ESTUDIOS
GENERALES
CIENCIAS

PONTIFICIA
UNIVERSIDAD
CATÓLICA
DEL PERÚ

FUNDAMENTOS DE PROGRAMACIÓN
LABORATORIO 1
PROPUESTAS DE SOLUCIÓN
SEMESTRE ACADÉMICO 2021-2

Horarios: Todos los horarios

Elaborado por David Allasi

INDICACIONES:
▪ Debe utilizar variables descriptivas, comentarios y mensajes descriptivos. ▪ El orden y la eficiencia de su implementación serán considerados en la calificación.

RESULTADOS ESPERADOS:
▪ Al finalizar la sesión, el alumno comprenderá la estructura clásica de los algoritmos y programas secuenciales. ▪ Al finalizar la sesión, el alumno construirá algoritmos y programas usando operaciones de lectura y salida de datos. ▪ Al finalizar la sesión, el alumno diseñará algoritmos secuenciales representándolos a través de pseudocódigos. ▪ Al finalizar la sesión, el alumno construirá programas secuenciales en lenguaje C. ▪ Al finalizar la sesión, el alumno construirá programas usando las funciones matemáticas de la librería estándar de lenguaje C.

CONSIDERACIONES:
▪ La solución presentada para cada problema corresponde a una propuesta de solución por parte del autor. ▪ En programación pueden existir muchas soluciones para un mismo problema pero debe cumplir con todo lo solicitado, incluyendo las restricciones brindadas.

Desarrolle los siguientes problemas en PSeInt:

1. Circunferencias Tangentes Interiores y Secantes

Dos circunferencias son tangentes interiores cuando tienen un punto común (punto de tangencia). Además, la distancia entre los centros de las dos circunferencias es igual a la diferencia de sus radios, ver la figura 1:

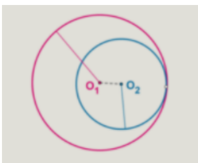

Figura 1: Circunferencias tangentes interiores

Dos circunferencias son secantes cuando tienen dos puntos en común. Además, la distancia entre los centros de las dos circunferencias es menor que la suma de sus radios y mayor que la diferencia de sus radios, ver la figura 2:

Se le pide que diseñe un algoritmo, expresado en pseudocódigo, que lea 2 puntos del plano cartesiano O_1 y O_2 que representan el punto centro de dos circunferencias. Además, debe leer el radio de las mismas. Con estos datos, debe calcular el área y longitud de cada circunferencia y además debe evaluar si ambas circunferencias son tangentes

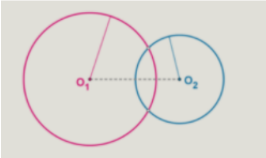

Figura 2: Circunferencias secantes

interiores o secantes, sin considerar los puntos en común. Muestre la salida en el formato indicado y utilice para los cálculos el valor de 3.141592 para PI.

Recuerde

- La distancia entre dos puntos $P(x_1, y_1)$ y $Q(x_2, y_2)$ se calcula con la siguiente fórmula: $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$.
- Área de una circunferencia es $\pi * radio^2$.
- Longitud de la circunferencia es $2 * \pi * radio$.

Comparación de números reales

Muchas veces el resultado de la comparación de números reales a través de la igualdad no es el deseado. En este caso es recomendable usar el valor absoluto de la diferencia de los números que se desean comparar. Si esta diferencia es cercana a cero (en esta ocasión será menor a 0.0001), se puede asumir que son iguales. Considere que en PSeInt se utiliza la función abs() para obtener el valor absoluto de un número.

A continuación se muestra unos ejemplos de ejecución como casos de prueba:

Caso de Prueba 1:

```
Ingrese las coordenadas x e y del centro de la circunferencia 1:
-1
4
Ingrese las coordenadas x e y del centro de la circunferencia 2:
3
4
Ingrese el radio de la circunferencia 1:
5
Ingrese el radio de la circunferencia 2:
1

Resultado de la circunferencia 1:
El área es: 78.5398
La longitud es: 31.41592
Resultado de la Circunferencia 2:
El área es: 3.141592
La longitud es: 6.283184
¿Son circunferencias tangentes interiores? VERDADERO
¿Son circunferencias secantes? FALSO
```

Caso de Prueba 2:

```
Ingrese las coordenadas x e y del centro de la circunferencia 1:
3
-1
Ingrese las coordenadas x e y del centro de la circunferencia 2:
-1
2
Ingrese el radio de la circunferencia 1:
```

```
4
Ingrese el radio de la circunferencia 2:
3
Resultado de la Circunferencia 1:
El área es: 50.265472
La longitud es: 25.132736
Resultado de la Circunferencia 2:
El área es: 28.274328
La longitud es: 18.849552
¿Son circunferencias tangentes interiores? FALSO
¿Son circunferencias secantes? VERDADERO
```

Caso de Prueba 3:

```
Ingrese las coordenadas x e y del centro de la circunferencia 1:
1.5
2.8
Ingrese las coordenadas x e y del centro de la circunferencia 2:
3.8
5.4
Ingrese el radio de la circunferencia 1:
6.5
Ingrese el radio de la circunferencia 2:
2.4

Resultado de la circunferencia 1:
El área es: 132.732262
La longitud es: 40.840696
Resultado de la Circunferencia 2:
El área es: 18.09556992
La longitud es: 15.0796416
¿Son circunferencias tangentes interiores? FALSO
¿Son circunferencias secantes? FALSO
```

