



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

IIC2233 - Programación Avanzada
2° semestre 2015

Actividad 09

Metaclases

Instrucciones

Deberá utilizar Metaclases para implementar funcionalidades en el sistema presentado a continuación.

Las viles autoridades de una universidad muy lejana desean infiltrarse en el nuevo sistema de toma de ramos: “Bummer UC”, con el fin de hacerlo fallar y poder volver al sistema anterior, haciéndolo pasar por una accidental “división por cero”. Para esto, necesitan la ayuda de gente que provenga del grupo de estudiantes (ya que así no se hacen responsables si es descubierto), y han escogido a los mejores programadores para esta traicionera tarea, entre ellos: usted. Amenazado con no poder botar Programación Avanzada, usted acepta y debe seguir las órdenes del malvado consejo.

Las autoridades no desean que se metan entre sus redes, por lo que han decidido implementar un robot que desconecte a cualquier “hacker” que intente acceder. A usted se le ordenó implementar ciertas funcionalidades del robot, pero hay un pequeño detalle: El robot conoce tan bien la red, que sería muy peligroso dejar que el resto de programadores de su equipo pudieran tener acceso a las funcionalidades que debe implementar.

Requerimientos

Debe programar la metaclase MetaRobot, que deberá agregar los siguientes métodos, atributos y comportamientos a la clase Robot (dentro del main.py entregado junto a este enunciado):

- Los atributos (que deben ser fijos o estáticos):
 - creador: Es el id de GitHub del creador del robot, es decir, tu id de GitHub (Para que nadie le robe el crédito).
 - ip_inicio: Es la dirección de IP desde donde se inicializa el robot. La dirección es: ”190.102.62.283” (para que ningún programador conspire contra otra persona y le eche la culpa reemplazando su IP).
- Los métodos:
 - check_creator: Este método verifica que el creador del robot se encuentre en la lista de programadores del robot (si, los líderes desconfían de que usted pueda inculpar a alguien fuera del equipo de programadores). Debe imprimir en consola un mensaje indicando si el creador se encuentra en la lista de programadores o no.

- `cortar_conexion`: Mediante este método, el robot puede cortar la conexión de algún hacker que se encuentre en el mismo puerto que el robot. Si es que encuentra un hacker en el puerto se debe imprimir en consola un mensaje avisando de esta situación, y que se le ha cortado la conexión (es decir, el atributo 'hacker' del puerto se cambió a 0).
- `cambiar_nodo`: Mediante este método, el robot puede cambiar de nodo (puerto), a cualquiera que se ingrese. Debe imprimir en consola un mensaje indicando de que nodo proviene y cuál es su destino.

Por último, solo la clase `Robot` podrá ser construida a partir de la metaclassa `MetaRobot`. En caso de que otra clase trate de ser construida a partir de `MetaRobot` se debe levantar un error.

Notas

- *Está prohibido modificar el archivo `main.py` entregado junto a este enunciado. Todo lo pedido debe realizarse mediante la metaclassa `MetaRobot`.*

To - DO

- (2.00 pts) Chequear que solo la clase `Robot` pueda ser construida desde `MetaRobot`.
- (2.00 pts) Implementar atributos pedidos a `MetaRobot`.
- (2.00 pts) Implementar métodos pedidos a `MetaRobot`.



Figura 1: Si, tú.