

Grupo:

Nombres: _____

PROGRAMACIÓN I - GRUPO D

PRÁCTICA Nº2

16 de octubre 2023

A. TEST [3 puntos] – 20 min

A realizar en el campus virtual. 10 preguntas, 20 min.

- Puntuación: Correcta suma 1, no contestado 0, incorrecta resta 0.25 puntos.

B. EJERCICIO PRÁCTICO [7 puntos] – 150 min

Se debe crear un programa, **P2Ejercicio1**, en C++ que funcione como una calculadora de figuras geométricas. El programa permitirá al usuario seleccionar una de las siguientes opciones:

1. Calcular propiedades de un círculo.
2. Calcular propiedades de un triángulo.
3. Calcular propiedades de un rectángulo.
4. Calcular propiedades de un cuadrado.
5. Calcular propiedades de un polígono regular.
6. Salir del programa.

Si el usuario selecciona cualquier opción del 1 al 5, se ejecutará lo indicado en cada punto y al finalizar se volverá a mostrar el menú inicial.

Para guardar la información, **utiliza estructuras** para almacenar los campos necesarios y relevantes para cada figura. El programa deberá ofrecer las siguientes funcionalidades, dependiendo de las opciones elegidas:

1. CIRCULO:

- Permitir al usuario ingresar el radio o el diámetro del círculo. El programa deberá dar las dos opciones, y que el usuario elija. La otra medida deberá obtenerse a partir de la primera, y guardarse en el campo correspondiente de la estructura.
- Calcular, almacenar y mostrar el área del círculo y el perímetro (circunferencia) del círculo.

2. TRIÁNGULO:

- Permitir al usuario ingresar los tres lados de un triángulo. Comprobar, mediante la desigualdad triangular, si esos tres lados pueden formar un triángulo. En caso contrario, volver a pedir las 3 longitudes.
 - o **Desigualdad triangular:** si tienes tres longitudes de lados a , b y c , entonces para que puedan formar un triángulo, deben cumplirse las siguientes condiciones:
 - $a + b > c$
 - $a + c > b$
 - $b + c > a$
- Calcular, almacenar y mostrar el área y el perímetro del triángulo.
- Determinar la clase del triángulo (equilátero, isósceles o escaleno) en función de los lados ingresados.

3. CUADRADO:

- Permitir al usuario ingresar el lado de un cuadrado.
- Calcular, almacenar y mostrar el área y el perímetro del cuadrado.

4. RECTÁNGULO:

- Permitir al usuario ingresar la longitud y el ancho de un rectángulo. Si la longitud y el ancho son iguales, informar al usuario que se trata de un cuadrado y volver a pedir las longitudes.
- Calcular, almacenar y mostrar el área del rectángulo y el perímetro del rectángulo.

5. POLÍGONO REGULAR GENÉRICO:

- Permitir al usuario ingresar el número de lados y la longitud de cada lado del polígono regular. Se debe determinar el nombre del polígono regular en función del número de

lados ingresados (por ejemplo, un triángulo, un cuadrado, un pentágono, etc. Llegar hasta el nonágono).

- Mostrar el nombre del polígono. Calcular, almacenar y mostrar el área y el perímetro del polígono regular.