Programa 1: Propuesta de solución - Circunferencias Tangentes Interiores y Secantes

```
Algoritmo CircunferenciasTangentes
    Escribir "Ingrese la coordenada x e y del centro de la circunferencia 1: "
    Leer x1,y1
    Escribir "Ingrese la coordenada x e y del centro de la circunferencia 2: "
    Leer x2,y2
    Escribir "Ingrese el radio de la circunferencia 1: "
    Leer radio1
    Escribir "Ingrese el radio de la circunferencia 2: "
    Leer radio2
    area1 <- 3.141592 * radio1^2
    longitud1 <- 2*3.141592* radio1
    area2 <- 3.141592 * radio2^2
    longitud2 <- 2*3.141592* radio2
    distancia <- rc((x2-x1)^2 + (y2-y1)^2)
    diferenciaRadios <- abs(radio1 - radio2)
    sumaRadios <- radio1 + radio2
    es_tangente <- abs(distancia-diferenciaRadios)<0.0001
    es_secante <- (distancia < sumaRadios y abs(distancia-sumaRadios)>=0.0001) y (distancia > diferenciaRadios y abs(distancia
        -diferenciaRadios)>=0.0001)
    Escribir "Resultado de la Circunferencia 1: "
    Escribir "El área es: ", area1
    Escribir "La longitud es: ", longitud1
    Escribir "Resultado de la Circunferencia 2: "
    Escribir "El área es: ", area2
    Escribir "La longitud es: ", longitud2
    Escribir "¿Son circunferencias tangentes interiores? ", es_tangente
    Escribir "¿Son circunferencias secantes? ", es_secante
FinAlgoritmo
```

2. Triángulo en el plano cartesiano

Un triángulo, en geometría analítica, es un polígono de tres lados. Los puntos comunes a cada par de lados se denominan vértices del triángulo. A continuación se describen los elementos del triángulo, los cuáles también pueden verse en la figura 3:

- `Vértices.-` Son los puntos comunes a cada par de lados del triángulo.
- `Lados.-` Son los segmentos determinados por cada par de vértices.
- `Ángulos.-` Cada par de lados con origen común al vértice de un triángulo y que contienen dos de esos lados concurrentes se llama ángulo del triángulo.

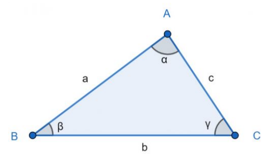

Figura 3: Elementros del triángulo

Se le pide que diseñe un algoritmo, expresado en pseudocódigo, que lea 3 puntos del plano cartesiano A, B y C que representan los vértices de un triángulo. Con estos puntos, debe calcular la distancia de cada uno de los lados (AB, BC y AC) del triángulo, además, debe mostrar la pendiente de la recta que pasa por dichos lados y finalmente debe indicar si el triángulo es escaleno, isósceles o equilátero. Muestre la salida en el formato indicado.

Recuerde

- La distancia entre dos puntos $P(x_1, y_1)$ y $Q(x_2, y_2)$ se calcula con la siguiente fórmula: $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$.
- La pendiente de una recta que pasa por dos puntos $P(x_1, y_1)$ y $Q(x_2, y_2)$ se calcula con la siguiente fórmula: $\frac{y_2 - y_1}{x_2 - x_1}$.
- Un triángulo es escaleno cuando sus 3 lados tienen distancias diferentes.
- Un triángulo es isósceles cuando 2 de sus lados tienen distancias iguales y el otro lado diferente.
- Un triángulo es equilatero cuando sus 3 lados tienen distancias iguales.

Comparación de números reales

Muchas veces el resultado de la comparación de números reales a través de la igualdad no es el deseado. En este caso es recomendable usar el valor absoluto de la diferencia de los números que se desean comparar. Si esta diferencia es cercana a cero (en esta ocasión será menor a 0.0001), se puede asumir que son iguales. Considere que en PSeInt se utiliza la función abs() para obtener el valor absoluto de un número.

A continuación se muestra unos ejemplos de ejecución como casos de prueba:

Caso de Prueba 1:

```
Ingrese las coordenadas x e y del punto A:
-3
-7
Ingrese las coordenadas x e y del punto B:
10
0
Ingrese las coordenadas x e y del punto C:
```

```
-10
6

La distancia del lado AB es: 14.7648230602
La distancia del lado BC es: 20.8806130178
La distancia del lado AC es: 14.7648230602
La pendiente de la recta que pasa por el segmento AB es: 0.5384615385
La pendiente de la recta que pasa por el segmento BC es: -0.3
La pendiente de la recta que pasa por el segmento AC es: -1.8571428571
¿El triángulo es escaleno? FALSO
¿El triángulo es isósceles? VERDADERO
¿El triángulo es equilátero? FALSO
```

