

# INTELIGENCIA ARTIFICIAL

**Luis Emilio Cabrera Crot**

lcabrera@ubiobio.cl

**Universidad del Bío Bío**

Facultad de Ciencias Empresarias

Ingeniería de Ejecución en Computación e Informática

## Lecture 4



# PREVIOUSLY ON AI...

- Solución de Problemas mediante búsqueda.
- Tipos de Problemas.
- Problemas bien definidos y soluciones.

# EN EL CAPÍTULO DE HOY

- ¿Qué estamos buscando?.
- Estrategias de búsqueda no informada.
- Comparativo de búsquedas.
- Actividad: Implementado Búsquedas.

# ¿QUÉ ESTAMOS BUSCANDO?

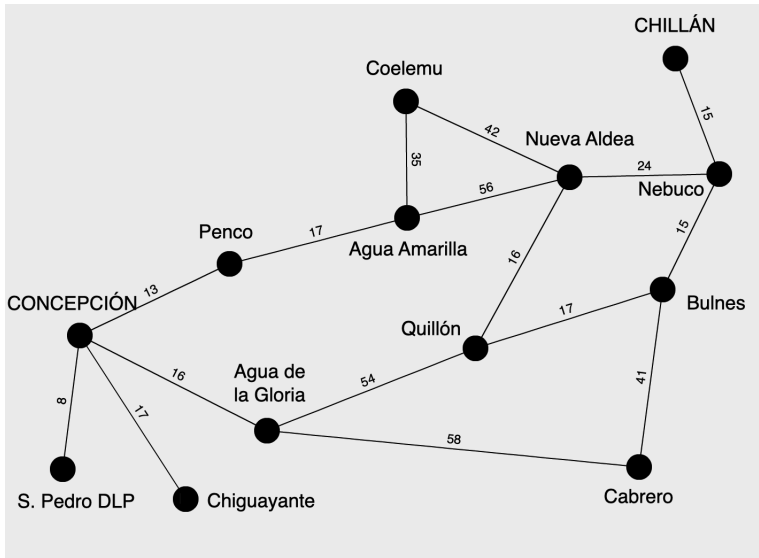
- Recordemos que tenemos un **problema**, en el cual queremos alcanzar una meta a través de una **solución**.
- El problema es la descripción del ambiente al cual nos enfrentamos.
- La solución es una secuencia de acciones que nos lleva a nuestra meta.

## RECORDEMOS CON UN EJEMPLO

- Veamos el problema típico de la ruta más corta.
- Este problema es muy común (aka WAZE, Google Maps).
- Se requiere encontrar la ruta más corta desde un punto **A** a un punto **B**.

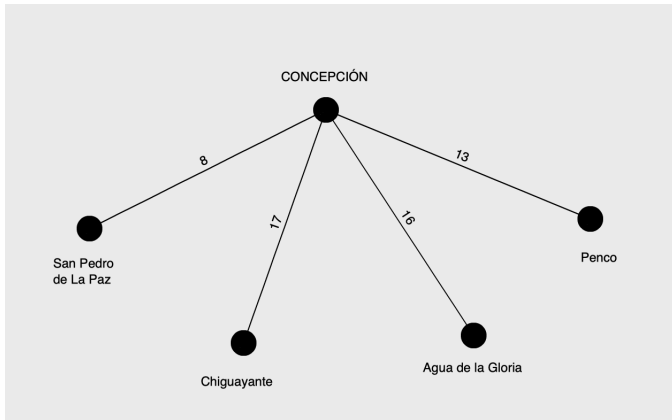
# PROBLEMA

Ruta más corta entre Concepción y Chillán.



# DEFINIENDO EL PROBLEMA

- Estados: 13 (ciudades)
- Árbol de búsqueda:



Estructura de datos para los árboles de búsqueda:

- **Estado:** El estado del espacio de estados al que corresponde un nodo.
- **Nodo Padre:** El nodo en el Árbol de búsqueda que ha generado este nodo.
- **Operador:** El operador que se aplicó para generar el nodo actual.
- **Profundidad:** El número de pasos a lo largo del camino desde el estado inicial.
- **Costo del Camino:** El costo de un camino desde el estado inicial al nodo, indicado por los punteros a los padres.



# ¿CÓMO ENCONTRAR UN BUEN ALGORITMO?

Medidas para evaluar el rendimiento de una estrategia de búsqueda:

- Completitud.
- Optimización.
- Complejidad Temporal
- Complejidad Espacial.

