

# Universidad Nacional de la Matanza

Departamento:

Ingeniería e Investigaciones Tecnológicas

Cátedra:

## Fundamentos de TIC's

(Tecnologías de la Información y la Comunicación)

JEFE DE CÁTEDRA:

Mg. Artemisa Trigueros

UNIDAD 6

**PARTE B:**

**INTELIGENCIA ARTIFICIAL**

COLABORACIÓN:

**DOCENTES DE LA CÁTEDRA**

CICLO LECTIVO: **2020**

# INTELIGENCIA ARTIFICIAL

## 1. Introducción.

### ¿Qué es la inteligencia?

Es difícil definir y comprender qué es la inteligencia, aún para aquellos que consagran su vida a estudiarla. Si pudiéramos a varias personas que definan la inteligencia, las respuestas serían diferentes, algunos ejemplos:

- ✓ La capacidad de aprender de la experiencia.
- ✓ La capacidad de discernir y evaluar.
- ✓ La capacidad para razonar.
- ✓ La capacidad para percibir relaciones.

### ¿Pueden pensar las máquinas?

Esta pregunta es el punto central de la inteligencia artificial. El objetivo principal de los estudiosos de la computación es desarrollar máquinas que se comuniquen con sus entornos a través de mecanismos sensoriales tradicionalmente humanos y actúen inteligentemente ante situaciones imprevistas sin intervención humana. Esto requiere que la máquina “entienda”, o perciba las entradas recibidas y pueda obtener conclusiones gracias a alguna especie de razonamiento. Tanto la percepción como el razonamiento son actividades del sentido común que, si bien son naturales para la mente humana, son bastante difíciles para las máquinas, por lo cual ésta área está en proceso de desarrollo si consideramos sus metas y expectativas, si bien se han obtenido teorías y técnicas bien fundamentadas.

Examinaremos algunas de ellas, aplicadas en el diseño de máquinas para resolver problemas y que posean capacidades elementales de percepción y razonamiento.

## 2. Definición de Inteligencia Artificial

La siguiente definición:

**“La inteligencia artificial es el estudio de las ideas que permiten a los computadores realizar aquello que hace a las personas parecer inteligentes.”**

*Patrick Henry Winston, Artificial Intelligence*

Esta definición captura la idea general de la inteligencia artificial, pero ¿la inteligencia artificial contempla la capacidad de realizar cálculos a velocidades pasmosas?, ¿recordar cientos de direcciones al mismo tiempo?, si alguien pudiera hacerlo, “parecería inteligente”. Pero estas acciones no son buenos ejemplos de inteligencia artificial, ya que para un computador son triviales.

Otra definición:

**“La inteligencia artificial es el estudio de cómo lograr que los computadores hagan cosas que, por ahora, los seres humanos pueden hacer mejor.”**

*Elaine Rich (Artificial Intelligence)*

Esta definición coloca a la inteligencia artificial como una “frontera móvil”; en los 50, muchos investigadores de inteligencia artificial trabajaron para crear computadores que pudieran jugar ajedrez, hoy en día pueden derrotar con facilidad a cualquier persona, con excepción de los mejores ajedrecistas, y son pocos los investigadores que estudian este juego.

La siguiente definición, resulta ser más concreta y completa:

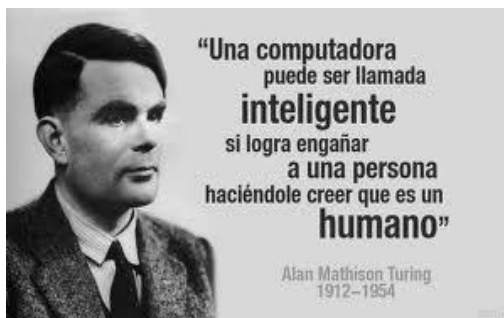
**“La inteligencia artificial es el estudio de los cómputos que hacen posible percibir, razonar y actuar.”**

Patrick Winston (*Artificial Intelligence*)

### 3. Prueba de Turing

En el fondo, la inteligencia es una característica interior que solo se puede detectar indirectamente desde el exterior, por ejemplo, a través de un diálogo estímulo/reacción, que permita diferenciar su existencia real de la simple apariencia de inteligencia.

En 1950, Alan Turing, teniendo en cuenta la característica engañosa de la inteligencia solo en apariencia, propuso una prueba (Prueba de Turing) para detectar inteligencia dentro de una máquina.



(Alan Mathison Turing; Londres, 1912 - Wilmslow, Reino Unido, 1954). Matemático británico. En 1937 publicó un célebre artículo en el que definió una máquina calculadora de capacidad infinita (máquina de Turing) que operaba basándose en una serie de instrucciones lógicas, sentando así las bases del concepto moderno de algoritmo. La máquina de Turing era tanto un ejemplo de su teoría de computación como una prueba de que un cierto tipo de máquina computadora podía ser construida. Durante la Segunda Guerra Mundial, diseñó los procesos

y máquinas que descifraron el código secreto alemán. Definió además un método teórico para decidir si una máquina era capaz de pensar como un hombre (Prueba de Turing) y realizó contribuciones a otras ramas de la matemática aplicada, como la aplicación de métodos analíticos y mecánicos al problema biológico de la morfogénesis.



Una computadora (B) es **INTELIGENTE** si un **evaluador humano** no puede distinguir si el que responde a sus preguntas es la computadora (B) u otra persona (A).

#### 4. Enfoques de la Inteligencia Artificial

Supongamos que un matemático y un psicólogo emprenden de manera independiente, proyectos para crear un programa que juegue al póquer. Lo más probable es que el matemático diseñe un programa basado en los principios de la probabilidad y la estadística: el resultado sería un programa que jugara sin arriesgarse y no mostraría emociones. El psicólogo, en cambio, probablemente crearía un programa basado en las teorías del raciocinio y el comportamiento humano. El programa del psicólogo podría "involucrarse emocionalmente" en el juego.

Reconsiderando, proponemos la hipótesis de que la preocupación principal del matemático al crear el programa sería su rendimiento final. Se dice que un enfoque así está orientado al **rendimiento**. En contraste, el psicólogo estaría más interesado en entender los procesos de la inteligencia natural; así, abordaría el proyecto como una oportunidad de probar teorías construyendo modelos de computador basados en esas teorías. Desde este punto de vista, el desarrollo del programa "inteligente" es en realidad un efecto secundario de otro esfuerzo: avanzar en la comprensión del raciocinio y el comportamiento humanos. Se dice que este enfoque está orientado a la **simulación**. Ambos enfoques son válidos y contribuyen significativamente al campo de la inteligencia artificial.

#### 5. Razonamiento

Gran parte del tema de inteligencia artificial conlleva un aura de misterio. Con el fin de poner luz al mismo consideremos la demostración de inteligencia a través de un sencillo problema, que no posee un método preestablecido de resolución, donde el programa que se le proporciona a la computadora deberá posibilitar que la misma:

- Tome decisiones.
- Saque conclusiones.

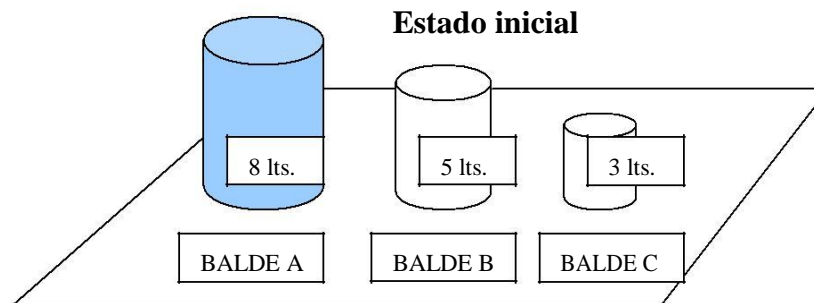
Por lo tanto, sea capaz de realizar actividades elementales de razonamiento.

El desarrollo de capacidades de razonamiento dentro de una máquina sigue siendo tema de investigación actual, por lo cual existen todavía controversias a cerca de la técnica correcta a seguir. Aquí abordaremos el tema en el contexto de los **Sistemas de Producción**.