Caso de Prueba 2:

```
Ingrese las coordenadas x e y del punto A:
-3
-2
Ingrese las coordenadas x e y del punto B:
-2
1
Ingrese las coordenadas x e y del punto C:
4
2

La distancia del lado AB es: 3.1622776602
La distancia del lado BC es: 6.0827625303
La distancia del lado AC es: 8.0622577483
La pendiente de la recta que pasa por el segmento AB es: 3
La pendiente de la recta que pasa por el segmento BC es: 0.1666666667
La pendiente de la recta que pasa por el segmento AC es: 0.5714285714
¿El triángulo es escaleno? VERDADERO
¿El triángulo es isósceles? FALSO
¿El triángulo es equilátero? FALSO
```

Programa 2: Propuesta de solución - Triángulo en el plano cartesiano

```
Algoritmo Evaluar_Triangulo
    Escribir "Ingrese las coordenadas x e y del punto A: "
    Leer xa, ya
    Escribir "Ingrese las coordenadas x e y del punto B: "
    Leer xb, yb
    Escribir "Ingrese las coordenadas x e y del punto C: "
    Leer xc, yc

    distanciaAB <- rc((xb-xa)^2 + (yb-ya)^2)
    distanciaBC <- rc((xc-xb)^2 + (yc-yb)^2)
    distanciaAC <- rc((xc-xa)^2 + (yc-ya)^2)

    mAB <- (yb-ya)/(xb-xa)
    mBC <- (yc-yb)/(xc-xb)
    mAC <- (yc-ya)/(xc-xa)

    es_equilatero <- abs(distanciaAB - distanciaBC)<0.0001 y abs(distanciaBC - distanciaAC)<0.0001
    es_isosceles <- (abs(distanciaAB - distanciaBC)<0.0001 y abs(distanciaAB - distanciaAC)>=0.0001) o (abs(distanciaBC - distanciaAC)<0.0001 y abs(distanciaBC - distanciaAB)>=0.0001) o (abs(distanciaAB - distanciaAC)<0.0001 y abs( distanciaAB - distanciaBC)>=0.0001)
    es_escaleno <- abs(distanciaAB - distanciaBC)>=0.0001 y abs(distanciaBC - distanciaAC)>=0.0001 y abs(distanciaAB - distanciaAC)>=0.0001

    Escribir "La distancia del lado AB es: ", distanciaAB
    Escribir "La distancia del lado BC es: ", distanciaBC
    Escribir "La distancia del lado AC es: ", distanciaAC

```

```
        Escribir ”La pendiente de la recta que pasa por el segmento AB es: ”, mAB
        Escribir ”La pendiente de la recta que pasa por el segmento BC es: ”, mBC
        Escribir ”La pendiente de la recta que pasa por el segmento AC es: ”, mAC

        Escribir ”¿El triángulo es escaleno? ”, es_escaleno
        Escribir ”¿El triángulo es isósceles? ”, es_isosceles
        Escribir ”¿El triángulo es equilátero? ”, es_equilatero
FinAlgoritmo
```

3. Circunferencias Tangentes Exteriores

Dos circunferencias son tangentes exteriores cuando tienen un punto común (punto de tangencia). La distancia entre los centros de las dos circunferencias es igual a la suma de sus radios, ver la figura 4:

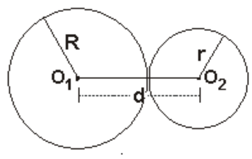

Figura 4: Circunferencias tangentes exteriores

Se le pide que diseñe un algoritmo, expresado en pseudocódigo, que lea 2 puntos del plano cartesiano O_1 y O_2 que representan el punto centro de dos circunferencias. Además, debe leer el radio de las mismas. Con estos datos, debe calcular el área y longitud de cada circunferencia y además debe evaluar si ambas circunferencias son tangentes exteriores. Muestre la salida en el formato indicado y utilice para los cálculos el valor de 3.141592 para PI.

Recuerde

- La distancia entre dos puntos $P(x_1, y_1)$ y $Q(x_2, y_2)$ se calcula con la siguiente fórmula: $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$.
- Área de una circunferencia es $\pi * radio^2$.
- Longitud de la circunferencia es $2 * \pi * radio$.

Comparación de números reales

Muchas veces el resultado de la comparación de números reales a través de la igualdad no es el deseado. En este caso es recomendable usar el valor absoluto de la diferencia de los números que se desean comparar. Si esta diferencia es cercana a cero (en esta ocasión será menor a 0.0001), se puede asumir que son iguales. Considere que en PSeInt se utiliza la función abs() para obtener el valor absoluto de un número.

