

Universidad del Valle  
Facultad de Ingeniería  
Escuela de Ingeniería de Sistemas y Computación  
Inteligencia Artificial  
Proyecto I

**SORTING ROBOT:** El Sorting Robot es un robot creado para ordenar recogiendo y reubicando diferentes tipos de objetos según la ubicación que les debe corresponder. El ambiente que se asume se podría modelar con una cuadrícula de N posiciones por cada lado. Se considerarán dos objetos de cualquier peso, el objetivo del robot es llevar los dos objetos a los lugares que le corresponde dentro de la cuadrícula. En cada exploración del ambiente, el agente puede realizar desplazamientos simples como moverse arriba, abajo, izquierda o derecha. El robot podría cargar ambos objetos al mismo tiempo. Aunque este ambiente no considera muros o límites más allá a los que delimita la cuadrícula, sí pueden existir algunas rejillas en las cuales sea más difícil desplazarse, estas rejillas tendrán un valor asociado mayor a cero, el resto de rejillas por defecto tienen un valor de cero.

El costo del movimiento del robot depende del peso que cargue: Si no lleva peso, el costo es 1 más el valor de la rejilla a la cual se desplazó, en caso contrario el costo corresponde a 1 más el peso de los objetos que lleve más el valor de la rejilla a la cual se desplazó.

Tenga en cuenta que cuando se llega a una casilla que tiene un objeto el agente debe obligatoriamente tomarlo, es decir, no puede decidir entre tomar, o no, el objeto. De igual forma, cuando se alcance la casilla con el punto donde debe estar localizado el objeto se deja allí obligatoriamente el objeto correspondiente. El problema termina cuando el robot ubique los objetos en las posiciones que le corresponden.

|                        |   |                        |   |
|------------------------|---|------------------------|---|
| POSICIÓN ROBOT         | 6 | LUGAR PARA EL OBJETO 1 |   |
|                        |   | 10                     | 2 |
| OBJETO 1 ( 4 kilos)    | 8 | OBJETO 2 (5 kilos)     |   |
| LUGAR PARA EL OBJETO 2 | 1 |                        |   |

La información del ambiente inicial se representa con una matriz de  $N \times N$  en la que cada rejilla o celda tiene los siguientes valores:

0 si es una celda vacía no difícil

-1 si en la celda se ubica el robot

-2 si en la celda se ubica el objeto 1

-3 si en la celda se ubica el objeto 2

-4 si la celda es el lugar del objeto 1

-5 si la celda es el lugar del objeto 2

M: donde M es un número mayor a cero, indica el grado de dificultad de la respectiva celda.

El archivo de entrada sería así :

| N         | Peso objeto1 | Peso objeto2 |           |
|-----------|--------------|--------------|-----------|
| $C_{1,1}$ | $C_{1,2}$    | ...          | $C_{1,n}$ |
| $C_{2,1}$ | $C_{2,2}$    | ...          | $C_{2,n}$ |
| $C_{3,1}$ | $C_{1,2}$    | ...          | $C_{3,n}$ |
| .         |              |              |           |
| .         |              |              |           |
| .         |              |              |           |
| $C_{n,1}$ | $C_{n,2}$    | ...          | $C_{n,n}$ |

Se asume que inicialmente la posición del robot, la de los objetos y los lugares que le corresponden son diferentes. Además las anteriores posiciones no pueden ser difíciles.

Usted debe desarrollar una aplicación que permita:

Ingresar los datos de un mundo determinado por medio de un archivo de texto que siga las convenciones dadas anteriormente. Desplegar gráficamente el mundo del agente en su estado inicial, es decir, tal como se lee del archivo.

Seleccionar el tipo de algoritmo de búsqueda a aplicar: "No informada" ó "Informada". Si se selecciona búsqueda "No informada" se puede elegir entre "Amplitud", "Costo uniforme" y "Profundidad evitando ciclos". Si se selecciona búsqueda "Informada" se puede elegir entre "Avara" y "A\*".

Una vez aplicado un algoritmo se debe mostrar en la interfaz gráfica el conjunto de movimientos que realiza el agente. Después de aplicar un algoritmo se debe mostrar un reporte con la siguiente información: cantidad de nodos expandidos,

profundidad del árbol y tiempo de cómputo.

Además, se debe entregar un informe que contenga:

Explicación de la estrategia voraz y un análisis de su optimalidad y completitud.

Explicación de la heurística utilizada en A\*. Como en el caso anterior debe haber un análisis de su optimalidad y completitud., incluyendo el análisis de admisibilidad correspondiente.

Opcional : Puede considerar opcional más objetos en el proyecto, puede incorporar obstáculos e implementar más estrategias para resolver el problema. En todo caso debe recordar que es obligatorio entregar lo solicitado inicialmente y los elementos adicionales opcionales no reemplazan en ningún sentido la obligatoriedad de lo que debe tener la aplicación y el informe.

Fecha de entrega: Octubre 7 de 2013.