<b>SISTEMA DE PRODUCCIÓN</b> (Componentes)	<b>COLECCIÓN DE ESTADOS</b>	<b><u>Estado</u></b> : situación que podría presentarse en el entorno de la aplicación. Se parte de un <i>estado inicial</i> y se debe llegar al <i>estado objetivo</i> .
	<b>COLECCIÓN DE PRODUCCIONES O REGLAS</b>	<b><u>Producción</u></b> : operación que puede efectuarse en el entorno de la aplicación para pasar de un estado al otro. Puede haber <i>condiciones</i> que deben estar presentes en el entorno antes de que se pueda aplicar una producción.
	<b>SISTEMA DE CONTROL</b>	Es la lógica que resuelve el problema de pasar del estado inicial al objetivo. En cada paso del proceso, el sistema de control es el que decide cuál de las producciones cuyas precondiciones se satisfacen debe aplicarse.  Para crear una máquina inteligente, el sistema de control se debe implantar como un programa almacenado, el cual examina el estado actual del sistema, identifica una secuencia de producciones que conduce al estado objetivo, y ejecuta dicha secuencia.

### 5.1. Sistema de producción. Problema de los baldes.

Un comerciante tiene tres (3) baldes. El primero de ocho (8) litros de capacidad lleno de líquido, el segundo de cinco (5) litros, vacío y el tercero de tres (3) litros, también vacío. Los recipientes no están graduados. No se puede agregar ni descartar líquido (en todo momento DEBE haber ocho (8) litros entre los 3 recipientes).



Objetivo:

- Medir cuatro (4) litros en cualquiera de los baldes capaces de contenerlos.  
Los otros cuatro (4) litros pueden quedar en cualquiera de los otros baldes.
- Identifiquemos para este caso los tres componentes principales de un sistema de producción:

#### 1) Colección de estados

<b>Estado inicial</b> (Estado del cuál partimos)	A = 8      B = 0      C = 0
<b>Estado objetivo</b> (Estado que queremos alcanzar)	A = 4      B = ?      C = ? A = ?      B = 4      C = ?
<b>Estados válidos</b>	<ul style="list-style-type: none"> <li>• No volcar agua, en cada paso la suma del líquido de los baldes deber ser siempre igual a 8.</li> <li>• El balde <b>A</b> tiene una capacidad máxima de <b>8 litros</b> (nunca puede contener más de 5 lts).</li> <li>• El balde <b>B</b> tiene una capacidad máxima de <b>5 litros</b> (nunca puede contener más de 5 lts).</li> <li>• El balde <b>C</b> tiene una capacidad máxima de <b>3 litros</b> (nunca puede contener más de 3 lts).</li> <li>• Ningún recipiente puede quedar con menos de 0 lts (situación absurda para un humano, físicamente imposible, pero posible para la aritmética del procesador) por lo cual deberemos informárselo en las producciones como estado no válido.</li> </ul> <p>Es imprescindible informarle a la computadora cuales son los estados válidos y cuáles no, de forma que la misma pueda, a través de algún algoritmo, verificar si un estado es válido o no y si es válido buscar la producción más eficaz para llegar desde el estado inicial al objetivo.</p>

## 2) Colección de producciones o reglas

Operaciones para pasar de un estado al otro:

<i>Producción</i>	<i>Condición previa SI</i>	<i>Comentario</i>	<i>Ejecución ENTONCES</i>	<i>Comentario</i>
1	$0 < A \leq 5 - B$	¿Hay lugar en B para volcar todo A?	$B = B + A$ $A = 0$	<b>Vaciar A en B.</b> B ahora va a tener su contenido + el de A. A queda vacío.
2	$A > 5 - B > 0$	¿Tiene A agua suficiente para completar a B?	$A = A - 5 + B$ $B = 5$	<b>Llenar B con A.</b> A volcó agua en B hasta completarlo. B quedó lleno.
3	$0 < A \leq 3 - C$	¿Hay lugar en C para volcar todo A?	$C = C + A$ $A = 0$	<b>Vaciar A en C.</b> C ahora va a tener su contenido + el de A. A queda vacío.
4	$A > 3 - C > 0$	¿Tiene A agua suficiente para completar C?	$A = A - 3 + C$ $C = 3$	<b>Llenar C con A.</b> A volcó agua en C hasta completarlo. C quedó lleno.
5	$0 < B \leq 8 - A$	¿Hay lugar en A para volcar todo B?	$A = A + B$ $B = 0$	<b>Vaciar B en A.</b> A ahora va a tener su contenido + el de B. B queda vacío.
6	$B > 8 - A > 0$	¿Tiene B agua suficiente para completar A?	$B = B - 8 + A$ $A = 8$	<b>Llenar A con B.</b> B volcó agua en A hasta completarlos. A quedó lleno.
7	$0 < B \leq 3 - C$	¿Hay lugar en C para volcar todo B?	$C = C + B$ $B = 0$	<b>Vaciar B en C.</b> C ahora va a tener su contenido + el de B. B queda vacío.
8	$B > 3 - C > 0$	¿Tiene B agua suficiente para completar C?	$B = B - 3 + C$ $C = 3$	<b>Llenar C con B.</b> B volcó agua en C hasta completarlo. C quedó lleno.
9	$0 < C \leq 8 - A$	¿Hay lugar en A para volcar todo C?	$A = A + C$ $C = 0$	<b>Vaciar C en A.</b> A ahora va a tener su contenido + el de C. C queda vacío.
10	$C > 8 - A > 0$	¿Tiene C agua suficiente para completar A?	$C = C - 8 + A$ $A = 8$	<b>Llenar A con C.</b> C volcó agua en A hasta completarlo. A quedó lleno.
11	$0 < C \leq 5 - B$	¿Hay lugar en B para volcar todo C?	$B = B + C$ $C = 0$	<b>Vaciar C en B.</b> B ahora va a tener su contenido + el de C. C queda vacío.
12	$C > 5 - B > 0$	¿Tiene C agua suficiente para completar B?	$C = C - 5 + B$ $B = 5$	<b>Llenar B con C.</b> C volcó agua en B hasta completarlo. B quedó lleno.

### 3) Sistema de control

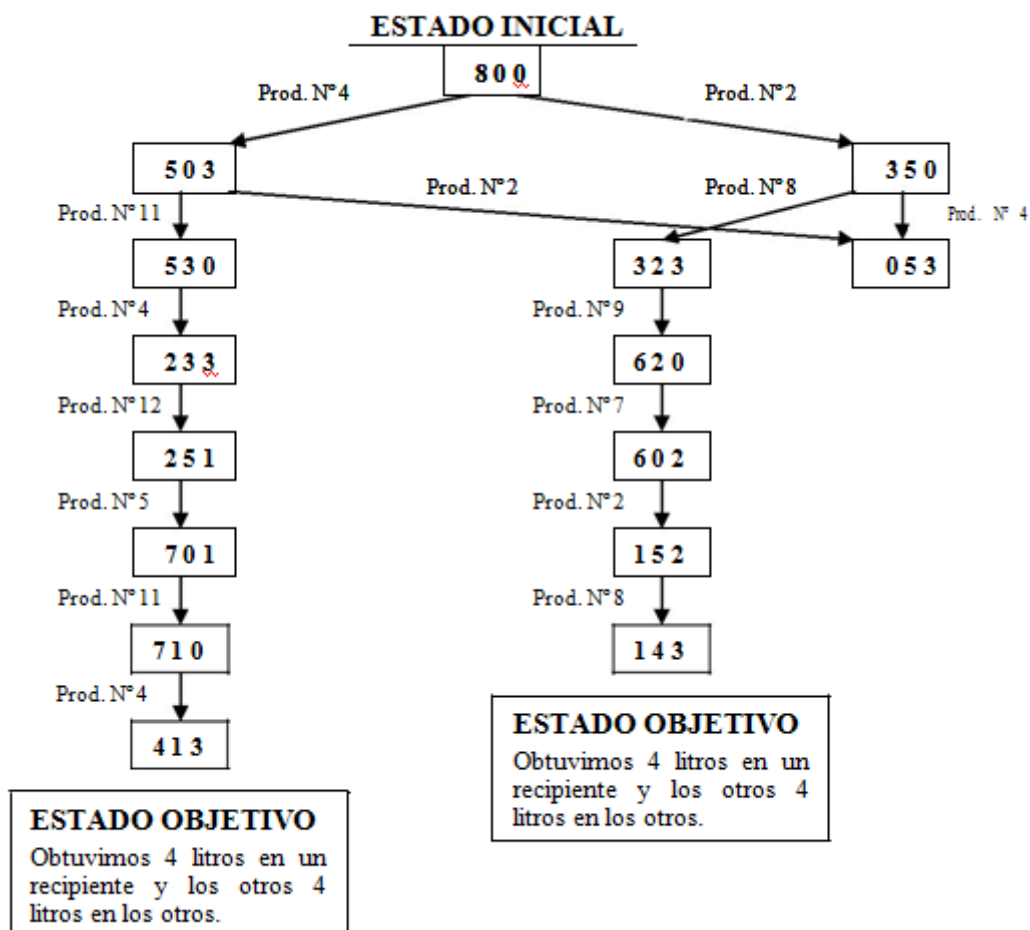
El estado  $A = 8$   $B = 0$   $C = 0$ , será representado como