## DEFINICIONES

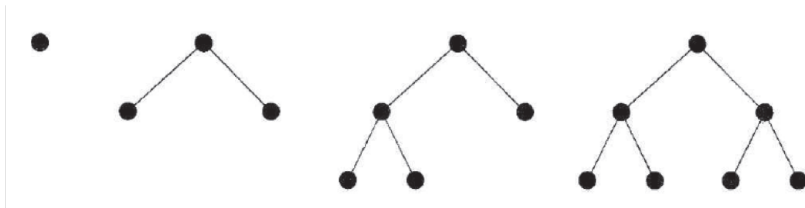
Para medir las complejidades, utilizaremos las siguientes variables:

- $b$ : Máximo número de hijos.
- $d$ : Profundidad primera solución.
- $m$ : Longitud máxima de caminos.

- Una búsqueda no informada ocurre cuando el agente no tiene información sobre su avance, sólo de la meta.
  - Búsqueda por Anchura.
  - Búsqueda por Profundidad.
  - Búsqueda Costo Uniforme.
  - Búsqueda Limitada Profundidad.
  - Búsqueda Profundidad Iterativa.
  - Búsqueda Bidireccional.

# BÚSQUEDA POR ANCHURA

Búsqueda que expande todos los hijos de cada nodo padre que revisa.



Es posible implementarla mediante FIFO (First In First Out).

Medidas para evaluar el rendimiento de una estrategia de búsqueda

- **Compleitud:** Sí, encuentra la solución (en caso de que exista).
- **Optimización:** Sólo si costo de paso es 1.
- **Complejidad Temporal:**  $\Theta(b^d)$ .
- **Complejidad Espacial:**  $\Theta(b^d)$ .

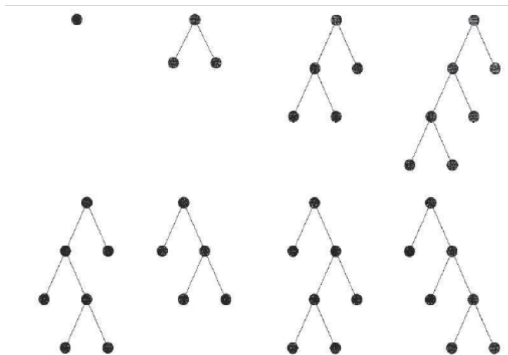
# BÚSQUEDA POR ANCHURA

Exponencial no es una buena medida.

Profundidad	Nodos	Tiempo	Memoria
0	1	1 milisegundo	100 bytes
2	111	0,1 segundos	11 KB
4	11.111	11 segundos	1 MB
6	$\approx 1.000.000$	18 minutos	111 MB
8	$\approx 100$ millones	31 horas	11 GB
10	$\approx 10$ mil millones	128 días	1 TB
12	$\approx 1$ billón	35 años	111 TB
14	$\approx 100$ billones	3.500 años	11.111 TB

## BÚSQUEDA POR PROFUNDIDAD

Búsqueda que elige un camino y lo sigue hasta el final.



Es posible implementarla mediante LIFO (Last In First Out).

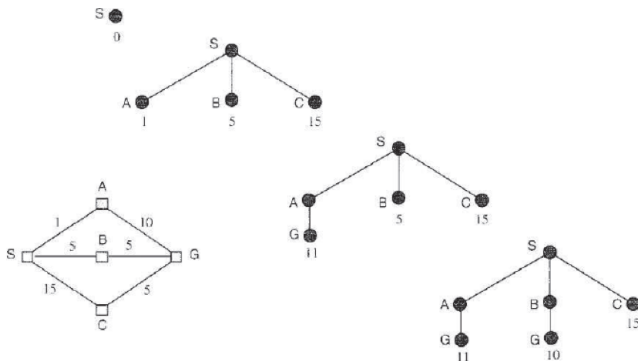
# BÚSQUEDA POR PROFUNDIDAD

Medidas para evaluar el rendimiento de una estrategia de búsqueda

- **Compleitud:** No cuando  $m = \infty$ .
- **Optimización:** Sólo si costo de paso es 1.
- **Complejidad Temporal:**  $\Theta(b^m)$ .
- **Complejidad Espacial:**  $\Theta(b * m)$ .

# BÚSQUEDA CON COSTO UNIFORME

Igual a Anchura, pero se expanden los nodos en orden de costo.





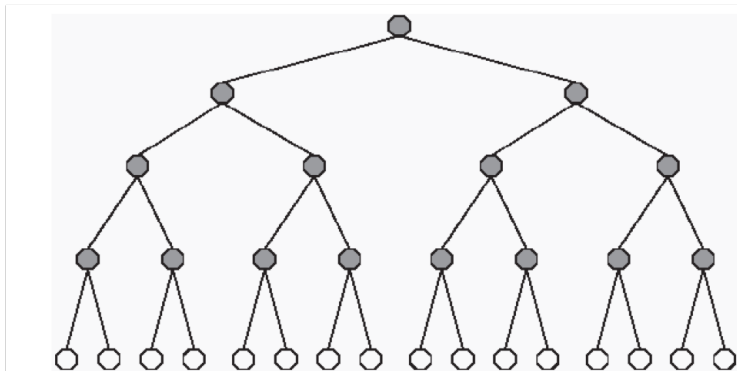
# BÚSQUEDA CON COSTO UNIFORME

Medidas para evaluar el rendimiento de una estrategia de búsqueda

- **Compleitud:** Sí.
- **Optimización:** Sí.
- **Complejidad Temporal:**  $\Theta(b^d)$ .
- **Complejidad Espacial:**  $\Theta(b^d)$ .

