

Fundamentos de la Teoría de la Computación - TP Final

Trabajo final de la materia Fundamentos de la Teoría de la Computación del primer semestre del año 2017.

Trabajo realizado por el alumno Sacha Spinelli, N° de legajo: 12282/3.

Enunciado

Diseñar e implementar un Agente Inteligente, similar al visto en clase (la aspiradora). Pueden utilizar cualquier lenguaje de programación. En el repositorio del libro de Russell y Norvin <https://github.com/aimacode> hay algunos ejemplos y pueden re-utilizar código.

Solución

Tomando el ejemplo del libro `Inteligencia artificial`. Un enfoque moderno de Russell y Norvig, realicé la solución detallada a continuación.

Se tiene una matriz de $N \times N$ la cual se encuentra ocupada de manera aleatoria por suciedad en algunos casilleros. Una vez que se crea la matriz, una aspiradora se ubica de manera aleatoria en algún casillero y a partir de allí realiza el siguiente comportamiento en loop:

- *Verifica si su posición actual se encuentra "sucia"*
 - *Caso afirmativo la "limpia"*
 - *Caso negativo no realiza ninguna acción*
- *Chequea las posiciones cercanas para ver si están "sucias" (8 como máximo dependiendo de en qué posición de la matriz se encuentre la aspiradora)*
 - *En caso de que encuentre posiciones cercanas "sucias" avanza hacia alguna de ellas en manera aleatoria*
 - *Si no hay ninguna posición cercana "sucia" avanza sobre alguna cercana "limpia" por la cual no haya avanzado antes (de ser posible)*
- *Chequea si terminó de limpiar toda la matriz*
 - *En caso afirmativo el loop finaliza*
 - *De lo contrario, vuelve al primer paso*

Aspectos de la Solución

La solución fue realizada en JavaScript con apoyo de las tecnologías web, HTML, CSS para brindar una visualización del comportamiento de la aspiradora.

Para hacer funcionar la aspiradora, siga los siguientes pasos en la consola de comandos (bash por ej):

- `git clone https://github.com/saczuac/vacuummer.js.git`
- `cd vacuummer.js`
- `npm install`

- `npm run babel`
- `npm start`
- Vaya al navegador a la dirección `http://localhost:8080`, allí encontrará la aspiradora funcionando.
- Ingrese el tamaño de la matriz que quiere obtener en el campo `Size of Matrix`.
- Haga click en `Run Vacuummer`. (Podrá ver funcionando la aspiradora en tiempo real)

Para poder visualizar el comportamiento de la aspiradora en el navegador se utilizó la sentencia `sleep(500)` en cada iteración de la aspiradora, puede remover esta línea del código fuente si así desea hacerlo y volver a correr los comandos `npm run babel` y `npm start`.

Tiempos de Ejecución

Tamaño	Tiempo (ms)
5	1
10	30
25	969
50	37255

Para la toma de tiempo de ejecución se quitó la línea `sleep(500)` del código fuente.

Conclusión

El tiempo de ejecución crece exponencialmente a medida que se aumenta el tamaño de la matriz, por lo cual el algoritmo no es eficiente.

Se pudo realizar una solución del agente inteligente, la aspiradora es capaz de elegir el mejor camino a la hora de limpiar.

Deploy Heroku

El proyecto se encuentra en producción funcionando en <https://vacuummer.herokuapp.com/>