A continuación se muestra unos ejemplos de ejecución como casos de prueba:

Caso de Prueba 1:

```
Ingrese las coordenadas x e y del centro de la circunferencia 1:
-1
1
Ingrese las coordenadas x e y del centro de la circunferencia 2:
4
1
Ingrese el radio de la circunferencia 1:
3
Ingrese el radio de la circunferencia 2:
2
```

```
Resultado de la circunferencia 1:
El área es: 28.274328
La longitud es: 18.849552
Resultado de la Circunferencia 2:
El área es: 12.566368
La longitud es: 12.566368
¿Son circunferencias tangentes? VERDADERO
```

Caso de Prueba 2:

```
Ingrese las coordenadas x e y del centro de la circunferencia 1:
1.5
2.9
Ingrese las coordenadas x e y del centro de la circunferencia 2:
1.2
3.5
Ingrese el radio de la circunferencia 1:
5.8
Ingrese el radio de la circunferencia 2:
4.7
Resultado de la Circunferencia 1:
El área es: 105.68315488
La longitud es: 36.4424672
Resultado de la Circunferencia 2:
El área es: 69.39776728
La longitud es: 29.5309648
¿Son circunferencias tangentes? FALSO
```

Programa 3: Propuesta de solución - Circunferencias Tangentes Exteriores

```
Algoritmo CircunferenciasTangentes
	Escribir "Ingrese la coordenada x e y del centro de la circunferencia 1: "
	Leer x1,y1
	Escribir "Ingrese la coordenada x e y del centro de la circunferencia 2: "
	Leer x2,y2
	Escribir "Ingrese el radio de la circunferencia 1: "
	Leer radio1
	Escribir "Ingrese el radio de la circunferencia 2: "
	Leer radio2
	area1 <- 3.141592 * radio1^2
	longitud1 <- 2*3.141592* radio1
	area2 <- 3.141592 * radio2^2
	longitud2 <- 2*3.141592* radio2
	distancia <- rc((x2-x1)^2 + (y2-y1)^2)
	sumaRadios <- radio1 + radio2
	es_tangente <- abs(distancia-sumaRadios)<0.0001
	Escribir "Resultado de la Circunferencia 1: "
	Escribir "El área es: ", area1
	Escribir "La longitud es: ", longitud1
	Escribir "Resultado de la Circunferencia 2: "
	Escribir "El área es: ", area2
	Escribir "La longitud es: ", longitud2
	Escribir "¿Son circunferencias tangentes? ", es_tangente
FinAlgoritmo
```

4. Movimiento Circular Uniforme

En física, el movimiento circular uniforme (también denominado movimiento uniformemente circular o MCU) describe el movimiento de un móvil con una velocidad constante y una trayectoria circular.

La velocidad del móvil no cambia pero cambia constantemente de dirección. El móvil tiene una aceleración que está dirigida hacia el centro de la trayectoria, denominada aceleración normal y su fórmula es: $a_n = v^2/r$, donde v = velocidad y r = radio de la circunferencia donde realiza el movimiento. (ver figura 5)

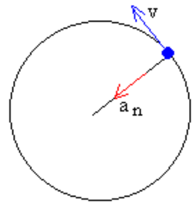

Figura 5: Aceleración normal - Movimiento circular uniforme

La segunda ley de Newton afirma, que la resultante de las fuerzas F que actúan sobre un móvil que describe un movimiento circular uniforme es igual al producto de la masa m por la aceleración normal a_n, de la siguiente manera $F = m * a_n$ siendo su unidad de medida el newton.

Se le pide que diseñe un algoritmo, expresado en pseudocódigo, que lea la distancia de la pista circular (en metros) que recorrerán dos autos de juguetes con una velocidad constante (MCU) para completar una vuelta y el tiempo (en segundos) que se demoraron cada uno de ellos. Además, debe solicitar la masa (en kilogramos) de cada auto de juguete que se utilizaron. Con estos datos, debe calcular cuál es la magnitud de la fuerza central (F), en newton, que lo mantiene en un círculo a cada auto de juguete e indicar cuál de los dos tuvo la mayor fuerza central. En caso tengan la misma fuerza central debe indicar que el auto 1 tuvo mayor fuerza central. Muestre la salida en el formato indicado.

Recuerde

- La $distancia = velocidad * tiempo$.
- La longitud de una circunferencia = $2 * \pi * r$, donde r = radio de la circunferencia.
- Utilice un valor de PI=3.141592 en caso de ser necesario

A continuación se muestra unos ejemplos de ejecución como casos de prueba:

Caso de Prueba 1:

```
Ingrese la distancia (en metros) de la pista circular:
200
Ingrese el tiempo (en segundos) que se demoró en recorrer la distancia de la
pista circular el auto 1:
25
Ingrese el tiempo (en segundos) que se demoró en recorrer la distancia de la
pista circular el auto 2:
30
Ingrese la masa (en kg) del auto 1:
1.5
Ingrese la masa (en kg) del auto 2:
2

La Fuerza central para el auto 1 es: 3.01592832 newton
La Fuerza central para el auto 2 es: 2.7925262222 newton
¿Fuerza central del auto 1 es mayor que el auto 2? VERDADERO
¿Fuerza central del auto 2 es mayor que el auto 1? FALSO
```

Caso de Prueba 2:

```
Ingrese la distancia (en metros) de la pista circular:
150
```

```
Ingrese el tiempo (en segundos) que se demoró en recorrer la distancia de la
pista circular el auto 1:
42
Ingrese el tiempo (en segundos) que se demoró en recorrer la distancia de la
pista circular el auto 2:
35
Ingrese la masa (en kg) del auto 1:
2.8
Ingrese la masa (en kg) del auto 2:
3.4

La Fuerza central para el auto 1 es: 1.4959961905 newton
La Fuerza central para el auto 2 es: 2.6158561959 newton
¿Fuerza central del auto 1 es mayor que el auto 2? FALSO
¿Fuerza central del auto 2 es mayor que el auto 1? VERDADERO
```