# BÚSQUEDA POR PROFUNDIDAD LIMITADA

Búsqueda por profundidad limitada por profundidad  $L$ .



No siempre es posible calcular un  $L$  apropiado.

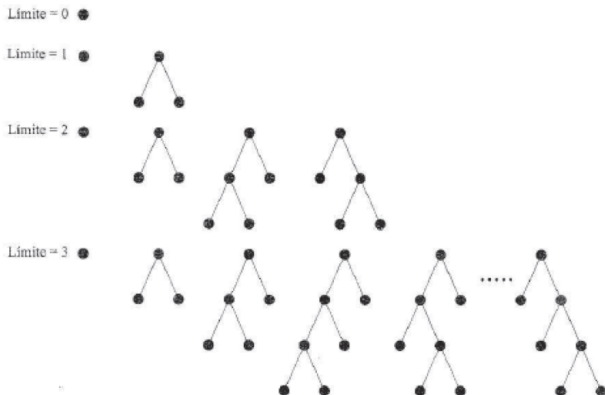
# BÚSQUEDA POR PROFUNDIDAD LIMITADA

Medidas para evaluar el rendimiento de una estrategia de búsqueda

- **Compleitud:** No si  $L$  es pequeño.
- **Optimización:** No.
- **Complejidad Temporal:**  $\Theta(b^l)$ .
- **Complejidad Espacial:**  $\Theta(b * l)$ .

# BÚSQUEDA POR PROFUNDIDAD ITERATIVA

Búsqueda por profundidad limitada por profundidad  $L$ , con  $L$  incremental.



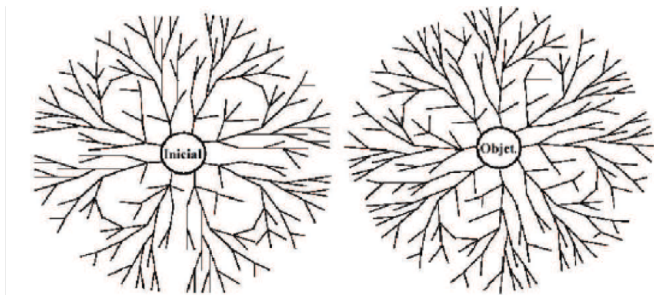
# BÚSQUEDA POR PROFUNDIDAD ITERATIVA

Medidas para evaluar el rendimiento de una estrategia de búsqueda

- **Complejidad:** Sí.
- **Optimización:** Sí.
- **Complejidad Temporal:**  $\Theta(b^d)$ .
- **Complejidad Espacial:**  $\Theta(b * d)$ .

# BÚSQUEDA BIDIRECCIONAL

Busca desde el inicio y desde el final.



No siempre factible. Operador debe ser reversible.

Medidas para evaluar el rendimiento de una estrategia de búsqueda

- **Compleitud:** Sí.
- **Optimización:** Sólo si costo de paso es 1.
- **Complejidad Temporal:**  $\Theta(b^{d/2})$ .
- **Complejidad Espacial:**  $\Theta(b^{d/2})$ .

# COMPARATIVO

Criterio	Tiempo	Espacio	¿Es óptima?	¿Es completa?
Amplitud	$b^d$	$b^d$	✓	✓
C. Uniforme	$b^d$	$b^d$	✓	✓
Profundidad	$b^m$	$b * m$	✗	✗
Prof. Limitada	$b^L$	$b * L$	✗	✓ (cuando $L \geq d$ )
Prof. Iterativa	$b^d$	$b * d$	✓	✓
Bidireccional (Cuando aplica)	$b^{d/2}$	$b^{d/2}$	✓	✓



# ACTIVIDAD

Dado el juego de placas deslizables:

5	4	
6	1	8
7	3	2

Estado de origen.

1	2	3
8		4
7	6	5

Estado meta.

Construya un programa que permita explorar el árbol de estados de este problema:

- Por Amplitud
- Por Profundidad Iterativa
- En forma Bidireccional

# INTELIGENCIA ARTIFICIAL

**Luis Emilio Cabrera Crot**

lcabrera@ubiobio.cl

**Universidad del Bío Bío**

Facultad de Ciencias Empresarias

Ingeniería de Ejecución en Computación e Informática

## Lecture 4