**8 0 0**

Utilizaremos el concepto de **GRAFO DE ESTADOS**, que es una forma útil de representar, o por lo menos de conceptualizar, todos los estados, producciones y condiciones previas de un sistema de producción. Usamos el término *grafo* en su sentido matemático: una colección de posiciones llamadas *nodos* conectadas mediante flechas llamadas *arcos*.

**Grafo de estados:** Colección de nodos que representan los estados del sistema conectados por arcos que representan las producciones que causan el movimiento de un estado a otro.

Por lo tanto 2 nodos pueden estar unidos por un arco (en nuestro caso usaremos flechas) en el grafo de estados, sí y solo si existe una producción que puede producir el pasaje de un estado al otro.

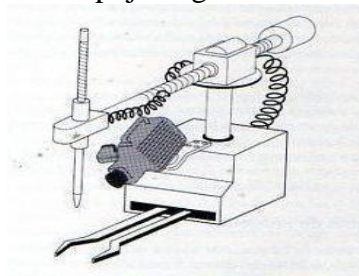


El sistema de control realiza una búsqueda del estado actual de los recipientes en las condiciones previas de cada producción y cuando encuentra una que coincide entonces ejecuta la producción, quedando el estado siguiente como indica esa producción.

Cuando queremos que una computadora, tome decisiones para resolver un problema, debemos “contarle” que significa estado válido y como hace una persona para resolver dicho problema. Mediante instrucciones perfectamente determinadas se escribe un programa que, al ser ejecutado por la computadora, le permite llegar a la solución, dotándola de lo que llamamos “Inteligencia Artificial”.

## 6. Una máquina “inteligente”

Con el fin de aclarar varios puntos en cuestión acerca de la “inteligencia” en las máquinas, consideraremos el diseño de una máquina que tiene propiedades elementales de inteligencia. La misma adopta la forma de una caja de metal equipada con pinzas, una cámara de video y un dedo forrado con caucho para que no resbale al empujar algo.



Está colocada junto a una mesa donde se encuentra un rompecabezas de 8, el cual consiste en un marco cuadrado en donde están montadas ocho fichas, cuadradas y etiquetadas con los números 1 a 8. En el marco caben nueve de estas fichas, por lo tanto, hay un hueco hacia el cual puede empujarse cualquiera de las fichas adyacentes a él. La disposición de las fichas es la siguiente:

1	2	3
4	5	6
7	8	

Figura 1

Luego, tomamos el rompecabezas y cambiamos disposición de las fichas empujándolas repetidamente en forma aleatoria hacia el hueco que va quedando, quedando las fichas desordenadas con respecto a su ubicación inicial.

Encendemos la máquina, las pinzas comienzan a abrirse y cerrarse, como pidiendo el rompecabezas. Le entregamos el rompecabezas y las pinzas se cierran sobre él. En unos instantes, el dedo descende y empuja las fichas dentro del marco (de una manera ordenada), hasta que quedan otra vez en el orden original. Luego la máquina coloca el rompecabezas sobre la mesa y se apaga.

A continuación, estudiaremos la capacidad de razonamiento de nuestra máquina. La acción de abrir y cerrar las pinzas como la de detectar la presencia del rompecabezas, no pueden ser consideradas como acciones inteligentes, ya que el simple mecanismo automático de la puerta de un ascensor es capaz de detectar la presencia de un obstáculo.



## 6.1. Análisis de imágenes

La primera conducta inteligente que tiene nuestra máquina es la de extraer información a través de la cámara de video. La máquina no se limita a producir y almacenar la imagen del rompecabezas, debe comprender la imagen para poder extraer la posición de los dígitos del rompecabezas, nuestra máquina debe tener la capacidad de *percibir*.

Existen varias técnicas de reconocimiento de imágenes:

- **Comparación de patrones:** Previamente codificar la imagen del rompecabezas en términos de unos y ceros en la memoria del computador (cada bit representa el nivel de brillantes de una parte específica de la imagen (píxel)) y dando por sentado que la imagen tiene un tamaño fijo, comparar las diferentes secciones de la imagen con los modelos pregrabados (patrones de bits producidos por los dígitos individuales utilizados) al detectarse coincidencias se determina la situación del tablero. Esta técnica utilizada en los lectores ópticos de caracteres (OCR) requiere uniformidad en el estilo, tamaño y orientación de los símbolos que se leen, ya que el patrón de bits producidos por un carácter físicamente grande no coincide con el patrón de una versión más pequeña. Los problemas se incrementan si el material es manuscrito.
- **Extracción y evaluación de rasgos:** Comparar las características geométricas más que la apariencia exacta de los símbolos. Así el dígito 1 se podría caracterizar como una sola línea vertical, el 2 una línea curva abierta unida a una recta horizontal y así sucesivamente. El proceso consiste en extracción de rasgos de la imagen procesada y posterior evaluación de rasgos (compararlos con los de los símbolos conocidos). No es infalible, errores menores en la imagen, pueden producir rasgos geométricos totalmente distintos, por ejemplo, entre un 3 y un 8.

En el problema del rompecabezas de 8 no nos enfrentamos a dificultades mayores como son trabajar con imágenes tridimensionales, comprender imágenes superpuestas, etc. Como vemos los problemas asociados al reconocimiento de imágenes son muchos, pero máquinas pensadas con otra arquitectura (redes neuronales) ayudan a superar algunos obstáculos, más adelante veremos algunos ejemplos.

Una vez descifradas las posiciones de las fichas a partir de la imagen visual, nuestra máquina debe resolver el rompecabezas. Se nos podría ocurrir pre-programar la máquina con soluciones a todas las posibles disposiciones de las fichas, de manera que nuestra máquina solo debería seleccionar y ejecutar el programa apropiado, esto no nos resultaría atractivo ya que aún en un este sencillo rompecabezas puede haber un total de 181.440 configuraciones distintas, pasando a ser poco factibles por limitaciones de tiempo y espacio de almacenamiento en memoria suministrar una solución explícita para cada una.

Por todo esto deberemos programar nuestra máquina para que pueda resolver el problema ella misma, siendo capaz de tomar decisiones, y sacar conclusiones o sea realizar operaciones elementales de razonamiento.

Analizaremos el problema en el contexto de los sistemas de producción, donde el estado inicial es la configuración del rompecabezas cuando se lo entregamos a la máquina, el estado objetivo es la configuración del acertijo resuelto (figura 1).

Las producciones son los movimientos de las fichas, cada movimiento tiene como condición previa que el hueco debe estar junto a la ficha en cuestión. El sistema de control deberá decidir qué ficha mover de las que se encuentran junto al hueco, o sea qué producción aplicar.