6. El programa se ejecutará hasta que el usuario seleccione la opción de salida.

Solución propuesta:

```
#include <iostream>
#include <cmath>
using namespace std;

struct Circulo {
    double radio;
    double diametro;
    double area;
    double perimetro;
};

struct Triangulo {
    double lado1;
    double lado2;
    double lado3;
    double area;
    double perimetro;
    string tipo;
};

struct Cuadrado {
    double lado;
    double area;
    double perimetro;
};

struct Rectangulo {
    double longitud;
    double ancho;
    double area;
    double perimetro;
};

struct PoligonoRegular {
    int lados;
    double longitudLado;
    string nombre;
    double area;
    double perimetro;
};
```

```
int main() {
    int opcion = 0;

    while(opcion!=6)
    {
        cout << "Seleccione una opción:" << endl;
        cout << "1. Calcular propiedades de un círculo." << endl;
        cout << "2. Calcular propiedades de un triángulo." << endl;
        cout << "3. Calcular propiedades de un cuadrado." << endl;
        cout << "4. Calcular propiedades de un rectángulo." << endl;
        cout << "5. Calcular propiedades de un polígono regular." << endl;
        cout << "6. Salir del programa." << endl;
        cin >> opcion;

        switch (opcion) {
            case 1: {
                Circulo circulo;
                cout << "Seleccione una opción:" << endl;
                cout << "1. Ingresar radio." << endl;
                cout << "2. Ingresar diámetro." << endl;
                int opcionCirculo;
                cin >> opcionCirculo;
                if (opcionCirculo == 1) {
                    cout << "Ingrese el radio del círculo: ";
                    cin >> circulo.radio;
                } else if (opcionCirculo == 2) {
                    cout << "Ingrese el diámetro del círculo: ";
                    cin >> circulo.diametro;
                    circulo.radio = circulo.diametro / 2;
                }
                circulo.area = M_PI * pow(circulo.radio, 2);
                circulo.perimetro = 2 * M_PI * circulo.radio;
                cout << "Área del círculo: " << circulo.area << endl;
                cout << "Perímetro del círculo: " << circulo.perimetro << endl;
                break;
            }

            case 2: {
                Triangulo triangulo;
                cout << "Ingrese los tres lados del triángulo:" << endl;
                cin >> triangulo.lado1 >> triangulo.lado2 >> triangulo.lado3;
                if (triangulo.lado1 + triangulo.lado2 > triangulo.lado3 &&
                    triangulo.lado1 + triangulo.lado3 > triangulo.lado2 &&
                    triangulo.lado2 + triangulo.lado3 > triangulo.lado1) {
                    double s = (triangulo.lado1 + triangulo.lado2 + triangulo.lado3) /
2;
                    triangulo.area = sqrt(s * (s - triangulo.lado1) * (s -
triangulo.lado2) * (s - triangulo.lado3));
                    triangulo.perimetro = triangulo.lado1 + triangulo.lado2 +
triangulo.lado3;
                }
            }
        }
    }
}
```

```
        if (triangulo.lado1 == triangulo.lado2 && triangulo.lado2 ==
triangulo.lado3) {
            triangulo.tipo = "Equilátero";
        } else if (triangulo.lado1 == triangulo.lado2 || triangulo.lado1 ==
triangulo.lado3 || triangulo.lado2 == triangulo.lado3) {
            triangulo.tipo = "Isósceles";
        } else {
            triangulo.tipo = "Escaleno";
        }
        cout << "Área del triángulo: " << triangulo.area << endl;
        cout << "Perímetro del triángulo: " << triangulo.perimetro <<
endl;

        cout << "Tipo de triángulo: " << triangulo.tipo << endl;
    } else {
        cout << "Estos lados no forman un triángulo." << endl;
    }
    break;
}

case 3: {
    Cuadrado cuadrado;
    cout << "Ingrese el lado del cuadrado: ";
    cin >> cuadrado.lado;
    cuadrado.area = cuadrado.lado * cuadrado.lado;
    cuadrado.perimetro = 4 * cuadrado.lado;
    cout << "Área del cuadrado: " << cuadrado.area << endl;
    cout << "Perímetro del cuadrado: " << cuadrado.perimetro <<
endl;

    break;
}

case 4: {
    Rectangulo rectangulo;
    cout << "Ingrese la longitud del rectángulo: ";
    cin >> rectangulo.longitud;
    cout << "Ingrese el ancho del rectángulo: ";
    cin >> rectangulo.ancho;
    if (rectangulo.longitud == rectangulo.ancho) {
        cout << "Esto es un cuadrado. Por favor, ingrese las
dimensiones nuevamente." << endl;
    } else {
        rectangulo.area = rectangulo.longitud * rectangulo.ancho;
        rectangulo.perimetro = 2 * (rectangulo.longitud +
rectangulo.ancho);
        cout << "Área del rectángulo: " << rectangulo.area << endl;
        cout << "Perímetro del rectángulo: " << rectangulo.perimetro <<
endl;

    }
    break;
}
```

```
case 5: {
    PoligonoRegular poligono;
    cout << "Ingrese el número de lados del polígono regular: ";
    cin >> poligono.lados;
    cout << "Ingrese la longitud de cada lado del polígono regular: ";
    cin >> poligono.longitudLado;
    if (poligono.lados >= 3 && poligono.lados <= 9) {
        if (poligono.lados == 3) {
            poligono.nombre = "Triángulo";
        } else if (poligono.lados == 4) {
            poligono.nombre = "Cuadrado";
        } else if (poligono.lados == 5) {
            poligono.nombre = "Pentágono";
        } else if (poligono.lados == 6) {
            poligono.nombre = "Hexágono";
        } else if (poligono.lados == 7) {
            poligono.nombre = "Heptágono";
        } else if (poligono.lados == 8) {
            poligono.nombre = "Octágono";
        } else if (poligono.lados == 9) {
            poligono.nombre = "Nonágono";
        }
        poligono.area = (poligono.lados * pow(poligono.longitudLado,
2)) / (4 * tan(M_PI / poligono.lados));
        poligono.perimetro = poligono.lados * poligono.longitudLado;
        cout << "Nombre del polígono: " << poligono.nombre << endl;
        cout << "Área del polígono: " << poligono.area << endl;
        cout << "Perímetro del polígono: " << poligono.perimetro <<
endl;
    } else {
        cout << "Número de lados no válido." << endl;
    }
    break;
}

case 6:
    cout << "Saliendo del programa." << endl;
    break;

default:
    cout << "Opción no válida. Por favor, seleccione una opción
válida." << endl;
    break;
}
}

return 0;
}
```