Programa 4: Propuesta de solución - Movimiento Circular Uniforme

```
Algoritmo MCU
    Escribir "Ingrese la distancia (en metros) de la pista circular: "
    Leer distancia
    Escribir "Ingrese el tiempo (en segundos) que se demoró en recorrer la distancia de la pista circular el auto 1: "
    Leer tiempo1
    Escribir "Ingrese el tiempo (en segundos) que se demoró en recorrer la distancia de la pista circular el auto 2: "
    Leer tiempo2
    Escribir "Ingrese la masa (en kg) del auto 1: "
    Leer masa1
    Escribir "Ingrese la masa (en kg) del auto 2: "
    Leer masa2

    velocidad1 <- distancia/tiempo1
    velocidad2 <- distancia/tiempo2
    radio <- distancia/(2*3.141592)
    F1 <- masa1 * ((velocidad1^2)/radio)
    F2 <- masa2 * ((velocidad2^2)/radio)

    Escribir "La Fuerza central para el auto 1 es: ", F1, " newton"
    Escribir "La Fuerza central para el auto 2 es: ", F2, " newton"
    Escribir "¿Fuerza central del auto 1 es mayor que el auto 2? ", F1>=F2
    Escribir "¿Fuerza central del auto 2 es mayor que el auto 1? ", F2>F1
FinAlgoritmo
```

Desarrolle los siguientes problemas en lenguaje C:

5. El ortoedro

Los ortoedros son paralelepípedos que tienen todas sus caras rectangulares (ver figura 6).

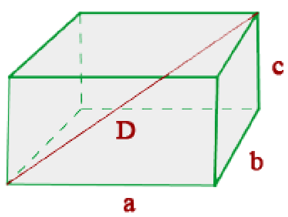

Figura 6: El ortoedro

Las fórmulas del ortoedro, tomando como referencia la figura 6, son las siguientes:

- Diagonal (D) es igual a $\sqrt{a^2 + b^2 + c^2}$.

- Área = $2 * (a * b + a * c + b * c)$.
- Volumen = $a * b * c$

Se le pide implementar un programa, en lenguaje C, que permita procesar dos ortoedros. Para ello debe solicitar el ingreso de los lados a, b y c (ver figura 6) de cada uno de ellos. Con esta información debe calcular la diagonal, el área y volumen de cada uno de ellos y además indicar cuál de los dos ortoedros tiene menor diagonal (D), en caso tengan la misma diagonal debe considerar que el ortoedro 1 es el de menor diagonal. Muestre los resultados de acuerdo al formato indicado en los casos de prueba.

A continuación se muestra unos ejemplos de ejecución como casos de prueba:

Caso de Prueba 1:

```
Ingrese los lados a, b y c del ortoedro 1: 10  4  5
Ingrese los lados a, b y c del ortoedro 2: 5  4  2.5

Resultados del ortoedro 1:
Diagonal: 11.87
Área: 220.00
Volumen: 220.00

Resultados del ortoedro 2:
Altura: 6.87
Área: 85.00
Volumen: 50.00

El ortoedro 2 tiene menor diagonal.
```

Caso de Prueba 2:

```
Ingrese los lados a, b y c del ortoedro 1: 10  2.5  3.7
Ingrese los lados a, b y c del ortoedro 2: 6  24.5  32

Resultados del ortoedro 1:
Diagonal: 10.95
Área: 142.50
Volumen: 92.50

Resultados del ortoedro 2:
Altura: 40.75
Área: 2246.00
Volumen: 4704.00

El ortoedro 1 tiene menor diagonal.
```

Caso de Prueba 3:

```
Ingrese los lados a, b y c del ortoedro 1: 5  4  2.5
Ingrese los lados a, b y c del ortoedro 2: 5  4  2.5

Resultados del ortoedro 1:
Altura: 6.87
Área: 85.00
Volumen: 50.00

Resultados del ortoedro 2:
Altura: 6.87
Área: 85.00
Volumen: 50.00

El ortoedro 1 tiene menor diagonal.
```

Programa 5: Propuesta de solución - El ortoedro

```c
#include <stdio.h>
#include <math.h>

int main(){
        double a1, b1, c1, a2, b2, c2;
        double diagonal1, diagonal2, area1, area2, volumen1, volumen2;
        int menor_ortoedro1, menor_ortoedro2;

        printf("Ingrese los lados a, b y c del ortoedro 1: ");
        scanf(" %lf %lf %lf",&a1,&b1,&c1);
        printf("Ingrese los lados a, b y c del ortoedro 2: ");
        scanf(" %lf %lf %lf",&a2,&b2,&c2);

        diagonal1 = sqrt(pow(a1,2) + pow(b1,2) + pow(c1,2));
        diagonal2 = sqrt(pow(a2,2) + pow(b2,2) + pow(c2,2));
        area1 = 2*(a1*b1 + a1*c1 + b1*c1);
        area2 = 2*(a2*b2 + a2*c2 + b2*c2);
        volumen1 = a1*b1*c1;
        volumen2 = a2*b2*c2;

        menor_ortoedro1 = diagonal1 <= diagonal2;
        menor_ortoedro2 = diagonal2 < diagonal1;

        printf("Resultados del Ortoedro 1: \n");
        printf("Diagonal: %.2lf\n",diagonal1);
        printf("Área: %.2lf\n",area1);
        printf("Volumen: %.2lf\n\n",volumen1);
        printf("Resultados del Ortoedro 2: \n");
        printf("Diagonal: %.2lf\n",diagonal2);
        printf("Área: %.2lf\n",area2);
        printf("Volumen: %.2lf\n\n",volumen2);
        printf("El ortoedro %d tiene menor diagonal",1*menor_ortoedro1 + 2*menor_ortoedro2);
        return 0;
}
```

