

Trabajo práctico final

Consigna

Integrar a un juego el algoritmo de Goal Oriented Action Planning (GOAP) visto en clase y agregarle las siguientes características:

- Además de hacer el plan, ejecute las acciones (mueva al agente, destruya objetos, recoja items, etc.). Para la ejecución de las mismas (incluyendo pathfinding), implementar una corrutina o una máquina de estados, lo que resulte más cómodo.
- El agente debe moverse a través de un grafo (o grilla) utilizando A* funcional lazy (visto en clase) como algoritmo de pathfinding.
- En vez del modelo de mundo basado en un diccionario de booleanos visto en clase, se desea operar sobre un modelo del mundo de juego (clonable, pero minimalista para evitar overhead excesivo de memoria). Este debe tener al menos un estado de cada uno de los siguientes tipos: **float, int, bool, string**.

Ejemplo: vida (float), monedas (int), vivo (bool), arma (string = “espada”, “lanza”,...).

Evalúe y aplique la estructura de datos que tenga más sentido para su juego.

IMPORTANTE: Los diferentes tipos de datos han de afectar de manera coherente a la heurística, y no se debe calcular la misma tratando a los mismos como si fueran booleanos (usando la heurística utilizada en clase).

-No es necesario que TODAS las variables afecten a la heurística, la influencia de las mismas sobre la heurística (si la tienen) dependerá del juego realizado. (algun juego puede priorizar la vida en int, otros dependeran de los strings, otros de ambos, etc.)

Ejemplo: Si tengo una variable vida (int), la heurística no debería ser la misma si vida = 10 que si vida = 100 .

- **Debe** haber por lo menos **5 acciones suficientemente distintas** en lo que causan sus **efectos**.
- Precondiciones y objetivos **formulados con lambdas**, por ejemplo, para una variable de estado “m” del modelo del mundo de juego,
 - una precondición podría ser “ $m \Rightarrow m.oro > 10$ ” o “ $m \Rightarrow 2*(m.mp + m.xp) < fv$ ”
 - un efecto puede ser “ $m \Rightarrow m.oro += 4$ ” o “ $m \Rightarrow m.hp *= 1.4f$ ”
- Nótese que **todo debe estar implementado de manera funcional y lazy con LINQ y/o generators como se ve en la cursada**.
- Los grupos pueden ser de **hasta** 2 alumnos.

Opcional:

- Se entregará **1 punto extra** por cada uno de los siguientes ítems que sean implementados, siempre y cuando su implementación sea correcta y el trabajo ya esté aprobado. (hasta una nota máxima de 10)
 - Aplique “time-slicing” a GOAP y A*, para ejecutar con una cantidad de tiempo determinada por frame los algoritmos mencionados.
 - Aplique el algoritmo de subdivisión espacial (grilla) visto en clase.

Entrega y defensa

- Se ha de presentar un **Informe escrito sobre el trabajo realizado**. El mismo ha de contener:
 1. Resumen del juego (mecánicas, objetivos).
 2. Las acciones que el goap puede realizar (junto a sus precondiciones, efectos, costos y demás)
 3. Ha de explicar detalladamente cómo se implementaron los cambios pedidos al algoritmo de GOAP, como se implementó el mismo en el juego (como guardaron las acciones, nivel de abstracción del plan con respecto a las acciones individuales, etc), por que se implementó de esa forma y no de otra, y comentar las dificultades que hubieran encontrado en la implementación del algoritmo (si las hubiera) y cómo fueron resueltas.
- Entrega **por e-mail una semana previa** a la fecha de final. Corrobore que el e-mail fue enviado con éxito.
- Además, entrega en un archivo .rar del trabajo en la fecha del final para este quedar en el archivo de la escuela. El mismo ha de:
 - Contener un .rar con el proyecto de Unity
 - Contener un .txt con los datos de los alumnos (nombre y apellido, comisión, turno, etc)
 - Contener una copia del informe escrito.
 - Contener una copia de las consignas (este archivo)
- Defensa oral, opcionalmente grupal, en la fecha de entrega.

Criterio de evaluación

1. Debe cumplir con todas las consignas implementadas correctamente (no perfectamente) para poder aprobar.
2. Debe tener cada punto implementado perfectamente para conseguir un 10.
3. Si se copia, le queda un 1.