

1. Estrategias de búsqueda sin información:
  - Búsqueda preferente por amplitud: Se explora todos los nodos a la misma profundidad antes de avanzar a la siguiente. Es útil para encontrar la solución más cercana en términos de nivel de profundidad.
  - Búsqueda de costo uniforme: Se selecciona el camino que tenga el menor costo acumulado para llegar a un estado. Es útil cuando los costos son importantes en el problema.
  - Búsqueda preferente por profundidad: Se explora todo un camino hasta llegar a la profundidad máxima antes de retroceder. Es útil para ahorrar memoria en problemas grandes.
  - Búsqueda limitada por profundidad: Se establece una profundidad máxima para la búsqueda y se detiene cuando se alcanza. Es útil cuando se conoce la profundidad de la solución.
  - Búsqueda por profundización iterativa: Se realiza una búsqueda en profundidad limitada, pero aumentando la profundidad en cada iteración. Es útil para ahorrar memoria y encontrar soluciones de forma progresiva.
  - Búsqueda direccional: Se busca hacia la dirección en la que se cree que está la solución. Es útil cuando se conoce el objetivo del problema y se tiene alguna idea de la ubicación de la solución.
2. Estrategias de búsqueda respaldada con información o heurística:
  - Búsqueda preferente por lo mejor: Se selecciona el nodo con menor costo estimado de llegar a la solución. Es útil cuando se tiene información heurística del problema.
  - Búsqueda limitada por la capacidad de la memoria: Se limita la cantidad de nodos que se almacenan en memoria, descartando aquellos que tienen menor probabilidad de llevar a la solución. Es útil para ahorrar memoria.
  - Búsquedas de mejoramiento iterativo: Se utiliza una heurística para mejorar la solución actual iterativamente. Es útil para problemas en los que se busca la mejor solución posible.
3. MiniMax es un algoritmo de búsqueda utilizado en juegos de dos jugadores con información completa y no cooperativos, como el ajedrez. El algoritmo se encarga de maximizar el beneficio del jugador que está haciendo la jugada, y minimizar el beneficio del otro jugador. Por otro lado, la poda Alpha Beta es una técnica utilizada para reducir el número de nodos evaluados en el árbol de búsqueda, disminuyendo así el tiempo de procesamiento. La técnica aprovecha el hecho de que, si un nodo es peor que otro para un jugador, entonces el jugador no considerará ese nodo y se puede descartar.
4. El proyecto Shakey fue uno de los primeros proyectos de robótica desarrollados en la década de 1960 por el Laboratorio de Inteligencia Artificial de Stanford. El objetivo del proyecto era desarrollar un robot que pudiera moverse y operar de manera autónoma en un entorno desconocido. Shakey tenía la capacidad de recibir instrucciones y procesar información sensorial para tomar decisiones. Se le considera un hito en la historia de la robótica y la inteligencia artificial debido a su complejidad para la época y las tecnologías que se utilizaron en su desarrollo.