

## Práctico 3

### Errores típicos en C++

1. En el siguiente fragmento de código en C++, determinar dónde está el error, qué tipo de error es (*lógico, de sintáxis, etc*) y proponer una versión del código corregida.

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

int main() {
    int x = 5;
    int y = 0;
    cout << x / y << endl;

    return 0;
}
```

2. En el siguiente fragmento de código, determinar dónde está el error, qué tipo de error es (*lógico, de sintáxis, etc*) y proponer una versión del código corregida.

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

int main() {
    int x;
    cout << x << endl;
    return 0;
}
```

3. En el siguiente fragmento de código, determinar dónde está el error, qué tipo de error es (*lógico, de sintáxis, etc*) y proponer una versión del código corregida.

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

int main() {
    int x = 5;
    if (x = 0) {
        cout << "x es cero" << endl;
    } else {
        cout << "x no es cero" << endl;
    }
    return 0;
}
```

4. En el siguiente fragmento de código, determinar dónde está el error, qué tipo de error es (*lógico, de sintáxis, etc*) y proponer una versión del código corregida. ¿Qué ocurriría en Python?

---

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

int main() {
    string frase = "Hay un cocodrilo en mi patio" ;

    cout << frase[-1] << endl;

    return 0;
}
```

---

## Errores típicos en Python

5. En el siguiente fragmento de código, determinar dónde está el error, qué tipo de error es (*lógico, de sintáxis, etc*) y proponer una versión del código corregida.

---

```
mis_numeros = [1, 2, 3]
print(mis_numeros[3])
```

---

6. En el siguiente fragmento de código, determinar dónde está el error, qué tipo de error es (*lógico, de sintáxis, etc*) y proponer una versión del código corregida.

---

```
frase = 'Hola Pepe, tanto tiempo!'
n = len(frase)

for pos in range(n):
    print(frase[pos])

print(frase[n])
```

---

7. En el siguiente fragmento de código, determinar dónde está el error, qué tipo de error es (*lógico, de sintáxis, etc*) y proponer una versión del código corregida.

---

```
x = "5"
y = 3
print(x + y)
```

---

8. En el siguiente fragmento de código, determinar dónde está el error, qué tipo de error es (*lógico, de sintáxis, etc*) y proponer una versión del código corregida. (*Investigue qué es un diccionario en Python*)

---

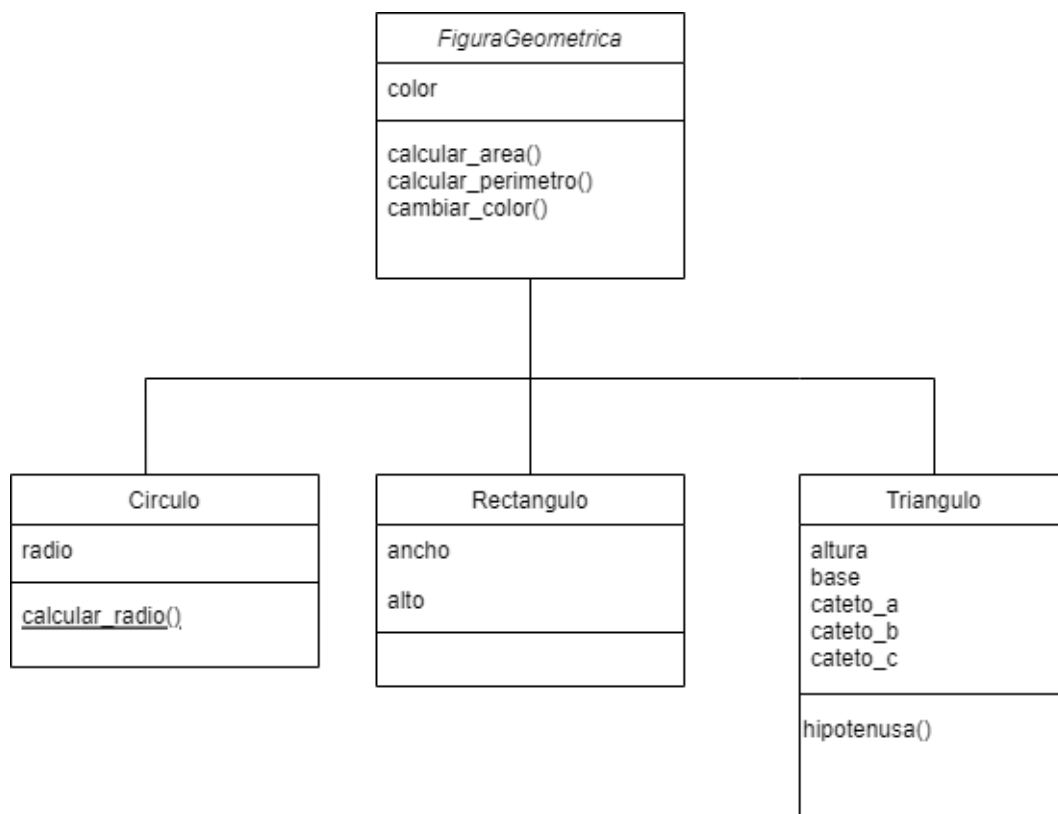
```
my_dict = {"a": 1, "b": 2}
print(my_dict["c"])
```

---

## Sobrecarga y Polimorfismo

La sobrecarga puede ser tanto de operadores (*por ejemplo el +*) como de funciones. *Polimorfismo es esencialmente la sobrecarga de métodos*. Realice los siguientes códigos, tanto una versión en C++ como otra versión en Python.

9. Crear una clase Animal, y dos clases hijas Perro y Gallo. La clase padre Animal tendrá un método llamado *chillar()* que imprimirá en pantalla un mensaje 'El animal ha chillado!'. Sobrecargar el método en las clases hijas, por la impresión de una onomatopeya acorde al animal en cuestión.
10. Crear una clase FiguraGeometrica, que contenga los siguientes métodos *calcular\_area()*, *calcular\_perimetro()*, *cambiar\_color()*. Crear 3 clases hijas, Circulo, Rectángulo y Triángulo. Del estilo del diagrama de clases que se muestra a continuación. Corroborar si cada clase tiene todos los atributos y métodos que se consideran necesarios.



11. Crear la clase Complejo para representar un número complejo. Definirle los atributos real y imaginaria, privados. (*Encapsulamiento*) Reflexione cuáles son todos los métodos que esta clase debería proveer para el manejo de un número complejo. Por ejemplo, ¿Qué operaciones debo poder realizar sobre un número complejo, o sobre dos? Implementar. Finalmente, sobrecargar el operador + y el - para que puedan realizar la suma y la resta respectivamente de dos números complejos.
12. Crear la clase Vector para representar un vector matemático de 3 coordenadas. Tendrá 3 atributos: x, y, z. Reflexione qué métodos debería tener esta clase que represente a un

vector matemático y las operaciones que se pueden realizar con él. Implementelo. Realice una sobrecarga de los operadores  $+$  y  $-$ .