**CARRERA: Ingeniería de Sistemas**

**Asignatura: Inteligencia Artificial I**

**Guía didáctica de la Clase N° 02**

UNIDADES: III y IV

**Identificación: Resolución de problemas por medio de búsquedas**

**Fecha: 19/08/2022**

**Docente:** Ing. M. Sc. Carlos Domingo Almeida Delgado

**Alumno (Nombres y Apellidos):**

**Capacidades**.

* Diseñar y resolver correctamente los árboles de búsqueda.

**Indicadores de aprendizajes esperados (se corresponde con los indicadores a ser evaluados)**

UNIDAD

1. Utilizar las herramientas de la inteligencia artificial en la resolución de problemas: árboles de búsqueda.

**Actividades de desarrollo de la unidad para logro de capacidades y aprendizajes esperados**

1. Análisis de las tareas encomendadas.
2. Ejercicios prácticos de las unidades III y IV.
3. Resolución de problemas.
4. Observación de los problemas a ser evaluados y registro de las actividades.

**INDICADORES DE APRENDIZAJES ESPERADOS**

1. Resolver problemas por medio de búsquedas.
2. Agente solucionador de problemas, problemas y soluciones bien definidos,
3. Formulación de problemas,
4. Búsqueda de soluciones.

**Materiales de consulta básica:**

* RUSSELL, Stuart; NORVIG, Peter. “Inteligencia Artificial Un enfoque moderno”. Madrid: Pearson Prentice Hall, 2004.

**Materiales de consulta complementaria**

* PONCE, Pedro. “Inteligencia Artificial con aplicaciones a la Ingeniería”. México: Alfaomega Grupo Editor, S.A. de C.V., 2010.

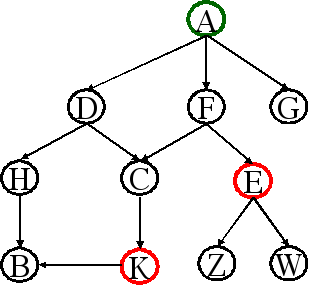
**1· ¿Cómo definiría con tus palabras….?**

* 1. Explique la diferencia entre un árbol de estados, un grafo de estados
  2. Diferencia entre estado y nodo y enumere los elementos que contiene un Nodo, explicar f(n), g(n)
  3. Enumere y explique los pasos genéricos de manipulación de una lista de nodos que permiten crear un árbol de nodos. ¿Se puede generar un grafo de nodos en la búsqueda? ¿por qué?, ¿cuál es el paso que determina esto último?
  4. ¿Qué significa que una estrategia de búsqueda sea ciega?
  5. ¿Qué problemas se originan en la búsqueda si el espacio de estados considerado define un grafo con ciclos o estados con dos padres? ¿Cómo se pueden solucionar esos problemas en la búsqueda? Identifique en que paso y cómo controlaría el problema
  6. Se dice que profundidad iterativa combina las ventajas de dos estrategias de búsqueda. ¿Cuáles son estas estrategias? Justificar adecuadamente

**2· Ejercicios para resolver**

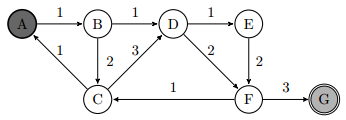
1. Grafo simple

El grafo que se muestra a continuación determina un problema de búsqueda. Cada nodo representa un estado; los arcos modelan la aplicación de los operadores. Si A es el estado inicial y K y E son los estados meta

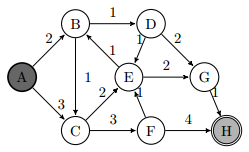


* 1. Siendo el espacio del problema un grafo de estados ¿Donde se repetirán los estados en los nodos generados?
  2. Desarrolla el árbol de búsqueda en amplitud y en profundidad, sin control de estados repetidos en los nodos. Escribe el estado de la lista abierta (frontera) en cada paso del algoritmo, detalla f(n) de cada nodo e identifica el paso del esquema conceptual (*1 sacar, 2 evaluar, 3 generar y 4 poner*)
  3. ¿Cuál de los nodos meta se expande primero?
  4. Repetir *item a* pero evitando se repitan estados.
  5. Dibujar el árbol de nodos resultante de la búsqueda e indicar qué nodos están en la lista abierta y cuales en la lista cerrada.

