



IIC2115 – Programación como Herramienta para la Ingeniería (I/2022)

Laboratorio 1

Aspectos generales

- **Objetivo:** evaluar individualmente el aprendizaje sobre el uso de técnicas de POO y estructuras de datos para generar una simulación eficiente.
- **Lugar de entrega:** lunes 4 de abril a las 23:59 hrs. en repositorio privado.
- **Formato de entrega:** archivo Python Notebook (**L1.ipynb**) con la solución del laboratorio. El archivo debe estar ubicado en la carpeta **L1**. Utilice múltiples celdas de texto y código para facilitar el trabajo del cuerpo docente. Laboratorios que no cumplan el formato de entrega tendrán un descuento de 0,5 pts.
- **Entregas atrasadas:** El descuento por atraso se realizará de acuerdo a lo definido en el programa del curso. Si su laboratorio es entregado fuera de plazo, tiene hasta el **martes 05 de abril a las 11:59 AM** para responder el formulario de **entregas fuera de plazo** disponible en el Syllabus.
- **Issues:** Las discusiones en las *issues* del Syllabus que sean relevantes para el desarrollo del laboratorio, serán destacadas y se considerarán como parte de este enunciado. Así mismo, el uso de librerías externas que solucionen aspectos fundamental del laboratorio no podrán ser utilizadas. Solo se podrán utilizar las que han sido aprobadas en las *issues*, previa consulta de los estudiantes.
- **Laboratorios con errores de sintaxis y/o que generen excepciones en todas las ejecuciones** serán calificados con **nota 1.0**.

El problema: Battle Royale Lento

De manera espontánea, los caracoles del planeta Tierra están siendo teletransportados a un planeta lejano, donde una raza de extraterrestres adictos a la adrenalina organiza un evento de supervivencia. Su labor en este laboratorio será el desarrollo de una simulación del evento, que permita que los caracoles terrícolas que aún no hayan sido teletransportados puedan evaluar distintas estrategias de supervivencia.

Instrucciones

En el evento de supervivencia, una cantidad arbitraria de caracoles son liberados en ubicaciones aleatorias de una ciudad abandonada, que esconde numerosas recompensas y trampas peligrosas. La ciudad además que está siendo invadida por un ejército de termitas, que comen todo lo que encuentran (incluidos caracoles) en su viaje al centro de la ciudad.

El objetivo de los caracoles es *correr* por sus vidas y llegar a la zona protegida ubicada en el centro de la ciudad, donde las termitas no pueden entrar. El caracol ganador será aquel que, habiendo llegado a la zona protegida una vez que las termitas hayan terminado de comer el resto de la ciudad, haya recolectado la mayor cantidad de puntos de recompensa.

Especificación de la ciudad

La ciudad es descrita a través de un mapa en forma de grilla regular, donde cada elemento de esta corresponde a uno de los siguientes elementos:

1. Espacio transitable. Se puede caminar sobre él. Se representa con un “_”.
2. Obstáculo. No se puede pasar sobre el. Se representa con una “X”.
3. Diamante: Se representa con la letra “D” y entrega 5 puntos.
4. Moneda: Se representa con la letra “M” y entrega 1 punto.
5. Cofre misterioso: Se representa con la letra “F”. De manera aleatoria, entrega 10 puntos o quita 3 puntos. La probabilidad de ocurrencia debe ser fijada por ud.
6. Velocidad: Se representa con una “V” y aumenta al doble la velocidad del caracol durante una cantidad acotada de tiempo, que debe ser definida por ud.
7. Zona protegida: se representa con una “Z”. Esta zona se debe ubicar aproximadamente al centro del mapa y no puede tener una superficie mayor a 8 celdas.

Requerimientos del laboratorio

Su misión en este laboratorio será construir una simulación de la situación antes descrita, en base a los conocimientos adquiridos sobre POO y estructuras de datos. El programa deberá considerar como entrada el mapa de la ciudad, representado como una lista de listas, y el número inicial de caracoles participantes, a partir de lo cual deberá simular la evolución del evento, actualizando el mapa adecuadamente. Concretamente, el programa deberá cumplir lo siguiente:

- Caracoles: debe considerar al menos 3 tipos de caracoles en su simulación, asignando aleatoriamente un tipo al momento de ser ubicados en el mapa. Los aspectos en que se diferencian quedan a su elección.
- Posición inicial: asigne a cada caracol una posición aleatoria (pero válida) en el mapa, desde la cual comenzar a explorar. Cabe aclarar que dos caracoles no pueden estar en la misma posición. Los caracoles se representan con el carácter “C” y un número natural positivo (*e.g.* “C1” para el primer caracol). Luego de asignar las posiciones iniciales, se debe imprimir el estado actual del mapa en pantalla. Es importante que el formato visual del mapa sea claro, entendible y mantenga correctamente el alineamiento de filas y/o columnas.
- Movimientos: considere que los movimientos de los caracoles son discretos, es decir, se mueven cierta cantidad de celdas contiguas por unidad de tiempo. Además, ningún caracol puede tener preferencia sobre otro, lo que implica que todos deben realizar su movimiento de manera simultánea por cada unidad de tiempo (a menos que un caracol decida no moverse). Finalmente, considere que en ningún momento de la simulación puede haber más de un caracol por celda.
- Termitas: cada movimiento de las termitas consistirá en la desaparición de todas las casillas de uno de los cuatro bordes externos del mapa. Cualquier caracol que se haya encontrado en el borde eliminado, debe considerarse como muerto y se retira de la simulación. La velocidad de avance de las termitas debe ser definida por ud.
- Eventos aleatorios: debe considerar al menos 4 tipos de eventos aleatorios en su simulación. Estos pueden ser positivos o negativos para los caracoles, y pueden ocurrir en cualquier momento.
- Visualización: el mapa debe ser actualizado en pantalla cada vez que hay un cambio relevante en él. Con el fin de facilitar la visualización, se recomienda actualizar el mapa cuando todos los caracoles y las termitas hayan tenido la oportunidad de hacer un movimiento. Además de esto, cada evento debe ser indicado mediante una descripción textual.

Estrategia de los caracoles

En cada momento, cada uno de los caracoles debe tener como objetivo moverse a un punto específico del mapa. Este punto puede ser una recompensa, un cofre misterioso o la zona segura. La estrategia de cada caracol debe ser óptima con respecto a su objetivo actual, es decir, su movimiento en esa dirección debe ser óptimo en la cantidad de celdas recorridas. Si en alguna parte del recorrido el objetivo deja de ser válido (por ejemplo, otro caracol recogió la recompensa), el caracol debe buscar otro objetivo, cambiando adecuadamente su planificación.

Ejemplo de mapa

El siguiente es un ejemplo muy acotado de como podría verse un mapa en su simulación. Puede basarse en él para comenzar a trabajar, pero su solución debiera ser suficientemente general para trabajar con mapas distintos.

```
[ [ 'F', '-', '-', '-', '-', 'X', 'M' ],  
  [ '-', '-', '-', '-', 'F', 'X', '-' ],  
  [ 'X', 'X', 'X', 'Z', '-', 'X', '-' ],  
  [ '-', 'F', '-', 'Z', '-', 'X', '-' ],  
  [ '-', '-', '-', 'Z', '-', 'X', '-' ],  
  [ '-', 'X', '-', '-', '-', '-', '-' ],  
  [ 'D', 'X', '-', 'V', '-', '-', 'M' ] ]
```

Bonus

Se puede obtener un bono de 5 décimas sobre la nota del laboratorio, si se implementa una interfaz gráfica distinta de la consola para mostrar el juego. Para optar a esta bonificación, la nota del laboratorio debe ser mayor o igual a 3,5.

Política de Integridad Académica

Los alumnos de la Escuela de Ingeniería deben mantener un comportamiento acorde al Código de Honor de la Universidad:

“Como miembro de la comunidad de la Pontificia Universidad Católica de Chile me comprometo a respetar los principios y normativas que la rigen. Asimismo, prometo actuar con rectitud y honestidad en las relaciones con los demás integrantes de la comunidad y en la realización de todo trabajo, particularmente en aquellas actividades vinculadas a la docencia, el aprendizaje y la creación, difusión y transferencia del conocimiento. Además, velaré por la integridad de las personas y cuidaré los bienes de la Universidad.”

En particular, se espera que mantengan altos estándares de honestidad académica. Cualquier acto deshonesto o fraude académico está prohibido; los alumnos que incurran en este tipo de acciones se exponen a un procedimiento sumario. Ejemplos de actos deshonestos son la copia, el uso de material o equipos no permitidos en las evaluaciones, el plagio, o la falsificación de identidad, entre otros. Específicamente, para los cursos del Departamento de Ciencia de la Computación, rige obligatoriamente la siguiente política de integridad académica en relación a copia y plagio: Todo trabajo presentado por un alumno (grupo) para los efectos de la evaluación de un curso debe ser hecho individualmente por el alumno (grupo), sin apoyo en material de terceros. Si un alumno (grupo) copia un trabajo, se le calificará con nota 1.0 en dicha evaluación y dependiendo de la gravedad de sus acciones podrá tener un 1.0 en todo ese ítem de evaluaciones o un 1.1 en el curso. Además, los antecedentes serán enviados a la Dirección de Docencia de la Escuela de Ingeniería para evaluar posteriores sanciones en conjunto con la Universidad, las que pueden incluir un procedimiento sumario. Por “copia” o “plagio” se entiende incluir en el trabajo presentado como propio, partes desarrolladas por otra persona. Está permitido usar material disponible públicamente, por ejemplo, libros o contenidos tomados de Internet, siempre y cuando se incluya la cita correspondiente.