6. El cilindro

Un cilindro es un cuerpo geométrico engendrado por un rectángulo que gira alrededor de uno de sus lados (ver figura 7).

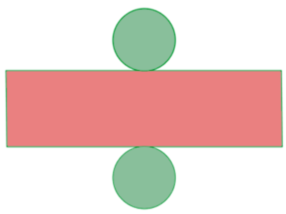

Figura 7: El cilindro

Los elementos de un cilindro recto son los siguientes:

- `Bases del cilindro.-` Son los círculos que conforman los bordes inferior y superior el cilindro. Estos círculos son iguales y paralelos.
- `Ejes del cilindro.-` Es la recta que pasa por los centros de las bases del cilindro; esta es perpendicular a dichas bases. Observa que el eje contiene al lado del rectángulo que gira sobre si mismo.
- `Altura.-` Es la longitud del segmento que tiene por extremos los centros de las dos bases. Es igual al lado del rectángulo que gira sobre si mismo.

- `Generatriz.-` Es el lado opuesto a la altura y es el lado que engendra el cilindro. Observa que la altura, en longitud, es igual a la generatriz (ver figura 8).

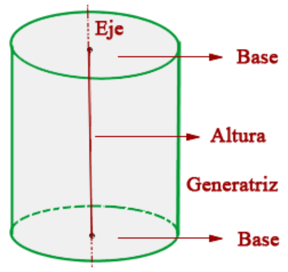

Figura 8: Elementos del cilindro

Se le pide implementar un programa, en lenguaje C, que permita procesar dos cilindros. Para ello debe ingresar solo la altura de cada uno de ellos. Además, debe considerar que la altura de cada cilindro tiene la misma longitud de la circunferencia del círculo que forma su base. Con está información debe calcular el área y volumen de cada uno de ellos y además indicar cuál de los dos cilindros tiene mayor volumen, en caso tengan el mismo volumen debe considerar que el cilindro 1 es el de mayor volumen. Utilice para los cálculos el valor de 3.141592 para PI.

Recuerde

Algunas fórmulas importantes para los cilindros:

- Área lateral del cilindro = $2 * \pi * radio * altura$.
- Área del cilindro = $2 * \pi * radio * (altura + radio)$.
- Volumen del cilindro = $\pi * radio^2 * altura$.
- Longitud de la circunferencia = $2 * \pi * radio$.

Donde radio = radio de la base del cilindro.

A continuación se muestra unos ejemplos de ejecución como casos de prueba:

Caso de Prueba 1:

```
Ingrese la altura del cilindro 1: 125.66
Ingrese la altura del cilindro 2: 95.84

Resultados del Cilindro 1:
Área: 18303.56
Volumen: 157899.73

Resultados del Cilindro 2:
Área: 10647.19
Volumen: 70053.63

El cilindro 1 tiene mayor volumen.
```

Caso de Prueba 2:

```
Ingrese la altura del cilindro 1: 65.24
Ingrese la altura del cilindro 2: 72.57

Resultados del Cilindro 1:
Área: 4933.66
```

```
Volumen: 22096.94

Resultados del Cilindro 2:
Área: 6104.58
Volumen: 30413.16

El cilindro 2 tiene mayor volumen.
```

Caso de Prueba 3:

```
Ingrese la altura del cilindro 1: 65.24
Ingrese la altura del cilindro 2: 65.24

Resultados del Cilindro 1:
Área: 4933.66
Volumen: 22096.94

Resultados del Cilindro 2:
Área: 4933.66
Volumen: 22096.94

El cilindro 1 tiene mayor volumen.
```

Programa 6: Propuesta de solución - El cilindro

```c
#include <stdio.h>
#include <math.h>
#define PI 3.141592

int main(){
	double altura1, altura2, area1, area2, radio1, radio2, volumen1, volumen2;
	int mayor_cilindro1, mayor_cilindro2;

	printf("Ingrese la altura del cilindro 1: ");
	scanf(" %lf",&altura1);
	printf("Ingrese la altura del cilindro 2: ");
	scanf(" %lf",&altura2);

	radio1 = altura1/(2*PI);
	radio2 = altura2/(2*PI);
	area1 = 2*PI*radio1*(altura1 + radio1);
	area2 = 2*PI*radio2*(altura2 + radio2);
	volumen1 = PI*pow(radio1,2)*altura1;
	volumen2 = PI*pow(radio2,2)*altura2;

	mayor_cilindro1 = volumen1 >= volumen2;
	mayor_cilindro2 = volumen2 > volumen1;

	printf("Resultados del Cilindro 1: \n");
	printf("Área: %.2lf\n",area1);
	printf("Volumen: %.2lf\n\n",volumen1);
	printf("Resultados del Cilindro 2: \n");
	printf("Área: %.2lf\n",area2);
	printf("Volumen: %.2lf\n\n",volumen2);
	printf("El cilindro %d tiene mayor volumen",1*mayor_cilindro1 + 2*mayor_cilindro2);
	return 0;
}
```

7. El cono

El cono es el resultante de hacer rotar un triángulo rectángulo de hipotenusa g (la generatriz), cateto inferior r (el radio) y cateto h (altura del cono), alrededor de h. (ver figura 9).

