
```

### Prueba de escritorio:

```
$ python problema_4.py
Ingrese la primera nota: 4.5
Ingrese la segunda nota: 2.3
Ingrese la tercera nota: 3.8
El promedio es: 3.5
```

## Problema 5

Elaborar un algoritmo que permita ingresar el número de partidos ganados, perdidos y empatados, por algún equipo en el torneo apertura, se debe demostrar su puntaje total, teniendo en cuenta que por cada partido ganado obtendrá 3 puntos, empatado 1 punto y perdido 0 puntos.

### Algoritmo

```
# INICIO

ganados = int(input("Ingrese los partidos ganados: "))
empatados = int(input("Ingrese los partidos empatados: "))
perdidos = int(input("Ingrese los partidos perdidos: "))

# PROCESOS

puntaje = ganados * 3 + empatados * 1 + perdidos * 0

#SALIDA

print(f"El puntaje total es: {puntaje}")
```

### Prueba de escritorio:

```
$ python problema_5.py
Ingrese los partidos ganados: 5
Ingrese los partidos empatados: 3
Ingrese los partidos perdidos: 2
El puntaje total es: 18
```



## Problema 6

Elaborar un algoritmo que permita calcular el número de CDs necesarios para hacer una copia de seguridad, de el disco cuya capacidad se conoce. Considerar que el disco duro está lleno de información y su tamaño es expresado expresado en gigabyte. Un CD virgen tiene 700 Megabytes de capacidad y una Gigabyte es igual a 1,024 megabyte.

### Algoritmo

```
import math

# INICIO

disco_gb = float(input("Ingrese el tamaño del disco (Gb): "))

# PROCESOS

disco_mb = disco_gb * 1024
numero_cds = math.ceil(disco_mb / 700)

#SALIDA

print(f"Se necesitan {numero_cds} discos para hacer la copia de seguridad")
```

### Prueba de escritorio

```
$ python problema_6.py
Ingrese el tamaño del disco (Gb): 375
Se necesitan 549 discos para hacer la copia de seguridad
```

## Problema 7

Se requiere el algoritmo para elaborar la planilla de un empleado. Para ello se dispone de sus horas laboradas en el mes, así como de la tarifa por hora.

### Algoritmo

```
# INICIO

horas_trabajadas = int(input("Ingrese las horas trabajadas: "))
hora_tarifa = int(input("Ingrese la tarifa por hora: "))

# PROCESOS

salario = horas_trabajadas * hora_tarifa

#SALIDA

print(f"\n>---- Planilla ----<\n" +
      f"Horas Trabajadas: \t{horas_trabajadas}\n" +
      f"Tarifa por Hora: \t{hora_tarifa}\n" +
      f"Salario: \t\t{salario}")
```

### Prueba de escritorio:

```
$ python problema_7.py
Ingrese las horas trabajadas: 72
Ingrese la tarifa por hora: 12000

>---- Planilla ----<
Horas Trabajadas:      72
Tarifa por Hora:      12000
Salario:               864000
```

## Problema 8

Elabore un algoritmo que lea los 3 lados de un triángulo cualquiera y calcule su área, considerar: Si  $a$ ,  $b$  y  $c$  son los lados, y  $S$  el semiperímetro.

$$A = \sqrt{S * (S - a) * (S - b) * (S - c)}$$

El semiperimetro es la suma de los lados dividido en 2.

$$S = (a + b + c)/2$$

### Algoritmo

```
import math

# INICIO

a = float(input("Ingrese el valor del lado a: "))
b = float(input("Ingrese el valor del lado b: "))
c = float(input("Ingrese el valor del lado c: "))

# PROCESOS

s = (a + b + c) / 2
area = math.sqrt(s * (s - a) * (s - b) * (s - c))

#SALIDA

print(f"El area del triangulo es {round(area,1)} y su semiperimetro es {round(s,1)}")
```

### Prueba de escritorio:

```
$ python problema_8.py
Ingrese el valor del lado a: 5
Ingrese el valor del lado b: 6
Ingrese el valor del lado c: 7
El area del triangulo es 14.7 y su semiperimetro es 9.0
```

## Problema 9

Se tiene los puntos A y B en el cuadrante positivo del plano cartesiano, elabore el algoritmo que permite obtener la distancia entre A y B.

### Algoritmo

```
import math

# INICIO

x1 = float(input("Ingrese la coordenada x del primer punto: "))
y1 = float(input("Ingrese la coordenada y del primer punto: "))
x2 = float(input("Ingrese la coordenada x del segundo punto: "))
y2 = float(input("Ingrese la coordenada y del segundo punto: "))

# PROCESOS

distancia= math.sqrt((x2 - x1)**2 + (y2 - y1)**2)

#SALIDA

print(f"La distancia entre los puntos ({x1},{y1}) y ({x2},{y2}) es {round(distancia,1)}")
```

### Prueba de escritorio:

```
$ python problema_9.py
Ingrese la coordenada x del primer punto: 2.3
Ingrese la coordenada y del primer punto: 1.1
Ingrese la coordenada x del segundo punto: 10.5
Ingrese la coordenada y del segundo punto: 5.2
La distancia entre los puntos (2.3,1.1) y (10.5,5.2) es 9.2
```

## Problema 10

Se tiene registrado la producción (unidades) logradas por un operario a lo largo de la semana (lunes a sábado). Elabore un algoritmo que nos muestre o nos diga si el operario recibirá incentivos sabiendo que el promedio de producción mínima es de 100 unidades.

### Algoritmo

### Prueba de escritorio:

```
$ python problema_9.py
```

## Problema 11

Elabore una aplicación Pseint que realice las siguientes operaciones matemáticas y las convierta en aplicaciones algorítmicas.

### Algoritmo

### Prueba de escritorio:

```
$ python problema_9.py
```