## 6.2 Árboles de búsqueda

Al estudiar los sistemas de control centraremos nuestra atención en el problema de recorrer grafos, ya que hallar la secuencia apropiada de producciones en un sistema de producción se puede formular en términos de hallar un camino a través de un grafo de estados (o en realidad dentro de lo que llamaremos *árbol de búsqueda*) que lleve del nodo inicial (nodo raíz) al nodo objetivo.

En términos del problema de recorrer grafos, presentamos en primer término una técnica más bien de fuerza bruta, luego veremos cómo usar la intuición para obtener un sistema más inteligente.

Una técnica común para realizar la búsqueda consiste en recorrer cada uno de los arcos que salen del nodo inicial y registrar el estado al que se llegó, luego recorrer los arcos que salen de esos nuevos estados y una vez más tomar nota de lo obtenido, y así sucesivamente, hasta que uno de los nuevos estados es el objetivo, o sea se encontró una solución. El sistema de control solo debe aplicar las producciones del camino descubierto desde el estado inicial al objetivo.

Se construye así un árbol de búsqueda, partiendo del nodo raíz, donde los hijos de cada nodo son los estados a los que se puede llegar aplicando una producción. Cada arco entre 2 nodos de un árbol de búsqueda representa la aplicación de una sola producción.

En particular, si el rompecabezas de 8 tuviera inicialmente sus fichas posicionadas de la siguiente forma:

1	3	5
4	2	
7	8	6

El árbol de búsqueda resultante sería:

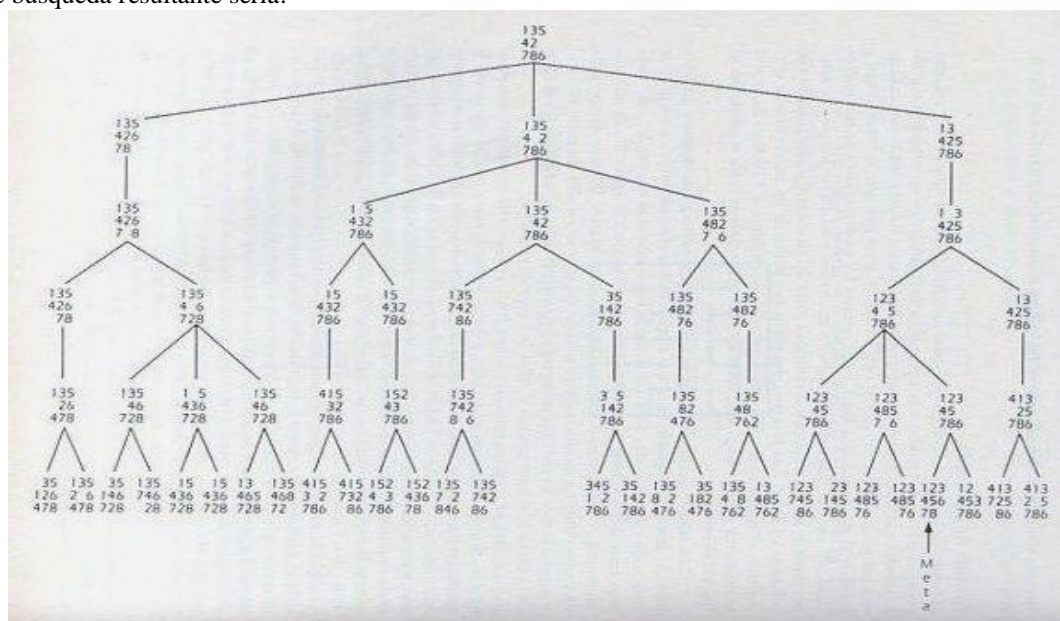


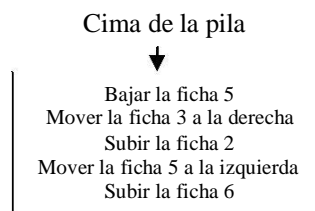
Figura 2

La rama más a la izquierda del árbol representa el intento de buscar una solución moviendo primero la ficha 6 hacia arriba, la central moviendo la ficha 2 a la derecha y la del extremo derecho si comenzamos moviendo la ficha 5 hacia abajo.

Vemos que, si comenzamos por subir la ficha 6, la única producción que se permitiría después sería mover la 8 a la derecha, aunque en ese punto también podríamos mover la ficha 6 hacia abajo, lo cual sería inútil pues nos devolvería a un estado ya representado anteriormente.

El estado objetivo aparece en el último nivel del árbol de búsqueda, el sistema de control no necesita construir más niveles del árbol y empieza a subir por el árbol de búsqueda desde el nodo objetivo hasta la raíz (estos árboles se construyen con un sistema de punteros que apuntan hacia arriba en vez de hacia abajo, o con dos series de punteros que permiten desplazarse en ambas direcciones dentro del árbol), guardando en una pila las producciones representadas por cada arco que se encuentra en el camino.

En nuestro ejemplo se obtendría la pila de producciones siguiente:



Esta secuencia de instrucciones servirá para resolver el problema en el mundo externo, o sea la que en realidad va a ejecutar nuestra máquina inteligente.

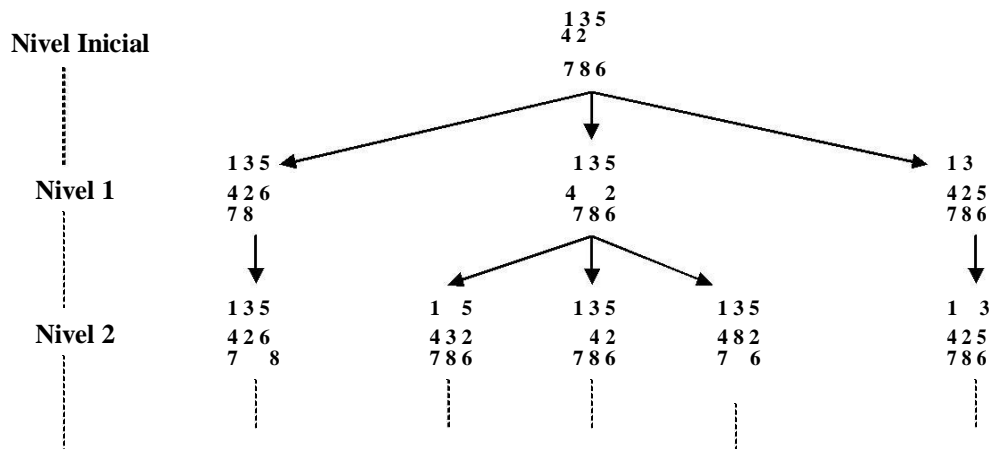
### 6.2.1 Eficiencia en la construcción de árboles de búsqueda

El ejemplo que hemos elegido produjo un árbol de búsqueda manejable, pero cualquier árbol de búsqueda generado en un intento por solucionar un problema no tan simple como el nuestro, crecería con mayor rapidez que el de nuestro ejemplo, debido al gran número de opciones disponibles en cada etapa y a la mayor profundidad que tendría el mismo antes de hallar la meta.

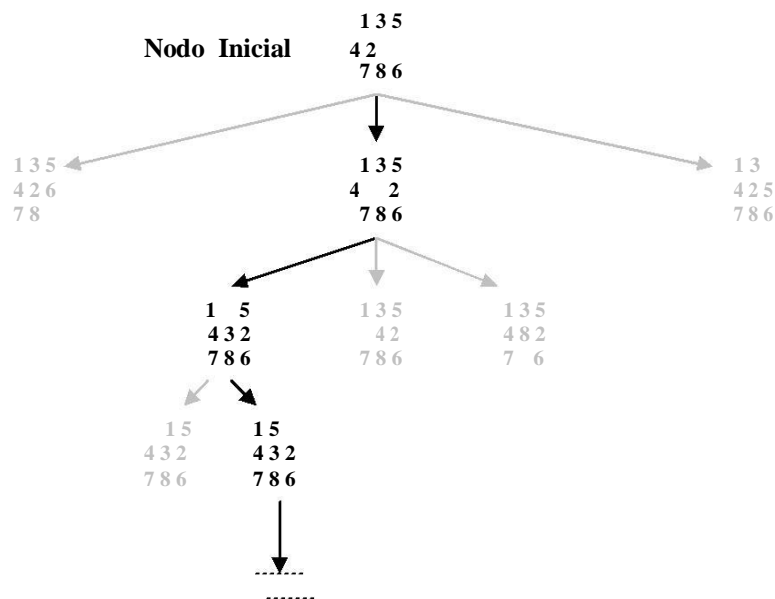
Si pensamos en el juego de ajedrez que tiene 20 primeras jugadas posibles, el nodo raíz tendría 20 hijos en vez de los tres que tiene en el rompecabezas de 8, y un juego de ajedrez fácilmente puede consistir en 30 o 35 pares de jugadas en vez de las 5 sencillas jugadas de nuestro ejemplo. Por todo esto, el desarrollo completo de un árbol de búsqueda resulta poco práctico, como la representación del grafo de estados completo, en término de tiempo y de espacio en memoria. Se necesitan métodos más económicos para manejar el árbol de búsqueda.

