

## Tarea VII: Programación Orientada a Prototipos (5 pts)

### Implementación:

1. (2 pts) – Propiedades y herencia.

- Considere un objeto con un constructor **Circulo**, con un único campo **radio**. Nota: Todo este problema debe ser implementado exclusivamente con **programación orientada a prototipos**.
  - a) (0.5 pts) – Defina una función **Circulo** que reciba un número para el **radio** y pueda usarse con **new** para crear un nuevo objeto con **radio** como campo. En caso de que el valor propuesto sea negativo o no sea un número, se debe arrojar un error con el mensaje: ``Radio invalido'``.
  - b) (0.25 pts) – Amplíe el constructor **Circulo** de manera que implemente para cada objeto creado con el constructor un método **area** que retorne el área del mismo.
- Considere una función **Cilindro** de **Circulo**, que agrega un único campo **altura**.
  - a) (0.75 pts) – Defina una función **Cilindro** que reciba dos números para el radio y altura y puede usarse con **new** para generar un nuevo objeto cilindro con dichos números. Primero, la función debe usar el constructor **Circulo** para crear un objeto con el radio propuesto. Luego, debe crearse un objeto que use dicho círculo como prototipo, y le sea definido el nuevo campo **altura**. En caso de que la altura propuesta sea negativo, se debe arrojar un error con el mensaje: ``Altura invalida'``. Finalmente, esta función debe retornar el objeto representando el cilindro. Nótese que esta función, al contrario de **Circulo**, no será usada como constructor.
  - b) (0.5 pts) – Amplíe la función **Cilindro** de manera que implemente para cada objeto creado con el un método **Cilindro** que retorne el volumen del cilindro. En el cálculo deben usar el método **area** que debió haber sido heredado del prototipo.

### Investigación:

(3 pts) – Considere un lenguaje de programación puramente orientado a objetos, donde una clase  $B$  hereda de otra clase  $A$  (esto es,  $B$  es subclase de  $A$ ).

- a) (1.5 pts) – Considere una clase **Lista**, parametrizable en el tipo de sus elementos. ¿Qué relación de herencia, de haberla, tienen **Lista<A>** y **Lista<B>**?

- (0.5 pts) – ¿Qué decisión toma el lenguaje *Java*? Explique dicha decisión.
  - (0.5 pts) – ¿Qué decisión toma el lenguaje *Scala*? Explique dicha decisión.
  - (0.5 pts) – Suponiendo que existe una clase `Bottom`, la cual es subclase de todas las demás clases: ¿Cuál es el tipo inferido de la lista vacía? Justifique su respuesta.
- b) (1.5 pts) – Considere ahora que el lenguaje de programación en el que se está trabajando es funcional. Como es puramente orientado a objetos, las funciones también deben ser objetos. Suponga otra clase cualquiera  $C$ .
- (0.75 pts) – ¿Qué relación de herencia, de haberla, tienen las funciones con firmas  $A \rightarrow C$  y  $B \rightarrow C$ ? Justifique su respuesta.
  - (0.75 pts) – ¿Qué relación de herencia, de haberla, tienen las funciones con firmas  $C \rightarrow A$  y  $C \rightarrow B$ ? Justifique su respuesta.

## Patrones de Diseño

## Detalles de la Entrega

La entrega de la tarea consistirá de un único archivo `t7_<carné1>&<carné2>.pdf`, donde `<carné1>` y `<carné2>` son los números de carné de los integrantes de su equipo. Tal archivo debe ser un documento PDF con su implementación para las funciones pedidas y respuestas para las preguntas planteadas. Por ejemplo, si el equipo está conformado por 00-00000 y 11-11111, entonces su entrega debe llamarse: `t7_00-00000&11-11111.pdf`.

La tarea deberá ser entregada a *ambos* profesores encargados del curso (Carlos Infante y Alexander Romero), *únicamente* a sus direcciones de correo electrónico: ( carlosinfante1220@gmail.com y alexanderromero360@gmail.com) a más tardar el Sábado 20 de Marzo, a las 11:59pm. VET. Por favor colocar **[CI3661] Entrega Tarea VII** en el asunto del correo. Esto será usado por los profesores para filtrar los correos de las entregas, por lo que correos que no tengan este asunto no serán revisados.