



Universidad Autónoma de Nuevo León
Facultad de Ciencias Físico Matemáticas



Inteligencia Artificial
Maestro: Juan Pablo Rosas Baldazo

Producto Integrador de Aprendizaje
“PROGRAMA”

Equipo 6:

Brenda Sarahi Ortiz Solís	1817454
Andrés Alexis Bravo Mesta	1821380
Fernando Elizondo Acevedo	1664590

Descripción del problema:

Descripción:

El problema de resolver es el de la ruta mas corta donde nuestro espacio de estados sería un conjunto de nodos conectados por sus distancias donde pueden moverse de un lado a otro según los recorridos que hicimos. Por lo que el objetivo es llegar desde el nodo inicial al nodo final con la ruta más corta:

Un camionero sale a carretera y empieza cargando gasolina en una estación cercana, pero es un largo camino por recorrer por lo que tendrá que detenerse varias veces en el camino a comprar, cargar mas gasolina o ir al baño por lo que tendrá que decidir en cual de las tantas gasolineras que hay se detendrá y en qué orden para poder llegar con el camino más corto posible.



Características:

El problema habla de un conjunto de gasolineras que vendrían representándose por nodos que seria donde el camión pasaría, y las aristas que tienen un peso representa la distancia en que están. Por esta razón se decidió usar grafos no dirigidos porque en si las gasolineras no te dirigen específicamente a otra gasolinera ni se una sigue otra, sino que simplemente están ahí dispersas y el problema que tiene el camionero es que debe decidir en cuales hacer una parada para tomar lo que necesita.

Se uso Python para realizar el código de este problema donde hicimos uso de la clase grafo y clase nodo para construir los grafos. También usamos 3 algoritmos de búsqueda que nos ayudaron a encontrar la ruta más corta del nodo inicial al nodo final y por último realizamos unas graficas que median el tiempo de ejecución de cada algoritmo,

cada algoritmo se ejecuta varias veces y después se saca el promedio de sus ejecuciones por que con esto puede darnos información mas precisa de su tiempo de ejecución para posteriormente graficarlos. Igualmente se realizo una grafica donde se comparan la cantidad de pasos que tienen que hacer hasta llegar al último nodo y también donde se comparan los costos dando un total de 3 graficas que describen a los algoritmos.

Reglas:

- El agente deberá empezar el grafo en uno de los nodos antes establecidos.
- El agente debe de tener a un objetivo al nodo al cual llegar.
- Los nodos deben de estar unidos por una arista en el grafo.
- Los nodos deberán marcarse con algún símbolo, letra o número para diferenciarlos entre sí.
- El agente siempre estará posicionado sobre un nodo.
- El agente puede moverse solo a los nodos que tengan una conexión con el nodo en el que está.
- El agente también solo puede evaluar los nodos que estén contactándose en el actual en el que se encuentre.
- Las aristas deberán tener un valor numérico de distancia entre cada nodo.
- Los grafos que utilizamos son grafos no dirigidos.

Tamaño de las instancias:

Grafo 1: 20	Grafo 11: 7
Grafo 2: 9	Grafo 12: 8
Grafo 3: 4	Grafo 13: 6
Grafo 4: 8	Grafo 14: 10
Grafo 5: 11	Grafo 15: 5
Grafo 6: 6	Grafo 16: 5

Grafo 7: 9	Grafo 17: 7
Grafo 8: 11	Grafo 18: 6
Grafo 9: 4	Grafo 19: 8
Grafo 10: 5	Grafo 20: 6

Parámetros:

- Cantidad de nodos
- Conexiones entre nodos
- Distancia de cada conexión
- Agente que recorra los nodos

Algoritmos utilizados:

BFS:

Búsqueda en anchura (en inglés BFS - Breadth First Search) es un algoritmo de búsqueda no informada utilizado para recorrer o buscar elementos en un grafo (usado frecuentemente sobre árboles). Intuitivamente, se comienza en la raíz (eligiendo algún nodo como elemento raíz en el caso de un grafo) y se exploran todos los vecinos de este nodo. A continuación para cada uno de los vecinos se exploran sus respectivos vecinos adyacentes, y así hasta que se recorra todo el árbol.

DFS:

Es un algoritmo de búsqueda no informada utilizado para recorrer todos los nodos de un grafo o árbol (teoría de grafos) de manera ordenada, pero no uniforme. Su funcionamiento consiste en ir expandiendo todos y cada uno de los nodos que va localizando, de forma recurrente, en un camino concreto. Cuando ya no quedan más nodos que visitar en dicho camino, regresa (Backtracking), de modo que repite el mismo proceso con cada uno de los hermanos del nodo ya procesado.

A* Star:

Este algoritmo es una mejora desarrollada a los postulados del algoritmo Dijkstra que se encarga de encontrar rutas más cortas dentro de un grafo. En esta modificación se toma como punto central la observación búsquedas informadas dentro del grafo que nos permitan tomar decisiones óptimas sobre los caminos que deben tomarse para recorrer de forma eficiente el grafo