➤ Una forma sería **cambiar el orden de construcción del árbol**.

En el árbol de la figura 2, la construcción del mismo se realizó *primero en amplitud* (nivel por nivel), se fueron creando sucesivas capas horizontales:



En lugar de construirlo primero en amplitud, podemos seguir las ramas más prometedoras (cuál es la más prometedora es tema de la sección siguiente) en profundidad y sólo considerar las otras en caso de no llegar al objetivo. Esta construcción del árbol de búsqueda es *primero en profundidad*, en ella se crean caminos verticales en lugar de capas horizontales:



➤ Otra forma de reducir el tamaño de un árbol de búsqueda es **evitar redundancias**.

En el caso del rompecabezas de 8 no conectaremos un nuevo nodo al árbol si ese nodo ya está representado anteriormente. En el caso de los recipientes, al llegar a un estado por el cual ya se pasó, se trunca, se abandona esa rama, no se la sigue. Esto resulta demasiado simplista como regla general, en otras aplicaciones puede ocurrir que la nueva ocurrencia del nodo resulte una solución más eficiente que la anterior, y deba añadirse al árbol eliminando la anterior.

Muchos sistemas de control utilizan métodos más complejos para eliminar redundancias en el árbol de búsqueda, generalmente asocian un costo a los diversos caminos representados en el árbol y siguen los que tienen menor costo.

Podemos adoptar este método en el caso del rompecabezas de 8, considerando que el costo de cualquier camino es el número de jugadas que abarca. Si ocurren repeticiones nos quedaremos con el camino menos costoso, o sea el más corto. Si construimos el árbol nivel por nivel, en el caso de repeticiones el nodo retenido será el más viejo, el que aparece más alto en el árbol.

### 6.3 Empleo de la heurística

Tratando de controlar el tamaño del árbol de búsqueda, veremos que es posible incorporar a nuestro sistema el equivalente de la intuición.

Si nos encontrásemos frente al rompecabezas de 8, lo más probable es que no siguiéramos varias opciones al mismo tiempo, sino que elijamos la opción que nos parece más prometedora y la sigamos, dejándonos llevar por nuestra intuición, aun cuando nos puede conducir al fracaso.

Modificaremos la forma de construir el árbol de búsqueda, incorporando la intuición para que nuestro sistema ahorre tiempo evitando el desarrollo de ramas improductivas en el árbol de búsqueda.

Nosotros, como personas, elegiríamos la opción del estado que se encuentra más cerca del objetivo, en un entorno de programación, debemos desarrollar una medida cuantitativa que permita al programa determinar cuál de varios estados se encuentra más cerca del objetivo. Una medida de estas características se llama heurística.

**Información heurística:** información empírica no comprobada, que las personas obtienen utilizando su intuición.

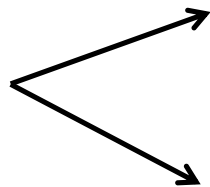
Una forma es asociar a cada estado un valor igual al número de fichas que no están en su posición deseada, y considerar el estado con el menor valor como el más cercano a la meta. Aquí no se tiene en cuenta qué tan lejos de su destino están las fichas, por lo tanto, podríamos adoptar una medida que tome en cuenta esta distancia.

El proceso consiste para cada estado, sumar el número mínimo de movimientos que debe hacer cada ficha para ubicarse en su posición objetivo (distancia de la ficha), sin tener en cuenta la existencia del resto de las fichas. Por ejemplo, una ficha inmediatamente adyacente a su destino final se asocia a una distancia de 1, y aquella cuya esquina toca el cuadrado de su destino final tiene una distancia de 2.

A la suma de las distancias de cada ficha la llamaremos **Costo Proyectado**. Por ejemplo, el costo proyectado asociado del siguiente estado es:

2	3	4	FICHA	MOVIMIENTOS
5	1		1	2
8	7	6	2	1
			3	1
			4	3
			5	1
			6	1
			7	1
			8	1
<b>COSTO PROYECTADO</b>				<b>11</b>

CARACTERÍSTICAS  
DEL COSTO  
PROYECTADO



Nos es útil para tomar una decisión, ya que nos proporciona estimativamente la cantidad de movimientos a efectuar para obtener la solución si llegamos a ese estado.

Se puede calcular fácilmente, no así el número real de jugadas necesarias para llegar a la meta desde un estado dado.

Ahora que tenemos una heurística la incorporaremos al proceso de toma de decisiones para el rompecabezas de 8. Modificaremos nuestro procedimiento de búsqueda de la Figura 2 considerando el costo proyectado de cada nodo hoja del árbol y continuando la búsqueda en el nodo hoja que posea el menor costo.

El algoritmo para desarrollar un árbol de búsqueda, para un sistema de control que emplee esta heurística, es:

- o Establecer el nodo inicial del grafo de estados como raíz del árbol de búsqueda y registrar su costo proyectado.

- o **MIENTRAS** (no se haya llegado al nodo objetivo), **hacer:**

[Seleccionar el nodo hoja más a la izquierda que tenga el costo proyectado más pequeño de todos los nodos hojas, y conectar como hijos al nodo seleccionado aquellos nodos a los que se puede llegar con una sola producción desde el nodo seleccionado. Registrar el costo proyectado de cada uno de estos nuevos nodos junto al nodo en el árbol de búsqueda.]

- o Recorrer hacia arriba el árbol de búsqueda desde el nodo meta hasta la raíz, metiendo en una pila la producción asociada a cada arco recorrido.

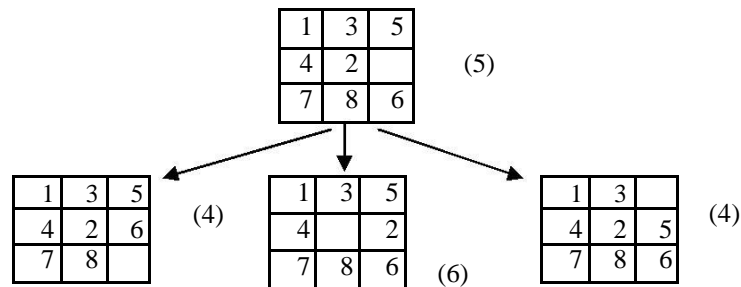
- o Resolver el problema original ejecutando las producciones conforme se desempilan.

Recorramos paso a paso este algoritmo aplicado al rompecabezas de 8, partiendo del estado inicial, que será tomada como nodo raíz. Calculamos su costo proyectado, 5.