**3- Dado el siguiente grafo, donde cada arco indica su coste, indica cuál sería la secuencia de nodos explorados que se obtendría mediante el algoritmo de CU,para encontrar el camino entre el nodo A y el nodo H.**

**a **

**sin control de estados repetidos y ciclos y desempatando a ciegas en anchura.**

**b**

**evitar estados repetidos en nodos ya evaluados - desempatando a ciegas en profundidad.**

**Realizar una búsqueda en profundidad limitada interactiva y determinar en ctas interacciones se encuentra la solución.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **interacciones** | **paso 1**  **Sacar**  nodo sacado de lista abierta | **paso 2**  **Evaluar**  Meta? (si/no) | **Guardar**  Nodos EVALUADO | **paso 3**  **Generar**  Posibles Nodos hijos generados  (aún no en lista) | **paso 4**  **Poner**  Lista FRONTERA o Nodos aún NO evaluados |
|  |  |  |  |  |  |

**B. Vamos a ver ejemplos[[1]](#footnote-1) de ejecución de los algoritmos vistos en estos capítulos.**

En la figura 1 podemos ver el grafo que vamos a utilizar en los siguientes ejemplos. Este grafo tiene como nodo inicial el 1 y como nodo final el 8, todos los caminos son dirigidos y tienen el mismo coste.

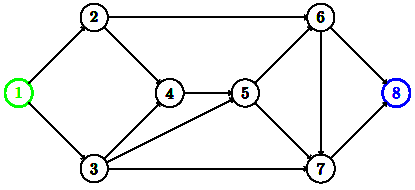


Figura 1 ejemplo de grafo de estaos del problema

Utilizaremos los algoritmos de búsqueda ciega para encontrar un camino entre estos dos nodos.

Supondremos que hacemos un tratamiento de los nodos repetidos durante la ejecución del algoritmo. Hay que tener en cuenta también que para poder recuperar el camino, los nodos deben tener un enlace al padre que los ha generado.

1. Indicar el paso (mapa conceptual, receta) representando nodo de la lista abierta y cerrada en cada interacción. Indicar el **paso a paso** (mapa conceptual, receta) representando el nodo de la lista abierta y lista cerrada en cada interacción. De cada nodo: Indicar función de coste f(n) entre paréntesis

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **interacciones** | **paso 1**  Lista cerrada o Nodos EVALUADO  (sacados de lista abierta) | **paso 2**  Meta?  (si/no) | **paso 3**  Posibles Nodos hijos generados  (aún no en lista) | **paso 4**  Lista ABIERTA o Nodos aún NO evaluados |
|  |  |  |  |  |

1. Dibujar los árboles de búsqueda final,  Indicar el orden de evaluación de los nodos y su función de coste (f(n). Indicar así mismo el camino encontrado.

**Considere un mundo de la aspiradora como problema**

1. ¿Cual de los algoritmos estudiados te parece más apropiado para resolver este problema? ¿Debería el algoritmo comprobar estados repetidos?
2. Aplique el algoritmo elegido para obtener una secuencia de acciones óptima para un mundo de 1x3 cuyo estado inicial tiene suciedad en los 3 cuadrados y el agente está en el del centro.

1. Extraido de: APUNTS D’INTEL.LIGENCIA ARTIFICIAL del CURS 2009/2010 2Q. Departament de Llenguatges i Sistemes Informatics` de la Facultad de Informatica de Barcelo, Universidad Politécnica de Catalunya [↑](#footnote-ref-1)