

Laboratorio 3: Herencia, polimorfismo, sobrecarga

M. Sc. Ricardo Román Brenes - `ricardo.roman@ucr.ac.cr`

II-2019

Tabla de contenidos

1. Enunciado	1
1.1. Vértices	1
1.2. Clase base	2
1.3. Clases derivadas	2
1.4. Entrada / Salida	3
1.5. Diagrama de clases	3
2. Consideraciones	3

1. Enunciado

Diseñe, desarrolle y documente (Doxygen) una serie de clases que modelen figuras geométricas en dos dimensiones en **Python**.

1.1. Vértices

Cree una clase **Vertice**, la cual modele un punto en un espacio 2D. Esta clase debe tener como atributos:

- decimal x
- decimal y
- entero identificador

Y como métodos:

- `__init__()`

- `__invert__()` // Transforma el objeto en una representación en texto.
- `double __rshift__(other)` // Calcula la distancia entre 2 vértices.

1.2. Clase base

Cree una clase base llamada **Figura** que tenga como atributos:

- texto nombre
- texto color
- entero identificador
- lista puntos // lista de vertices que forman la figura.
- Un número **estático** autoincrementado cada vez que se genera una nueva figura.

Y como métodos:

- `__init__()`
- `@abstractmethod decimal superficie()`
- `@abstractmethod decimal perimetro()`
- `__invert__()` // Transforma el objeto en una representación en texto.

1.3. Clases derivadas

Además cree clases para las siguientes figuras:

- Círculo
- Rectángulo
- Triangulo

Estas clases heredan de la clase Figura y contienen sus características propias (equiláteros, radio, largo del lado...) y además reimplementan los métodos virtuales puros de la clase base. Además la clase triángulo debe tener otras 3 especializaciones:

- Equilátero
- Isósceles
- Escaleno

Que a su vez reimplementan los métodos virtuales puros que la clase Triangulo les pasa de Figura.

1.4. Entrada / Salida

Cree una clase llamada `Impresora`. Esta clase se encarga de imprimir los objetos, sin importar su tipo, así como escribirlos en archivos de texto. Así, sus métodos son:

- `imprimirObjeto(obj)`
- `escribirObjeto(obj, rutaArchivo)`

También el operador `=` para todas las clases.

1.5. Diagrama de clases

Por último construya el diagrama de clases de estas cuatro clases siguiendo el estándar de UML¹.

2. Consideraciones

- Haga grupos de 2 o 3 personas.
- Genere un reporte PDF en \LaTeX que incluya su código, el diagrama de clases, y sus conclusiones.
- Suba su código y documentación (doxygen, README, INSTALL) al GitLab respectivo de su grupo y laboratorio.
- Cada estudiante debe subir el reporte a Schoology
- Recuerde que por cada día tardío de entrega se le rebajaran puntos de acuerdo con la formula: 4^d , donde $d > 1$ es la cantidad de días tardíos.

¹https://en.wikipedia.org/wiki/Unified_Modeling_Language