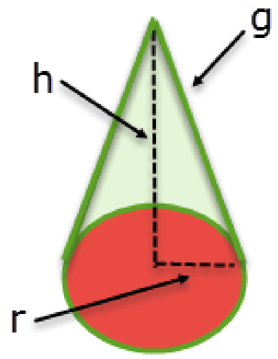

Figura 9: El cono

El área total del cono es el resultado de la suma del área lateral y el área del círculo. Si se puede observar, alguno de los valores de la generatriz, cateto inferio o cateto se podrían calcular directamente por el teorema de pitágoras.

Las fórmulas para calcular el área de un cono son las siguientes:

- Área lateral del cono = $\pi * radio * generatriz$.
- Área del círculo= $\pi * radio^2$.
- Área total del cono = $AreaLateral + AreaCirculo$.
- Volumen del cono = $\frac{1}{3} * \pi * radio^2 * altura$
- Diámetro del círculo = $2 * radio$

Se le pide implementar un programa, en lenguaje C, que permita procesar dos conos. Para ello debe solicitar el ingreso del diámetro de la circunferencia de la base y la generatriz de cada uno de ellos. Con está información debe calcular el área y volumen de cada uno de ellos y además indicar cuál de los dos conos tiene mayor altura (h), en caso tengan la misma altura debe considerar que el cono 1 es el de mayor altura. Utilice para los cálculos el valor de 3.141592 para PI.

A continuación se muestra unos ejemplos de ejecución como casos de prueba:

Caso de Prueba 1:

```
Ingrese el diámetro y la generatriz del cono 1: 32  30
Ingrese el diámetro y la generatriz del cono 2: 25  40

Resultados del Cono 1:
Altura: 25.38
Área: 2312.21
Volumen: 6803.17

Resultados del Cono 2:
Altura: 38.00
Área: 2061.67
Volumen: 6217.20

El cono 2 tiene mayor altura.
```

Caso de Prueba 2:

```
Ingrese el diámetro y la generatriz del cono 1: 25.2  35.4
Ingrese el diámetro y la generatriz del cono 2: 12.4  32.4
```

```
Resultados del Cono 1:
Altura: 33.08
Área: 1900.03
Volumen: 5499.94

Resultados del Cono 2:
Altura: 31.80
Área: 751.85
Volumen: 1280.14

El cono 1 tiene mayor altura.
```

Caso de Prueba 3:

```
Ingrese el diámetro y la generatriz del cono 1: 25.2  35.4
Ingrese el diámetro y la generatriz del cono 2: 25.2  35.4

Resultados del Cono 1:
Altura: 33.08
Área: 1900.03
Volumen: 5499.94

Resultados del Cono 2:
Altura: 33.08
Área: 1900.03
Volumen: 5499.94

El cono 1 tiene mayor altura.
```

Programa 7: Propuesta de solución - El cono

```c
#include <stdio.h>
#include <math.h>
#define PI 3.141592

int main(){
        double altura1, altura2, area1, area2, radio1, radio2, volumen1, volumen2, diametro1, diametro2;
        double generatriz1, generatriz2, areaLateral1, areaLateral2, areaCirculo1, areaCirculo2;
        int mayor_cono1, mayor_cono2;

        printf("Ingrese el diámetro y la generatriz del cono 1: ");
        scanf(" %lf  %lf",&diametro1,&generatriz1);
        printf("Ingrese el diámetro y la generatriz del cono 1: ");
        scanf(" %lf  %lf",&diametro2,&generatriz2);

        radio1 = diametro1/2;
        radio2 = diametro2/2;
        areaLateral1 = PI*radio1*generatriz1;
        areaLateral2 = PI*radio2*generatriz2;
        areaCirculo1 = PI*pow(radio1,2);
        areaCirculo2 = PI*pow(radio2,2);
        area1 = areaLateral1 + areaCirculo1;
        area2 = areaLateral2 + areaCirculo2;
        altura1 = sqrt(pow(generatriz1,2) - pow(radio1,2));
        altura2 = sqrt(pow(generatriz2,2) - pow(radio2,2));
        volumen1 = (PI*pow(radio1,2)*altura1)/3;
        volumen2 = (PI*pow(radio2,2)*altura2)/3;

        mayor_cono1 = altura1 >= altura2;
        mayor_cono2 = altura2 > altura1;

        printf("Resultados del Cono 1: \n");
        printf("Altura:  %.2lf\n",altura1);
        printf("Área Total:  %.2lf\n",area1);
```

```
        printf("Volumen: %.2lf\n\n",volumen1);
        printf("Resultados del Cono 2: \n");
        printf("Altura: %.2lf\n",altura2);
        printf("Área: %.2lf\n",area2);
        printf("Volumen: %.2lf\n\n",volumen2);
        printf("El cono %d tiene mayor altura",1*mayor_cono1 + 2*mayor_cono2);
        return 0;
}
```

8. El prisma

Un prisma cuadrangular es un poliedro cuya superficie está formada por dos cuadriláteros iguales y paralelos llamados bases y por cuatro caras laterales que son paralelogramos.