IMPORTANTE:

No se considerará el área del polígono para la nota de la práctica. Al o incluir la apotema, el cálculo se complicaba.

Para aquellos que estén interesado en saber cómo se calcula sin la apotema:

$$\text{Área} = \frac{\text{Perímetro} \cdot \text{Apotema}}{2} = \frac{nx \cdot x \tan \left(90^\circ - \frac{360^\circ}{2n} \right)}{2 \cdot 2}$$

<https://elmatenavegante.blogspot.com/2017/02/la-formula-sin-apotemas.html>

Para el triángulo, se ha usado la fórmula de Herón. Para la práctica, se tendrá en cuenta el cálculo del área cogiendo dos lados cualesquiera para sacar la altura.

RUBRICA DE CALIFICACIÓN

- El 30% de la práctica 2, simulacro de examen, vendrá dado por el test (10 preguntas, 20 minutos).
- El 70% restante será de la práctica entregada (misma nota para cada integrante de manera general), siguiendo la siguiente rúbrica para cada apartado:

Se debate en equipo sobre la estrategia a seguir, haciendo un boceto de la estructura que tendrá el programa (a papel, en la siguiente hoja).	1 pto
El programa muestra las diferentes opciones del menú, permitiendo al usuario seleccionar una opción, y repitiéndose hasta que se pulse salir.	1 ptos
Se crean las estructuras necesarias para el programa.	1 pto
Se pide al usuario que introduzca la información necesaria en cada apartado y se guarda en los campos correspondientes de su estructura.	1 pto
Se calcula, almacena y se muestra el área y el perímetro de todas las figuras.	1 pto
Se comprueba que los 3 lados del triángulo permiten formarlo y se repite la petición de datos si es necesario.	1 pto
Se determina la tipología de triángulo.	1 pto

Para cada apartado, se tendrá en cuenta lo siguiente:

% máx. (*)

El elemento evaluable no compila o no se asemeja a lo que se pide	0%
El elemento evaluable no se aproxima suficientemente a lo pedido	40%
El elemento evaluable se aproxima suficientemente a lo pedido	60%
El elemento evaluable funciona correctamente y las estrategias y elementos de código elegidos son adecuados.	100%

(*) El porcentaje (% max.) representa el valor máximo sobre el que se evalúa el elemento indicado.

Tip: La struct TRIANGULO tendría los campos: lado1, lado2, lado3, area, perímetro.

Boceto/Diagrama del programa (1 punto):