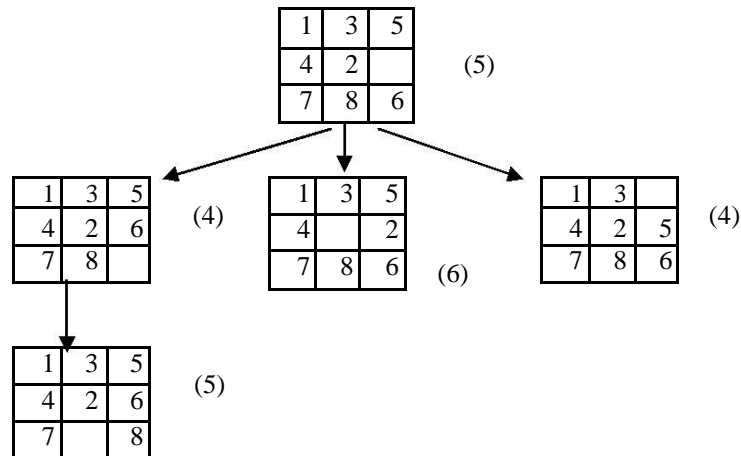
1	3	5
4	2	
7	8	6

(5)

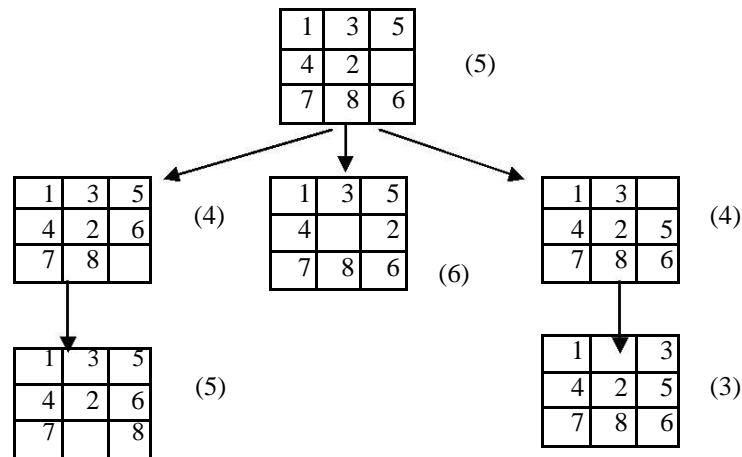
A continuación, la primera ejecución del cuerpo de la estructura mientras produce el agregado de tres nodos hojas a los cuales se les ha calculado su costo proyectado:



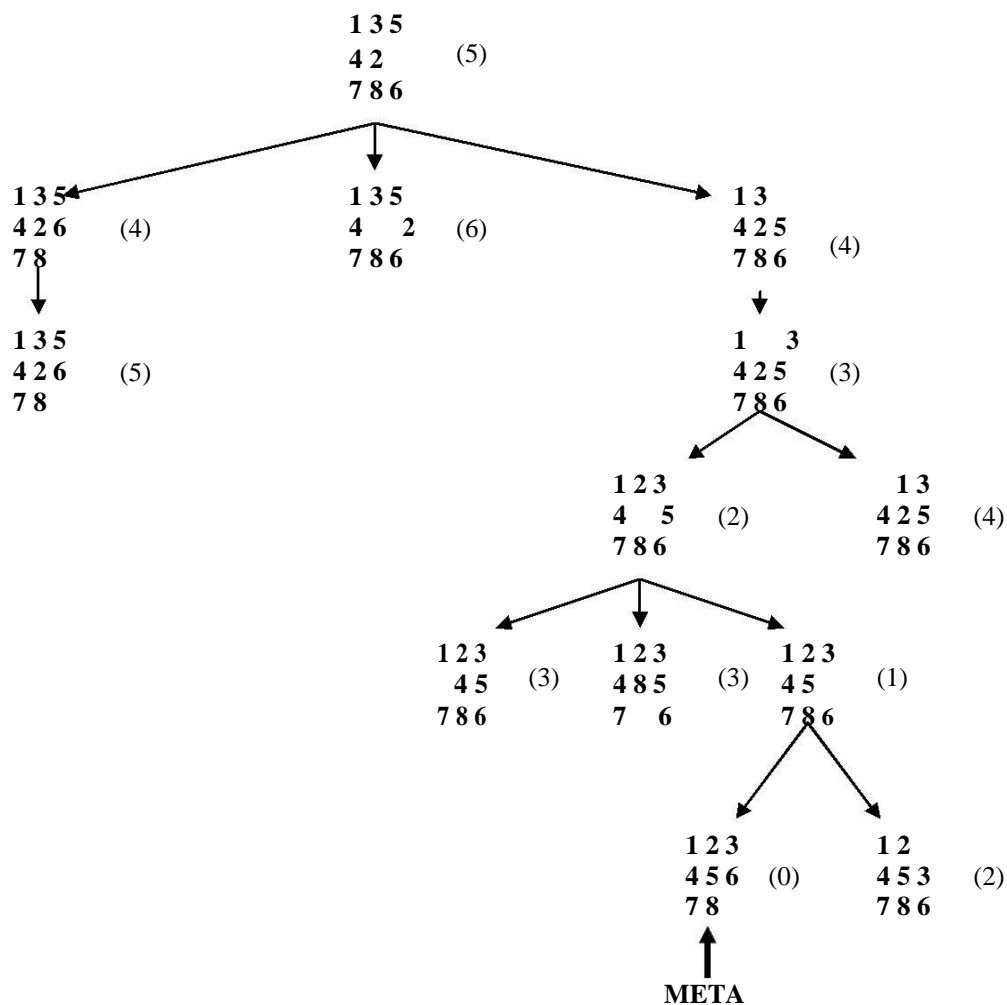
No hemos llegado al nodo objetivo, así que ejecutamos otra vez el cuerpo de la estructura mientras, extendiendo esta vez nuestra búsqueda desde el nodo del extremo izquierdo (“el nodo hoja más a la izquierda que tenga el costo proyectado más pequeño”), luego de lo cual el árbol de búsqueda se muestra de la siguiente forma:



El costo proyectado del nodo hoja del extremo izquierdo es ahora 5, lo que indica que ésta tal vez no sea una buena opción, el algoritmo descubre esto y en la siguiente ejecución del ciclo ordena expandir el árbol a partir del nodo del extremo derecho, que ahora es “el nodo hoja más a la izquierda que tenga el menor costo proyectado”:



Ahora el algoritmo parece ir por buen camino, como el costo proyectado de este último nodo es apenas 3, se ordena seguir por esta rama, por la cual se llega a la meta, y el árbol de búsqueda formado por nuestro sistema de heurística queda de la siguiente forma:





## 7. Aplicaciones de la Inteligencia Artificial

Consideraremos algunas de las áreas en las cuales la inteligencia artificial tiene aplicación.

### Procesamiento de lenguajes naturales

El procesamiento de lenguajes naturales (PLN) se ocupa de la formulación e investigación de mecanismos eficaces computacionalmente (que se puedan realizar por medio de programas) para la comunicación entre personas o entre personas y máquinas por medio de lenguajes naturales.

Una de las aplicaciones es en el terreno de la Traducción Automática.

Los sistemas tradicionales están diseñados de modo de buscar el enunciado original (o una parte del mismo) en una tabla en la cual está almacenado junto con su equivalente traducido, en estos casos la máquina no tiene que entender enunciados sólo reconocer su sintaxis. Pero la traducción de lenguajes naturales (idiomas) como el inglés, español, etc. requiere una “comprensión” del enunciado para ser correcta. Por ejemplo, para traducir los enunciados:

María rompió con su novio.

Y

María rompió esos platos.

Se requiere que el traductor sea capaz de comprender los mismos. En esta tarea el principal inconveniente es que las personas pocas veces se ajustan a reglas estrictas cuando hablan y generalmente no dicen lo que quieren decir. El proceso de entender un enunciado en un lenguaje natural requiere varios niveles de análisis:

Análisis sintáctico	Identifica el papel gramatical de cada palabra. <u>Ejemplo:</u> María lavó la camisa de Pedro. Se identifica a María como sujeto del enunciado.
Análisis semántico	Busca identificar la acción descripta, el agente de esa acción y el objeto de la acción. <u>Ejemplo:</u> reconocería que “María lavó la camisa de Pedro” y “La camisa de Pedro fue lavada por María” dicen lo mismo.
Análisis contextual	Se considera el contexto del enunciado en el proceso de comprensión. <u>Ejemplo:</u> el enunciado “El tiempo cambió de repente”, solo podemos saber su significado si conocemos su contexto.

