

Examen parcial # 2: *Python y OOP*

Universidad de San Carlos de Guatemala
Facultad de Ingeniería
EIME - IPC1 (0769) "N"

Ing. Iván René Morales
1º Semestre 2014

¿Qué debe hacer?

Utilizando lenguaje Python 2.7 implementar un juego similar al presentado en *holaPython11.py* (número secreto). Ahora se utilizará otro acercamiento para plantear el problema. Usted deberá crear un *mundo* de dos dimensiones compuesto por un muro perimetral y el área de juego (todo lo que quede dentro del muro). En el área de juego deben existir dos objetos: un tesoro y un jugador (el usuario). Ambos deben aparecer en una ubicación aleatoria dentro del área de juego al momento de iniciar el juego.

El objetivo es que el usuario *atrape* el tesoro, moviéndose en una dirección (saltando uno o dos espacios) cada turno. La dificultad es que el tesoro también es capaz de moverse (un espacio) cada vez que el usuario lo hace.

Restricciones y Limitaciones

Se le proveerá a usted un código base para que implemente el proyecto. Deberá utilizar como mínimo las dos clases que se proponen (*mundo* y *objeto*). Dentro de las clases existe también un machote de los posibles métodos a utilizar, aunque el único que debe implementarse forzosamente es *imprimeMundo(self)* (aparte del constructor). Es necesario que utilice constantes para las declaraciones como *tamaño inicial del mundo*, *símbolo que representa a cada objeto*, etc. Puede utilizar únicamente los comandos básicos incluidos en Python, así como los módulos *Random* y *Time* (de ser necesarios).

¿Cómo jugar?

Inicialmente deben imprimirse las instrucciones (esto solo se hace una vez). Seguido, deberá mostrarse el *mundo* o mapa de juego inicial, incluyendo a los dos

```
+-----+  
+                                             +  
+          0                               x      +  
+                                             +  
+                                             +  
+                                             +  
+                                             +  
+                                             +  
+                                             +  
+                                             +  
+                                             +  
+                                             +  
+                                             +  
+                                             +  
+-----+  
Pasos a saltar (1,2)           --> 2  
Direccion ('U','D','L','R') --> R
```

Al confirmar los datos de movimiento, el objeto que simboliza al usuario debe moverse según las instrucciones y el tesoro deberá tomar una posición aleatoria que esté a un salto de distancia (no son válidos movimientos en diagonal). Para poder visualizar el cambio, deberá imprimir nuevamente el mapa completo, y pedir al usuario que ingrese el nuevo movimiento que desea hacer. La figura 2 es una muestra de lo que podría implementarse.

Documentos de ayuda

- Video en *YouTube* con un ejemplo del funcionamiento:
<http://youtu.be/-1qV5GVg0wg>
- Código fuente base para iniciar el proyecto (machote):
<http://goo.gl/Sjo9Az>

Metodología de entrega

Con el fin de agilizar el proceso de calificación y evitar hacinamiento de los alumnos, la entrega será a través de un formulario en la web, pero la calificación individual se realizará por bloques (divididos por números de carnet). Esta calendarización será publicada luego de la fecha de entrega.

El proyecto es trabaja de forma individual, y cualquier indicio de copia (parcial o total) será motivo de anulación inmediata. El límite de la entrega del proyecto es el domingo 20/04/2014 a las 18:00 hrs. Este proceso será via web: <http://goo.gl/iCLRZj>. Puede enviar su examen solamente una vez, de lo contrario no será calificado. *Todo debe estar debidamente indentado y comentado.* Al momento de la calificación deberá lleva 2 COPIAS IMPRESAS de la última página de este documento. No es necesario que lleve computadora en este instante.

Calificación

Carnet: _____

| Descripción | Calificación |
|---------------------------|--------------|
| Uso adecuado de clases | /40 |
| Implementación de métodos | /20 |
| Restricciones de juego | /20 |
| Interfaz del usuario | /20 |

Nota:

Calificó: