

# *SWI-Prolog*

## *Resolución de Consultas*

Conceptos de Inteligencia Artificial  
Sistemas Inteligentes Artificiales

Departamento de Ciencias e Ingeniería de la Computación  
Universidad Nacional del Sur

# SWI-Prolog

- Sitio web de **SWI-Prolog**: <https://www.swi-prolog.org/>
- Editores: **PceEmacs** (incorporado en el SWI-Prolog: **emacs/0**)  
**SWI-Prolog-Editor** (Windows)

Predicado a consultar en el intérprete:  
?- **emacs**.

Ver Program Development Tools: <https://www.swi-prolog.org/IDE.html>

- Tracer y GUI-Tracer:
  - **trace/0**: predicado que habilita el modo debug
  - **nodebug/0**: predicado para salir del modo debug
  - **guitracer/0**: predicado que habilita la interfaz de usuario para los trace futuros

# SWI-Prolog

- Comentarios:      % linea simple

/\* multiples  
  lineas \*/

- Ayuda en SWI-Prolog:
  - **help/0**. Predicado sin argumentos que abre una ventana de ayuda.
  - **apropos/1**. Predicado para consultar acerca de un predicado predefinido, cuyo nombre es pasado como argumento.

# SWI-Prolog – Resolución de Consultas

- Una **consulta** en SWI-Prolog corresponde a una **regla sin cabeza**. Ejemplos:

**`:-a`**

**`:-b, c`**

- Para hallar la respuesta a la consulta **`:- a`** se busca que exista una derivación (prueba) para **`a`**.
- Prolog efectúa una **búsqueda hacia atrás (*backward chaining*)** partiendo de la consulta. Esta búsqueda se asemeja a un **recorrido en profundidad**, mediante el cual se construye un **árbol de derivación**.

# SWI-Prolog – Resolución de Consultas

- El intérprete de SWI-Prolog emplea los siguientes **criterios** para la **resolución de consultas**:
  - **Recorre (y selecciona)** las **cláusulas** de un programa **de arriba hacia abajo**.
  - Cuando la consulta a resolver corresponde a una **meta conjuntiva** (conjunción de metas), éstas son resueltas **de izquierda a derecha**.
  - Al seleccionar una cláusula para la **resolución de una meta 'm'**, el intérprete **reemplaza la meta** por el **cuerpo de la cláusula** seleccionada (y continúa el proceso de derivación).

# *Ejemplo – Resolución de Consultas*

## *Árbol de Derivación*

p :- a, b, c.

p :- a, d.

a :- q, h.

a :- h.

d :- j.

d :- i.

h.

j.

Consideremos el siguiente  
programa Prolog

# *Ejemplo – Resolución de Consultas*

## *Árbol de Derivación*

Consideremos el siguiente  
programa Prolog

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

# *Ejemplo – Resolución de Consultas*

## *Árbol de Derivación*

(1)  $p :- a, b, c.$

Enumeramos las cláusulas del programa  
para referenciarlas fácilmente:  
reglas (1-6) y hechos (7-8)

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

Consulta que queremos  
resolver: **determinar si p es  
derivable** a partir del programa

**?- p.**

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

$\{p\}$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

Meta conjuntiva a resolver  
por el intérprete

(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

$?- p.$

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

$\{p\}$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

$?- p.$

Al ser una **única meta**, la **selecciona**

OBS: Recordar que el intérprete resuelve las metas conjuntivas seleccionando las sub-metas de izquierda a derecha

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

$\{p\}$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

$?- p.$

El intérprete **recorre las cláusulas** del programa (de arriba hacia abajo) **buscando una** cláusula **cuya cabeza corresponda a la meta seleccionada**

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

$\{p\}$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

$?- p.$

El intérprete **recorre las cláusulas** del programa (de arriba hacia abajo) **buscando una** cláusula **cuya cabeza corresponda a la meta seleccionada**

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

$\{p\}$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

Utilizando la cláusula seleccionada se continúa la construcción del árbol de derivación:

El nuevo conjunto de metas es tal que **se reemplaza la meta seleccionada por el cuerpo de la cláusula elegida por el intérprete.**

$?- p.$

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

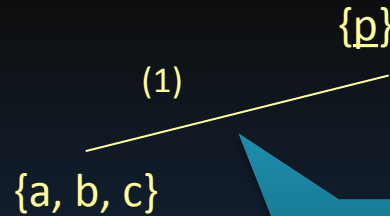
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



Referencia al número de  
cláusula utilizada

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$  ←

(3)  $a :- q, h.$

(4)  $a :- h.$

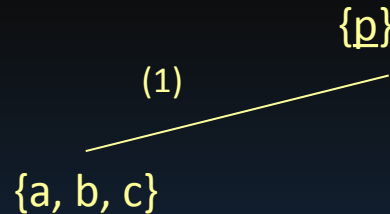
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



¿Qué sucede si existen otras cláusulas que podrían haber sido seleccionadas (cuya cabeza correspondía a la meta seleccionada)?



# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$  

(3)  $a :- q, h.$

(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?- p.



El intérprete deja abierta una **rama pendiente** en el árbol de derivación, para ser **explorada a futuro en caso de ser necesario**.

OBS: Recordar que la construcción del árbol de derivación se realiza siguiendo un recorrido en profundidad!

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?- p.



El intérprete continúa la construcción del árbol de derivación considerando la **nueva meta conjuntiva** pendiente de resolución

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?- p.



El intérprete continúa la construcción del árbol de derivación considerando la **nueva meta conjuntiva** pendiente de resolución

Al existir **múltiples sub-metas**, el intérprete **selecciona la primera** (de izquierda a derecha)

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



El intérprete **recorre las cláusulas** del programa (de arriba hacia abajo) **buscando una** cláusula **cuya cabeza corresponda a la meta seleccionada**

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



El intérprete **recorre las cláusulas** del programa (de arriba hacia abajo) **buscando una** cláusula **cuya cabeza corresponda a la meta seleccionada**

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?- p.



Utilizando la cláusula seleccionada se continúa la construcción del árbol de derivación:

El nuevo conjunto de metas es tal que **se reemplaza la meta seleccionada por el cuerpo de la cláusula elegida por el intérprete.**

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

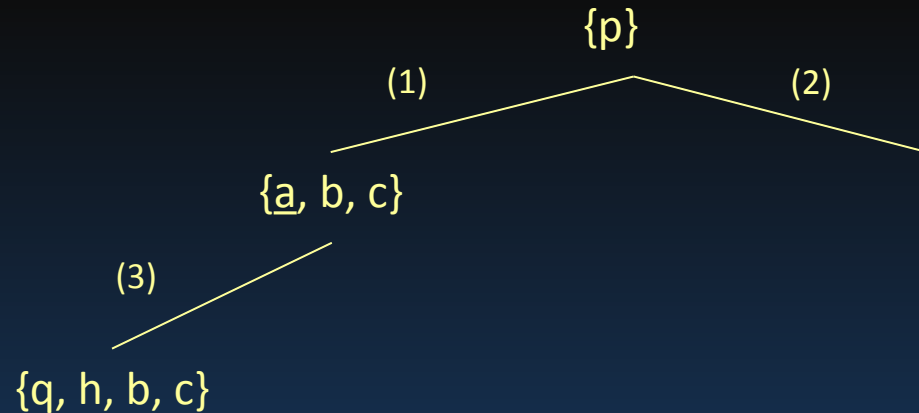
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

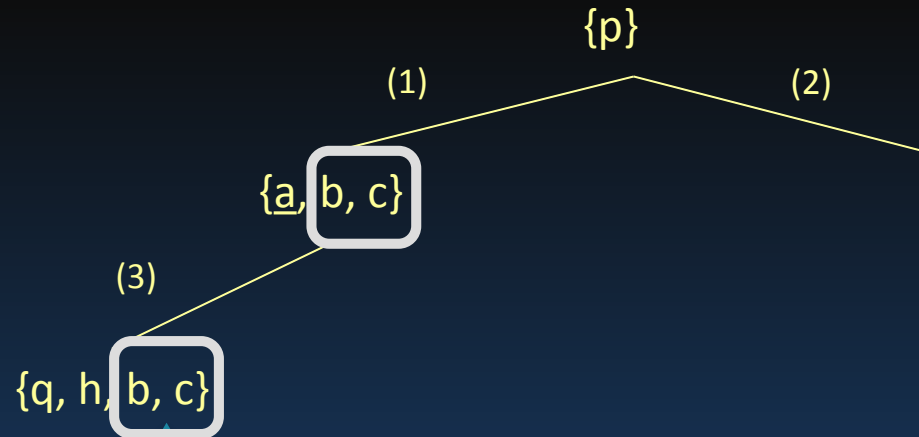
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$





# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

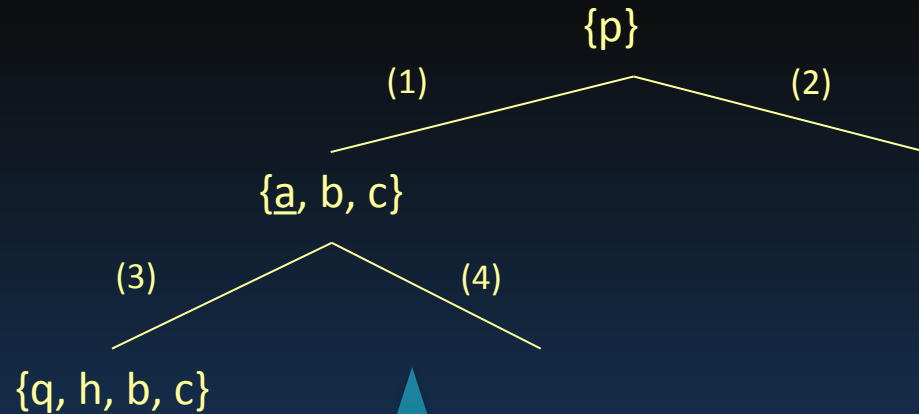
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



Para cada cláusula alternativa el intérprete deja abierta una **rama pendiente** en el árbol de derivación, para ser **explorada a futuro en caso de ser necesario**.

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

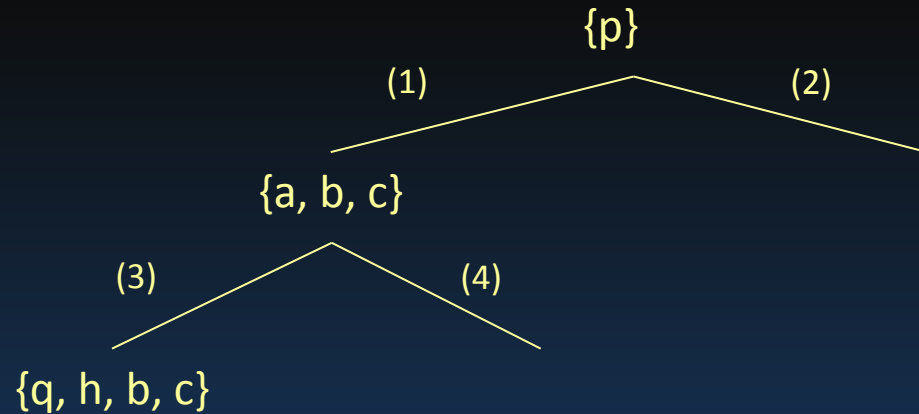
(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



?- p.

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

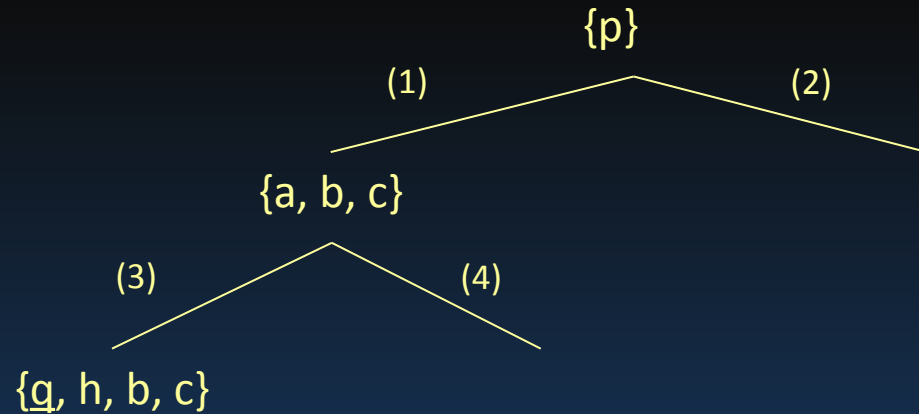
(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



Continúa la construcción del árbol de derivación considerando la **nueva meta conjuntiva** pendiente de resolución, y **seleccionando la sub-meta** que se encuentra más **a la izquierda**

?- p.

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

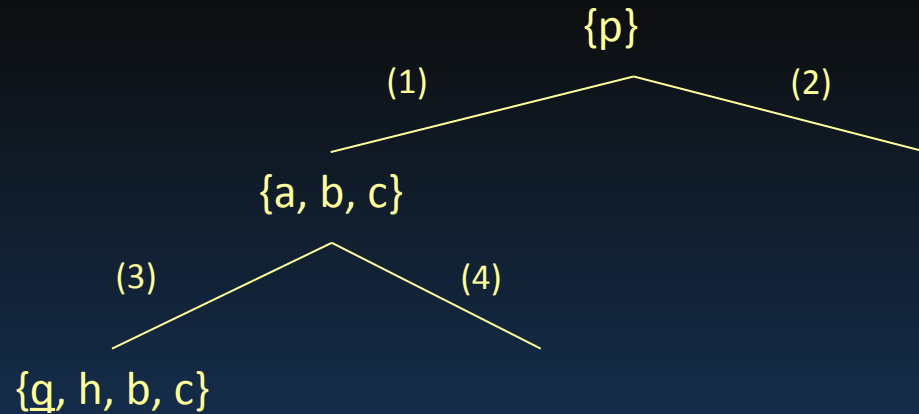
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

$?- p.$



El intérprete **recorre las cláusulas** del programa (de arriba hacia abajo) **buscando una** cláusula **cuya cabeza corresponda a la meta seleccionada**

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

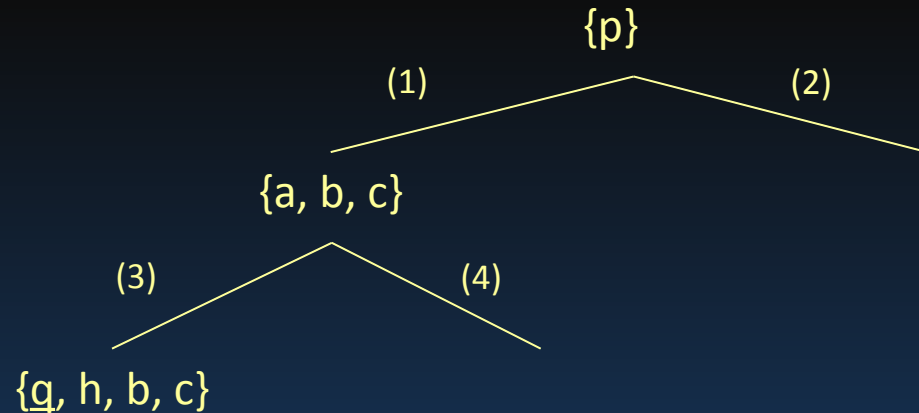
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



El intérprete **recorre las cláusulas** del programa (de arriba hacia abajo) **buscando una** cláusula **cuya cabeza corresponda a la meta seleccionada**

**No existe** tal cláusula!

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

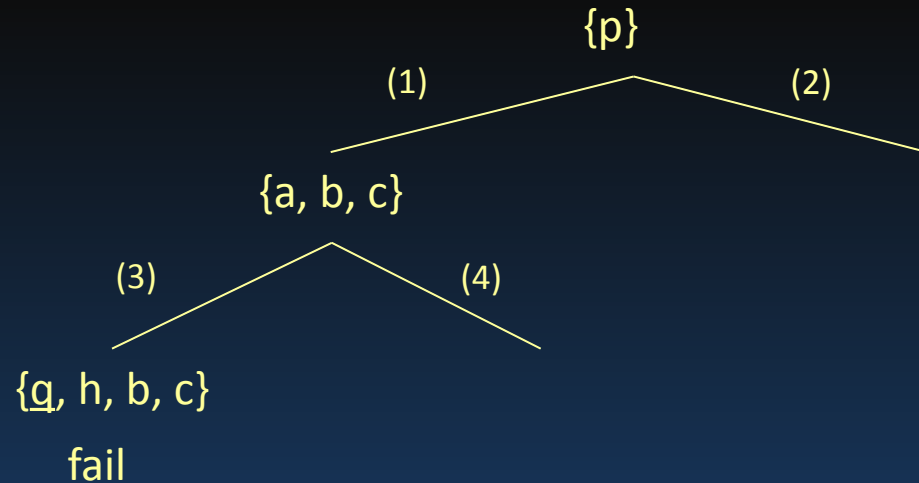
(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



La derivación por esta rama del árbol falla

?- p.

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

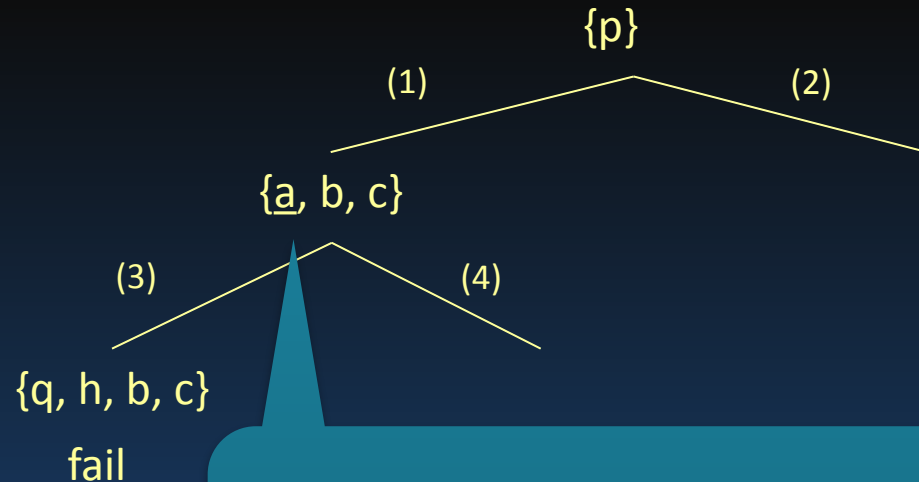
(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



El intérprete realiza **backtracking** hasta el punto más cercano del árbol en el que existan ramas abiertas pendientes de exploración, reanudando el proceso de derivación en ese punto.

Selecciona la sub-meta de más a la izquierda.

?- p.

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

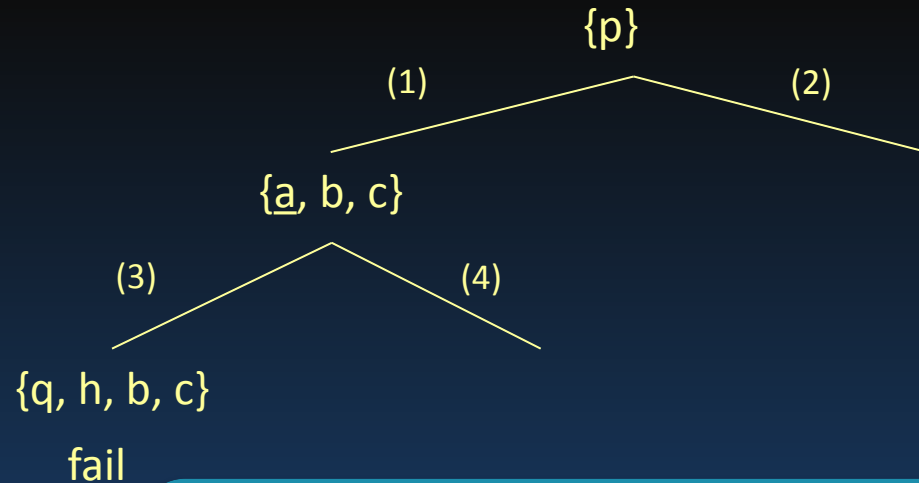
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



El intérprete **selecciona** la cláusula correspondiente a la **rama abierta** que se encontraba **pendiente**



# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

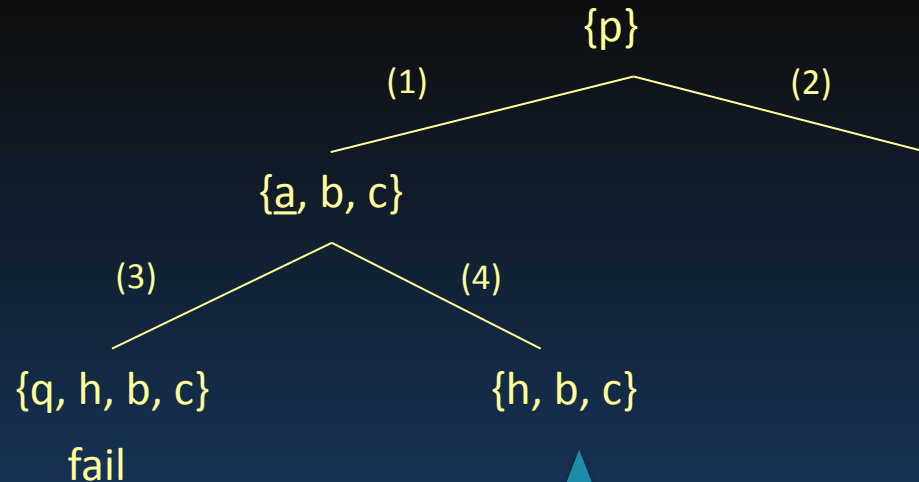
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



El nuevo conjunto de metas es tal que **se reemplaza la meta seleccionada por el cuerpo de la cláusula elegida por el intérprete.**

OBS: Las metas que quedaban pendientes de resolución se mantienen!

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

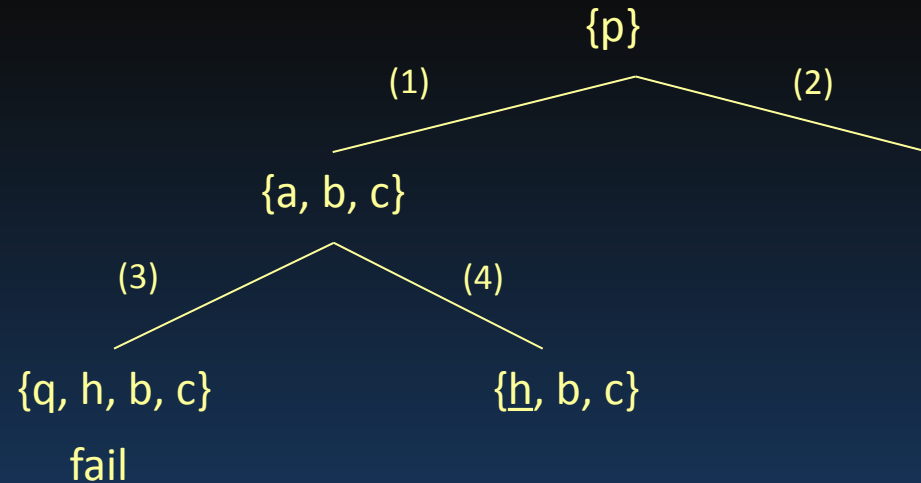
(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



$?- p.$

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

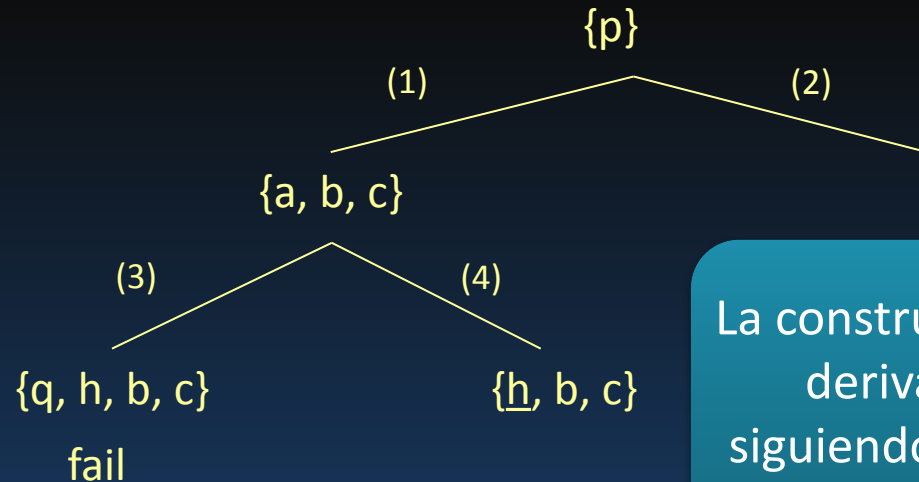
(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



La construcción del árbol de derivación continúa siguiendo la estrategia que vimos hasta el momento...

?- p.

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

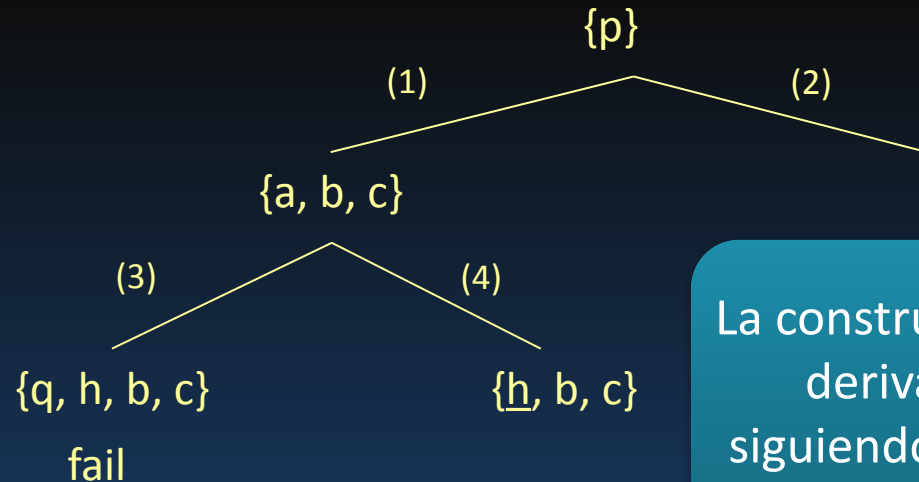
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?- p.



La construcción del árbol de derivación continúa siguiendo la estrategia que vimos hasta el momento...

La **respuesta del intérprete** es obtenida una vez que:

- No quedan metas por resolver (**afirmativa**); ó
- Se exploraron todas las ramas existentes sin éxito (**negativa**)

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p \text{ :- } a, b, c.$

(2)  $p \text{ :- } a, d.$

(3)  $a \text{ :- } q, h.$

(4)  $a \text{ :- } h.$

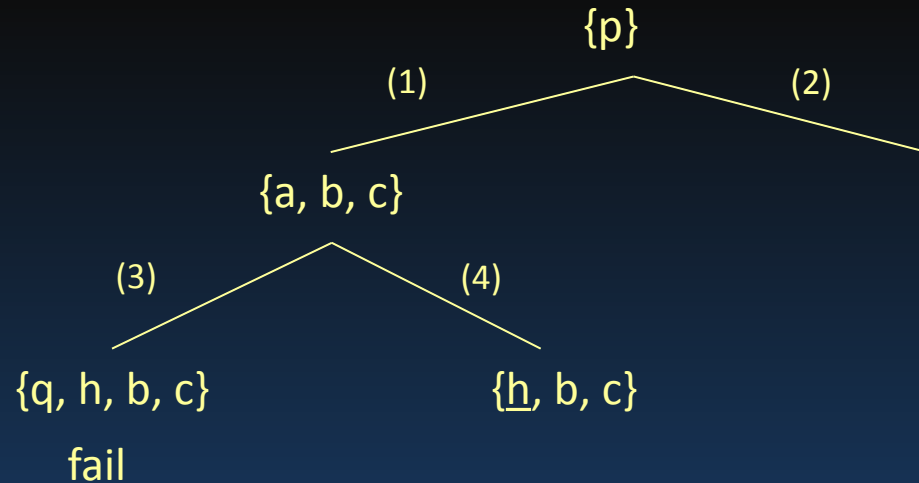
(5)  $d \text{ :- } j.$

(6)  $d \text{ :- } i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p \text{ :- } a, b, c.$

(2)  $p \text{ :- } a, d.$

(3)  $a \text{ :- } q, h.$

(4)  $a \text{ :- } h.$

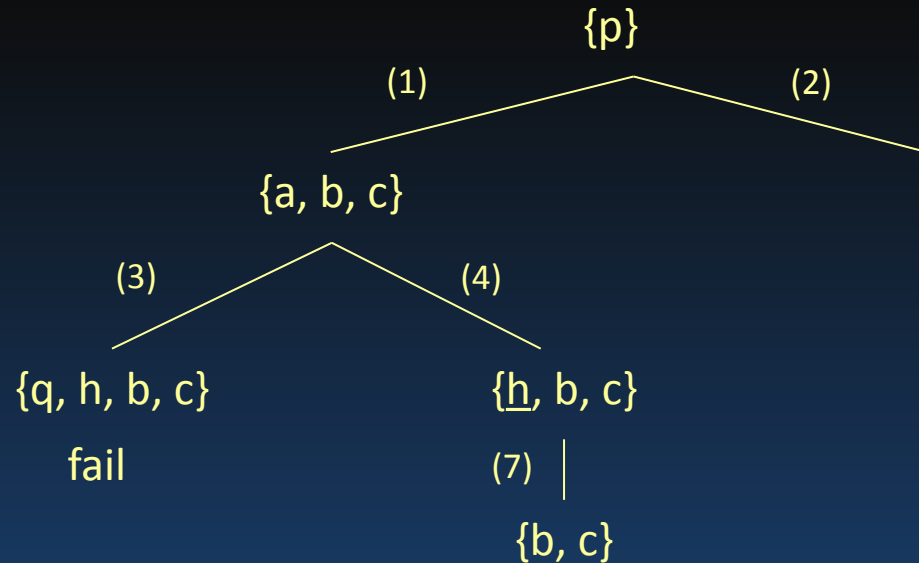
(5)  $d \text{ :- } j.$

(6)  $d \text{ :- } i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

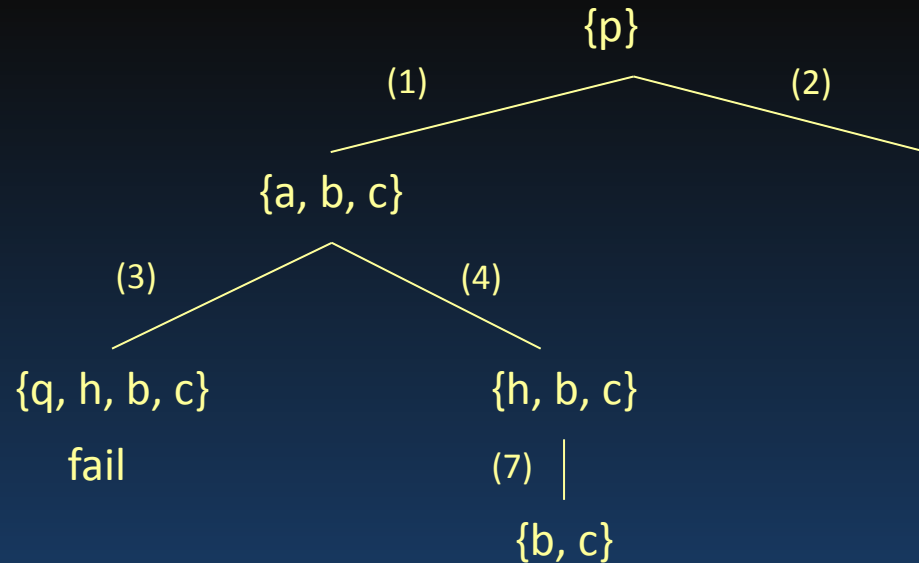
(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



$?- p.$

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p \text{ :- } a, b, c.$

(2)  $p \text{ :- } a, d.$

(3)  $a \text{ :- } q, h.$

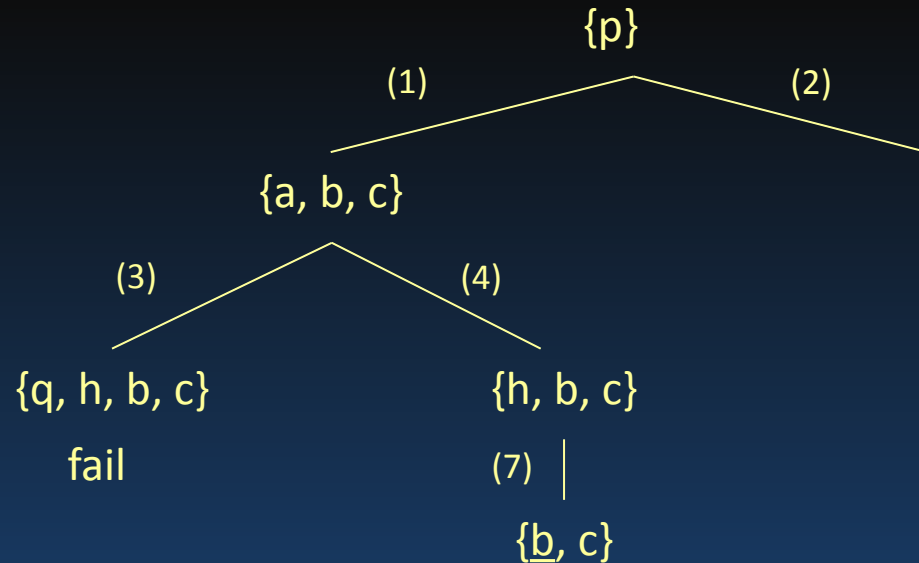
(4)  $a \text{ :- } h.$

(5)  $d \text{ :- } j.$

(6)  $d \text{ :- } i.$

(7)  $h.$

(8)  $j.$



$?- p.$



# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

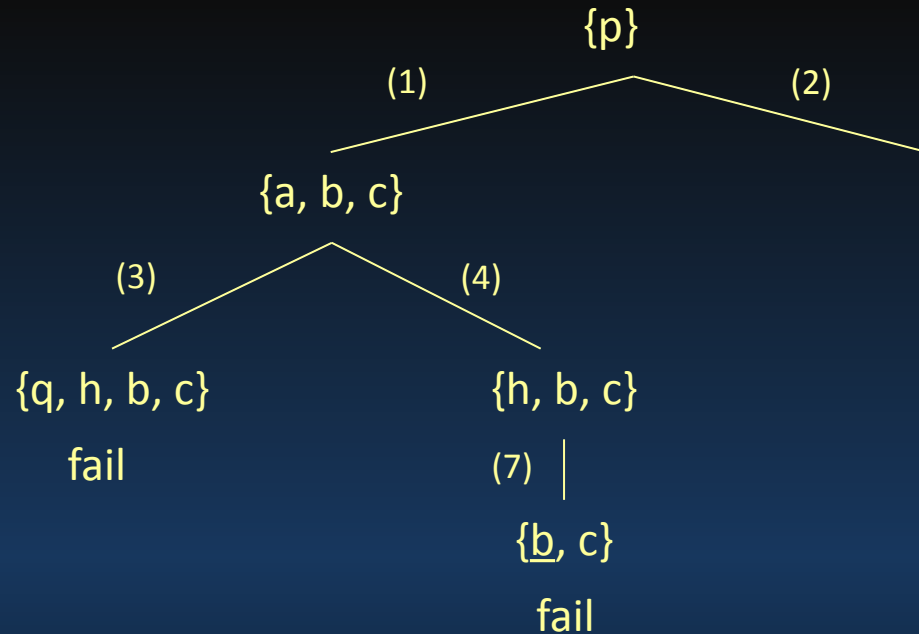
(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



?- p.

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p \text{ :- } a, b, c.$

(2)  $p \text{ :- } a, d.$

(3)  $a \text{ :- } q, h.$

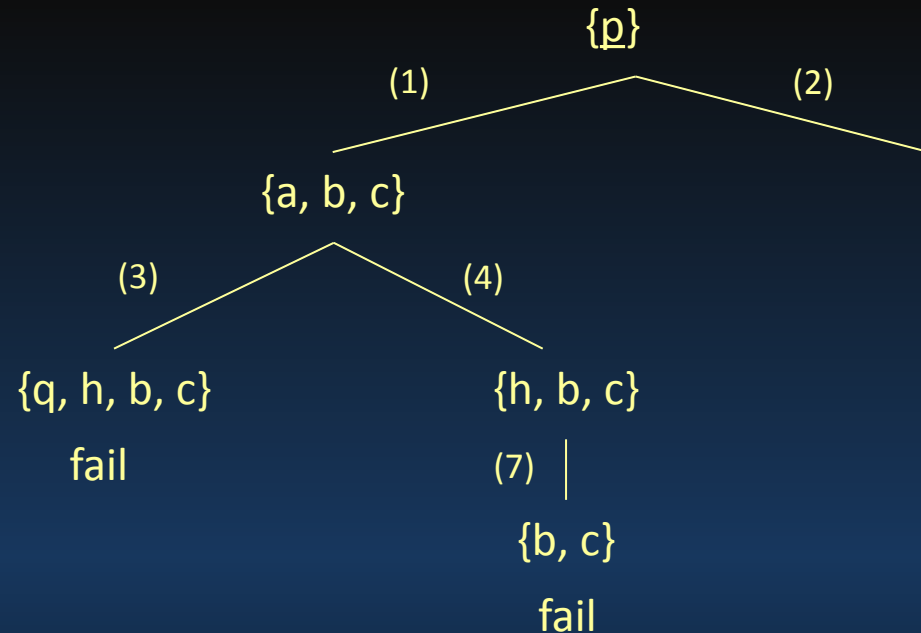
(4)  $a \text{ :- } h.$

(5)  $d \text{ :- } j.$

(6)  $d \text{ :- } i.$

(7)  $h.$

(8)  $j.$



$?- p.$

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

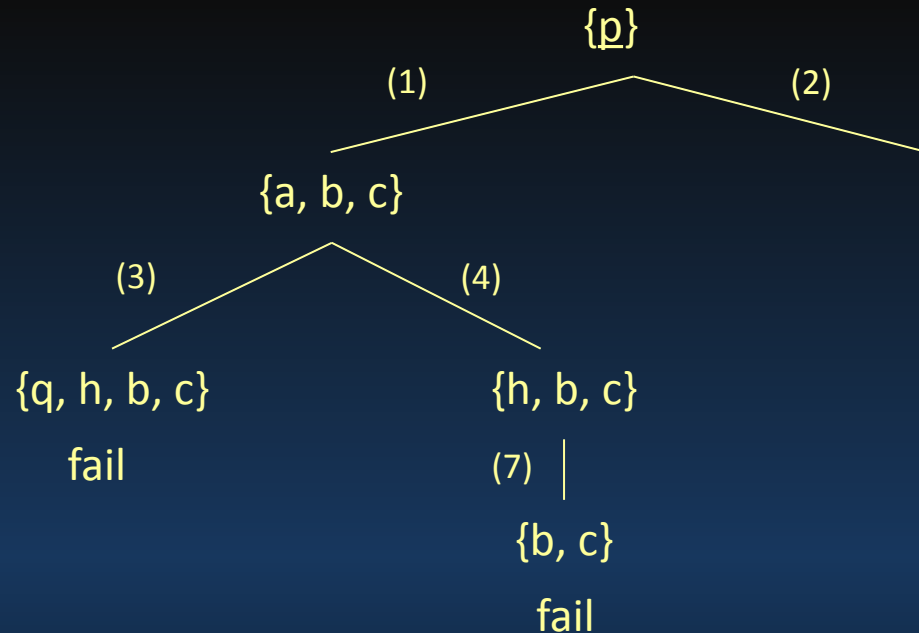
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

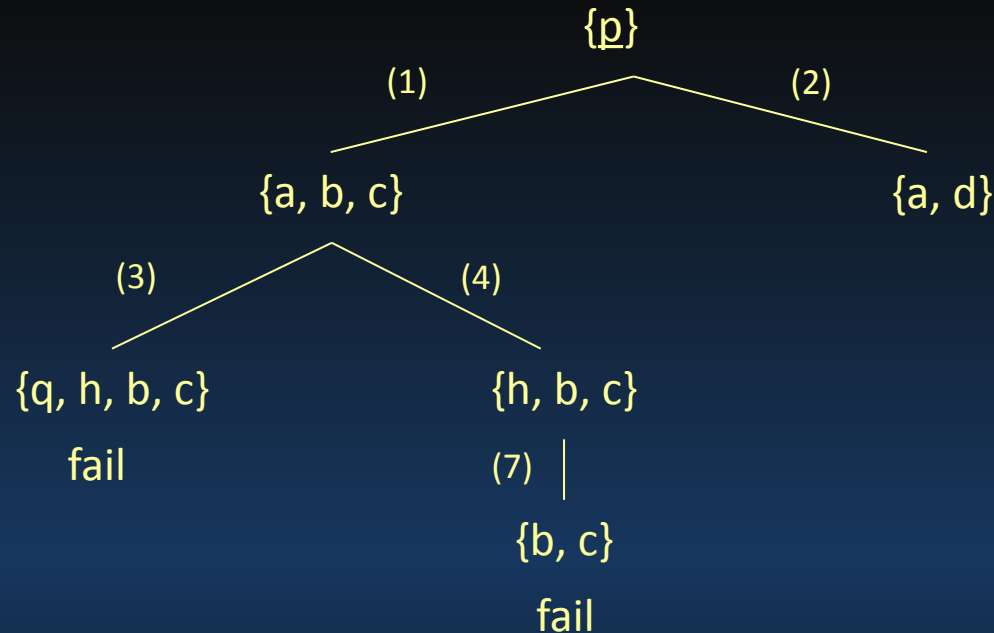
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

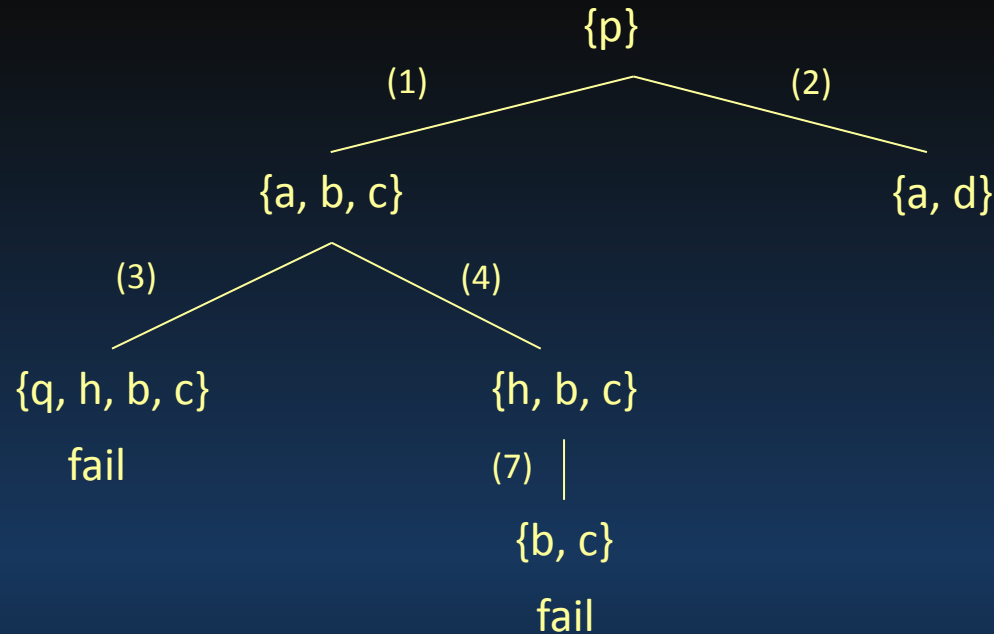
(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



?- p.

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

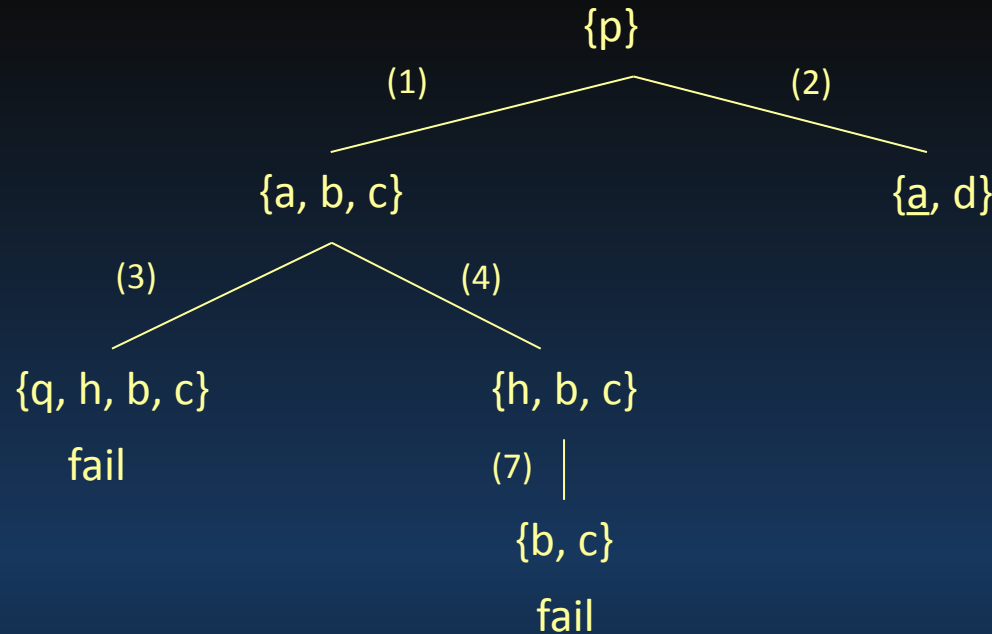
(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



?- p.