### Reconocimiento automático de la voz

Los digitalizadores de audio son dispositivos de entrada que pueden capturar palabras habladas, música y otros sonidos para poder almacenarlos como datos digitales. Pero deben ser procesados por software complejo antes que el computador pueda interpretarlos como palabras.

Los sistemas de reconocimiento automático de voz usan técnicas de reconocimiento de patrones similares a las que se emplean en los sistemas de visión y de reconocimiento óptico de caracteres

(OCR):

- Segmentación de los patrones de sonido de entrada para formar palabras y fonemas individuales.
- Reglas expertas para interpretar los sonidos.
- “Expertos” de contexto para manejar sonidos ambiguos.
- Aprendizaje de un entrenador humano.

Existen grandes diferencias entre las voces humanas, por lo cual es necesario entrenar muchos de los sistemas comerciales actuales para que reconozcan la voz de una persona en particular. Aun así, trabajarán de manera confiable si el usuario habla de manera pausada y usa un vocabulario pequeño, previamente definido. Las investigaciones actuales apuntan a superar estas limitaciones y producir sistemas que tengan como características:

- Independencia del hablante.
- La capacidad de manejar la voz sin límite de vocabulario.
- La capacidad de manejar voz continua, el habla natural a velocidad normal. Hasta ahora no existe sistema, salvo el cuerpo humano, que logre estos tres objetivos.

A pesar de las limitaciones, los sistemas de reconocimiento de voz son usados por trabajadores de fábricas y otras personas que tienen las manos ocupadas mientras que usan el computador.

Es posible comunicar números y órdenes por teléfono para operaciones bancarias automatizadas, verificación de tarjetas de crédito y otras aplicaciones remotas. Reducen las limitaciones de muchos minusválidos pues les permiten dar órdenes verbales a computadores y dispositivos robóticos.

## **Robótica**

En ningún lugar es más visible la tecnología de la inteligencia artificial que en el campo de la robótica, la visión, la audición, el reconocimiento de patrones, la toma de decisiones, la comprensión del lenguaje natural, el habla, están presentes en los robots actuales.

La diferencia más importante de hardware entre los robots y otros computadores son los periféricos de entrada y salida. En vez de enviar la salida a una pantalla o a la impresora, un robot envía órdenes a articulaciones, brazos y otras partes móviles.

La mayoría de los robots cuentan con algún tipo de sensores de entrada, que les permite corregir o modificar sus acciones con base en la retroalimentación del mundo exterior.

Un robot o máquina a la cual se le pide tomar determinadas piezas y colocarlas en una caja, las piezas llegan por una cinta a intervalos regulares y las cajas llenas son reemplazadas por cajas vacías en forma consistente en el mismo lugar, o sea la máquina realiza su tarea en un entorno controlado, una aplicación de este tipo no incluye inteligencia.

Surge una diferencia importante cuando la máquina debe realizar sus tareas en un entorno no controlado, por ejemplo, en la exploración espacial. En nuestro ejemplo supongamos que las piezas montadas llegan en una caja con otros tipos de piezas, en vez de estar montadas en la cinta

transportadora, la tarea de la máquina consiste en reconocer la pieza correcta, dejar de lado las otras piezas, y tomar los objetos buscados.

Suponemos que las piezas se colocan en las cajas en forma arbitraria, la recuperación de las piezas requiere una secuencia única de pasos que se debe desarrollar dentro de la máquina misma, la cual deberá vigilar y entender constantemente la situación ya que es posible que las piezas dentro de la máquina se muevan, necesitando modificarse las actividades requeridas. Todas estas cuestiones caen dentro del ámbito de la Inteligencia Artificial.

Los robots son utilizados en fábricas, para desactivar bombas, soldar tubos en el fondo del mar, realizar trabajos en centrales nucleares, cirugías automatizadas para implantes de cadera, etc.

En el caso de los robots espaciales, estos sustituyen al hombre en tareas que son demasiado peligrosas, difíciles, repetitivas o incluso imposibles para los astronautas, teniendo que cumplir algunas exigencias específicas:

- Resistir un lanzamiento.
- Funcionar en condiciones ambientales difíciles, algunos ejemplos:
  - la baja presión en la órbita provoca que el frío suelde las partes metálicas entre sí.
  - la radiación es diferente de la encontrada en la Tierra, en el espacio, las partículas pesadas hacen que la electrónica digital se comporte mal.
- Funcionar autónomamente. Ya que son manejados lejos de su base, las señales para controlarlos y supervisarlos tienen que viajar durante mucho tiempo produciendo retrasos en las comunicaciones que impiden la tele-operación en tiempo real, por lo tanto, deben ser capaces de funcionar solos y solucionar cualquier problema que ocurra mientras realiza sus tareas.

### Sistemas de bases de datos

Los sistemas de almacenamiento y recuperación de datos representan una aplicación importante de los sistemas de proceso de los lenguajes naturales. El objetivo es poder solicitar información a estos sistemas usando un lenguaje natural en vez de ajustarnos a un lenguaje técnico. También se utilizan técnicas de inteligencia artificial en el proceso de responder al usuario.

Sistemas tradicionales	Sistemas de inteligencia artificial
- Recuperan solo los datos solicitados explícitamente	-Recuperan también información que no fue solicitada explícitamente. <u>Por ejemplo:</u> en las búsquedas legales, si se requiere tener información de casos similares se acostumbra pedir al usuario que identifique una palabra que oriente la búsqueda, si la palabra fuera “menores” un sistema inteligente traería también información de aquellos casos que contiene la palabra “infantes”.
- Recuperan solo los datos solicitados explícitamente.	-Recuperan también información que no fue directamente solicitada, pero está relacionada. <u>Por ejemplo:</u> si se tiene una base de datos con información acerca de los presidentes argentinos y se desea preguntar si algún presidente argentino midió más de 3 metros, el sistema

	<p>tradicional no podría dar respuesta a menos que se guardara la altura de todos los presidentes.</p> <p>Un sistema inteligente, podría responder (NO) sin tener las alturas, ya que razona: si hubiera habido algún presidente de tres metros de altura, al ser un dato significativo estaría guardado en la base de datos.</p>
- Se limitan a contestar literalmente la pregunta hecha.	<p>-Recuperan también información que no fue directamente solicitada, pero está relacionada.</p> <p><u>Ejemplo:</u> solicitamos información acerca de la cantidad de alumnos “promocionados” por el profesor Pérez, nos contesta “cero”; podríamos decir que dicho profesor es bastante exigente, luego preguntamos por los “aplazados” y por los que cursaron la materia, en ambos casos la respuesta es “0”. Ante la situación sospechosa preguntamos si Pérez dictó la materia, la base de datos contesta “no”. Un sistema inteligente debería habernos contestado eso al principio.</p>

## Sistemas expertos

**Sistemas expertos:** Paquetes de software diseñados para ayudar a las personas en situaciones en las que se requiere un experto en un área específica

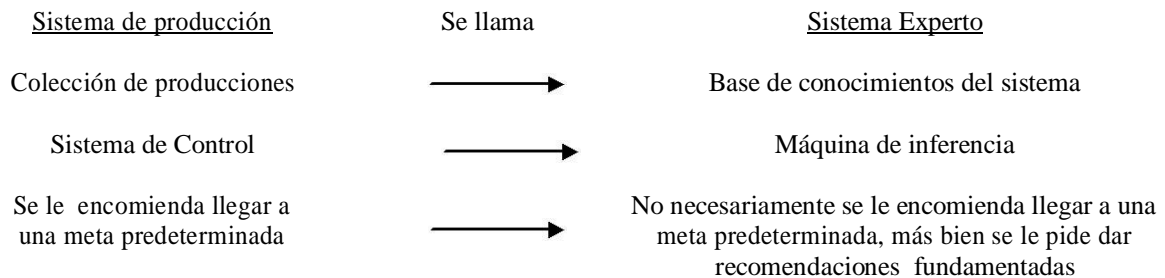
El desarrollo de sistemas expertos es una extensión importante del concepto de bases de datos inteligentes. Se construyen de modo que simulen el razonamiento de causa y efecto que seguirían los expertos en dicha situación. Así, por ejemplo, un sistema experto médico propondrá el mismo procedimiento que un experto médico humano que sabe que debe realizarse una biopsia si se observa alguna anormalidad y si una radiografía indica la presencia de una masa en ese lugar.