El `prisma cuadrangular regular` es aquel que tiene como bases dos cuadrados. Sus caras laterales son rectángulos iguales (ver figura 10).

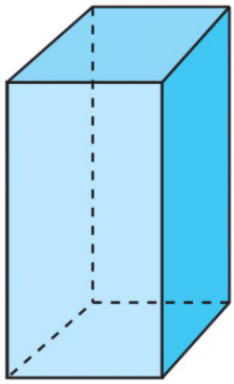

Figura 10: Prisma cuadrangular regular

Se le pide implementar un programa, en lenguaje C, que permita procesar dos prismas cuadrangulares regulares. Para ello debe ingresar el lado de la base y la altura de cada uno de ellos. Con está información debe calcular el área y volumen de cada uno de ellos y además indicar cuál de los dos prismas tiene mayor área, en caso tengan la misma área debe considerar que el prisma 2 es el de mayor área.

Recuerde

Algunas fórmulas importantes para los cilindros:

- Área del prisma cuadrangular regular = $areaLateral + 2 * areaBase$.
- Área lateral del prisma cuadrangular regular = Suma de las áreas de las cuatro caras rectangulares iguales.
- Volumen del prisma cuadrangular regular = $areaBase * altura$.
- Área del rectángulo = $base * altura$.
- Área del cuadrado = $lado^2$.

A continuación se muestra unos ejemplos de ejecución como casos de prueba:

Caso de Prueba 1:

```
Ingrese la base y altura del prisma cuadrangular regular 1: 1.20  4
Ingrese la base y altura del prisma cuadrangular regular 2: 0.85  5.6
```

```
Resultados del Prisma cuadrangular regular 1:
Área: 22.08
Volumen: 5.76

Resultados del Prisma cuadrangular regular 2:
Área: 20.48
Volumen: 4.05

El prisma cuadrangular regular 1 tiene mayor área.
```

Caso de Prueba 2:

```
Ingrese la base y altura del prisma cuadrangular regular 1: 1.57  5.2
Ingrese la base y altura del prisma cuadrangular regular 2: 2.24  4.6

Resultados del Prisma cuadrangular regular 1:
Área: 37.59
Volumen: 12.82

Resultados del Prisma cuadrangular regular 2:
Área: 51.25
Volumen: 23.08

El prisma cuadrangular regular 2 tiene mayor área.
```

Caso de Prueba 3:

```
Ingrese la base y altura del prisma cuadrangular regular 1: 2.24  4.6
Ingrese la base y altura del prisma cuadrangular regular 2: 2.24  4.6

Resultados del Prisma cuadrangular regular 1:
Área: 51.25
Volumen: 23.08

Resultados del Prisma cuadrangular regular 2:
Área: 51.25
Volumen: 23.08

El prisma cuadrangular regular 2 tiene mayor área.
```

Programa 8: Propuesta de solución - El prisma

```c
#include <stdio.h>
#include <math.h>

int main(){
	double base1, base2, altura1, altura2, areaLateral1, areaLateral2, volumen1, volumen2;
	double areaPrisma1, areaPrisma2;
	int mayor_prisma1, mayor_prisma2;

	printf("Ingrese la base y altura del prisma cuadrangular regular 1: ");
	scanf(" %lf %lf",&base1,&altura1);
	printf("Ingrese la base y altura del prisma cuadrangular regular 2: ");
	scanf(" %lf %lf",&base2,&altura2);

	areaLateral1 = 4*base1*altura1;
	areaPrisma1 = areaLateral1 + 2*pow(base1,2);
	volumen1 = pow(base1,2)*altura1;
	areaLateral2 = 4*base2*altura2;
	areaPrisma2 = areaLateral2 + 2*pow(base2,2);
	volumen2 = pow(base2,2)*altura2;
```

```c

    mayor_prisma1 = areaPrisma1 > areaPrisma2;
    mayor_prisma2 = areaPrisma2 >= areaPrisma1;

    printf("Resultados del Prisma cuadrangular regular 1: \n");
    printf("Área: %.2lf\n",areaPrisma1);
    printf("Volumen: %.2lf\n\n",volumen1);
    printf("Resultados del Prisma cuadrangular regular 2: \n");
    printf("Área: %.2lf\n",areaPrisma2);
    printf("Volumen: %.2lf\n\n",volumen2);
    printf("El prisma cuadrangular regular %d tiene mayor área",1*mayor_prisma1 + 2*mayor_prisma2);
    return 0;
}
```

No podrá usar estructuras de control de flujo, como selectivas o iterativas en ambas preguntas.