Para construir un Sistema Experto se deberá:

- Obtener los conocimientos requeridos de un experto. Esto representa dos cuestiones:
  - Lograr y mantener la cooperación del experto, esto es difícil ya que en general las consultas suelen ser largas y el experto es reacio a compartir sus conocimientos con un sistema que podría sustituirlo.
  - La mayoría de los expertos desconocen el proceso de razonamiento que siguen para sacar conclusiones.
- Organizar los conocimientos obtenidos del experto en un formato compatible con un sistema de software, en general se expresan los conocimientos como una colección de reglas en forma de enunciados SI-ENTONCES, por ejemplo, la regla de nuestro sistema experto médico se puede expresar como:
  - SI se percibe anormalidad y la radiografía indica presencia de masa ENTONCES realizar biopsia.

Existe una similitud entre las reglas de un sistema experto y las producciones de un sistema de producción. La parte “si” de la regla expresa las condiciones para efectuar el enunciado de la parte “entonces”.

De hecho, muchos sistemas expertos son en lo esencial sistemas de producción en los que las reglas obtenidas del experto son las producciones y el sistema de control simula el razonamiento basado en esas reglas.



El motor de inferencia es la estructura de control de un sistema experto, contiene el programa que gestiona la Base de Conocimientos y otros mecanismos necesarios para administrar un sistema de naturaleza interactiva.

El sistema experto se puede dividir en dos componentes, unos de razonamiento y el otro de conocimientos. Gracias a esto, es factible aplicar el sistema de rutinas de razonamiento a distintas situaciones, añadiendo una nueva base de conocimientos, obteniendo nuevos sistemas expertos. De la misma manera que un sistema de control puede aplicarse a distintos problemas con solo sustituir las producciones por las que corresponden a dichos problemas.

## 8. Redes neuronales artificiales

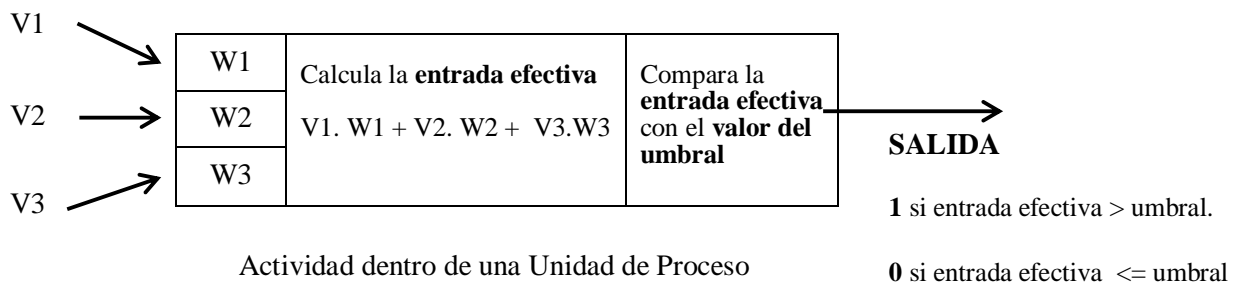
Las redes neuronales artificiales es una de las técnicas que utiliza la Inteligencia Artificial para simular el comportamiento inteligente del hombre, ya que poseen un gran potencial para resolver problemas complejos.

Las unidades centrales de proceso que ejecutan secuencias únicas de instrucciones no pueden percibir ni razonar en niveles comparables con los del multiprocesador que es la mente humana, por tal razón se recurre a unidades de procesamiento interconectadas, llamadas neuronas (por su semejanza funcional con las neuronas de los seres humano), las cuales reciben, procesan y transmiten señales, tal cual lo hacen las neuronas en los sistemas biológicos vivos.

Las *redes neuronales* artificiales están formadas por muchos procesadores individuales, que llamaremos *unidades de proceso*.

Cada unidad de proceso es un dispositivo que produce una salida 1 o 0, según la *entrada efectiva* de la misma exceda un cierto valor umbral.

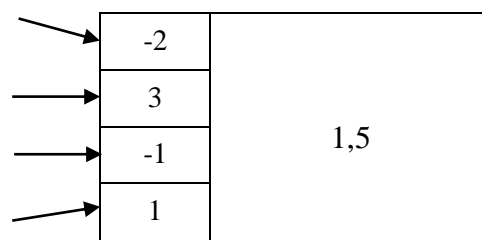
La *entrada efectiva* es una suma ponderada de las entradas reales



$v_1$ ,  $v_2$  y  $v_3$  son las **entradas** a esta unidad de proceso (son salidas de otras unidades de proceso), pueden ser 0 o 1 y cada entrada está asociada a un determinado valor llamado **peso** ( $w_1$ ,  $w_2$ ,  $w_3$ ).

Estos pesos pueden ser positivos o negativos según la entrada correspondiente tenga un efecto de excitación o de inhibición sobre la unidad receptora. Si el peso es negativo, un 1 en esa entrada reduce la suma ponderada, tendiendo a mantener la suma ponderada por debajo del umbral, en cambio un peso positivo, aumenta la posibilidad de que la suma exceda el valor umbral. Por lo tanto, ajustando los valores de los pesos en toda la red neuronal, podemos programarla para que responda a diferentes entradas de una manera determinada.

Veamos algunos ejemplos:

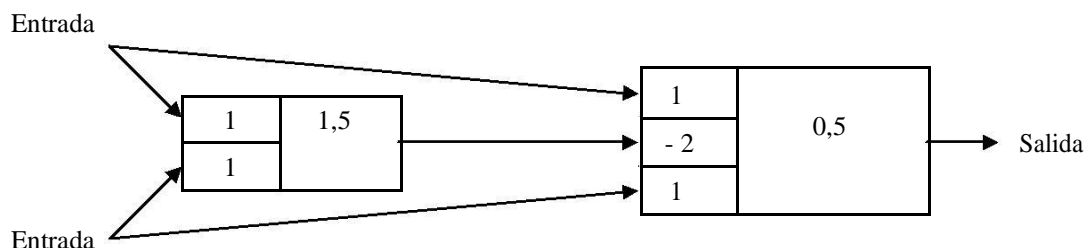


La figura anterior representa una unidad de proceso con cuatro entradas y un valor umbral de 1.5. Las entradas tienen los pesos siguientes: la primera se pondera con el valor -2, la segunda con 3, la tercera con -1 y la cuarta con 1.

Si la unidad recibe las entradas 1,1,1 y 0, su entrada efectiva será  $(1)(-2) + (1)(3) + (1)(-1) + (0)(1) = 0$ , y por lo tanto su salida será 0.

En cambio si la unidad recibe 0,1,1 y 1, su entrada efectiva será  $(0)(-2) + (1)(3) + (1)(-1) + (1)(1) = 3$ , que supera el umbral y por lo tanto su salida será 1.

La siguiente red neuronal, está programada para producir una salida de 1 si sus dos entradas son distintas y una salida de 0 en caso contrario:



# Contenido

INTELIGENCIA ARTIFICIAL .....	2
1. Introducción. ....	2
¿Qué es la inteligencia? .....	2
¿Pueden pensar las máquinas? .....	2
2. Definición de Inteligencia Artificial .....	2
3. Prueba de Turing .....	3
4. Enfoques de la Inteligencia Artificial .....	4
5. Razonamiento.....	4
1) Colección de estados .....	5
2) Colección de producciones o reglas .....	6
3) Sistema de control .....	7
6. Una máquina “inteligente” .....	8
6.1. Análisis de imágenes .....	9
6.2 Árboles de búsqueda .....	10
6.3 Empleo de la heurística .....	13
7. Aplicaciones de la Inteligencia Artificial .....	17
Procesamiento de lenguajes naturales .....	17
Reconocimiento automático de la voz .....	17
Robótica .....	18
Sistemas de bases de datos .....	19
Sistemas expertos.....	20
8. Redes neuronales artificiales .....	21