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

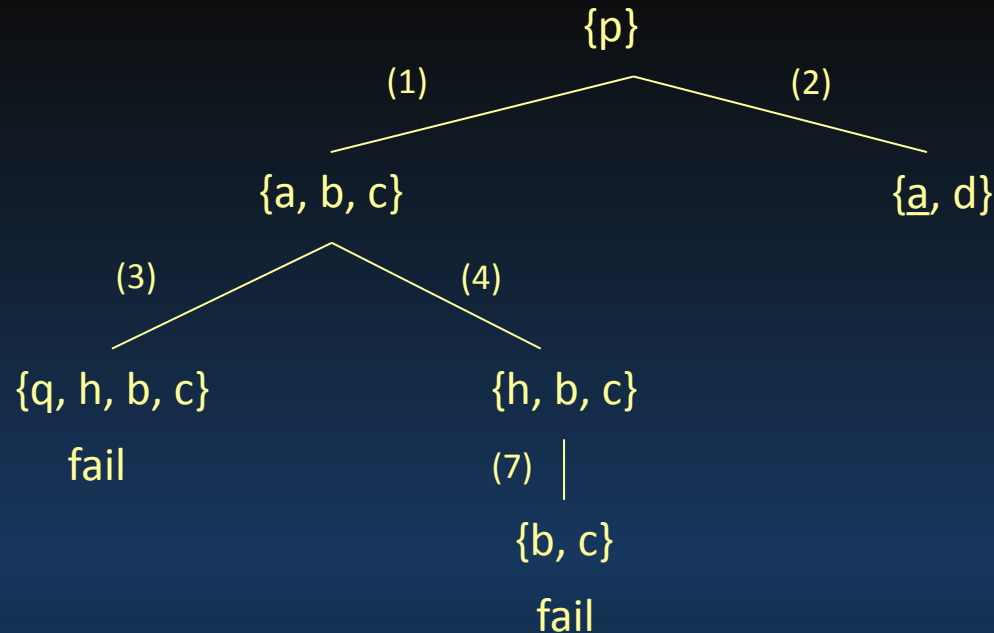
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

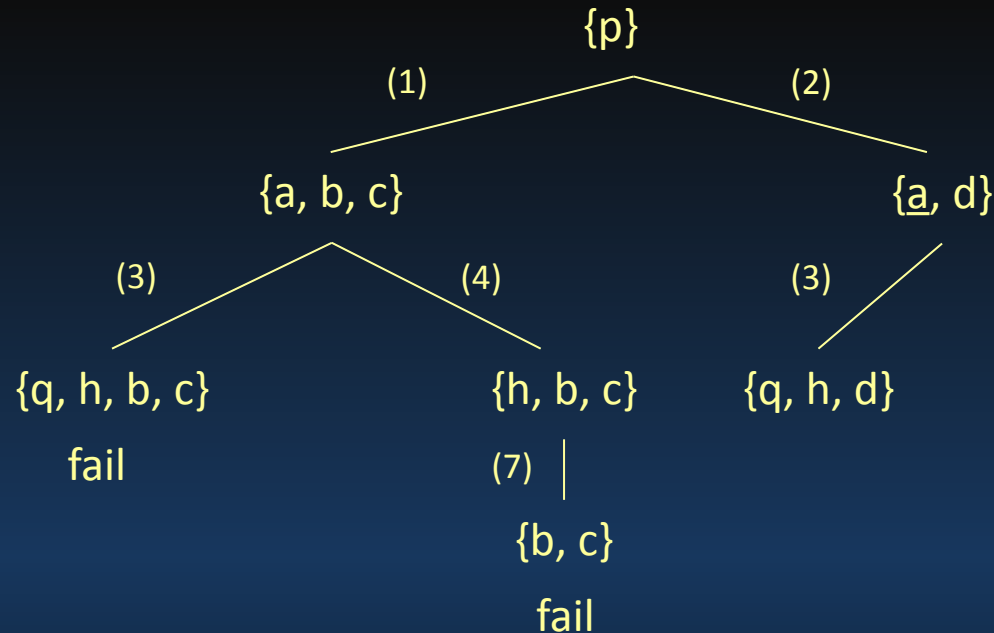
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$





# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$  

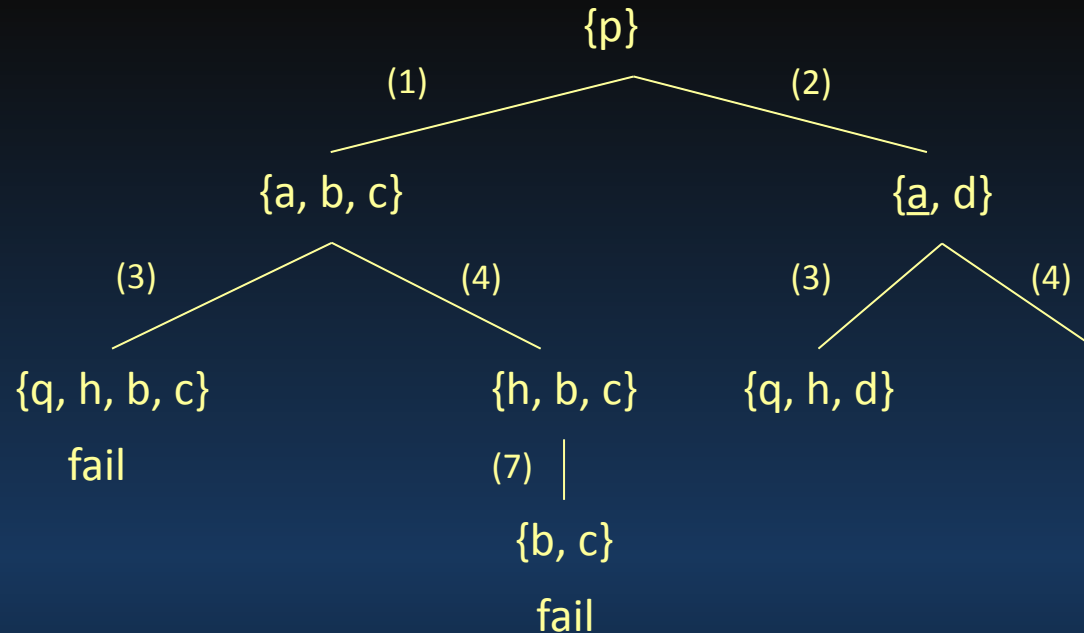
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

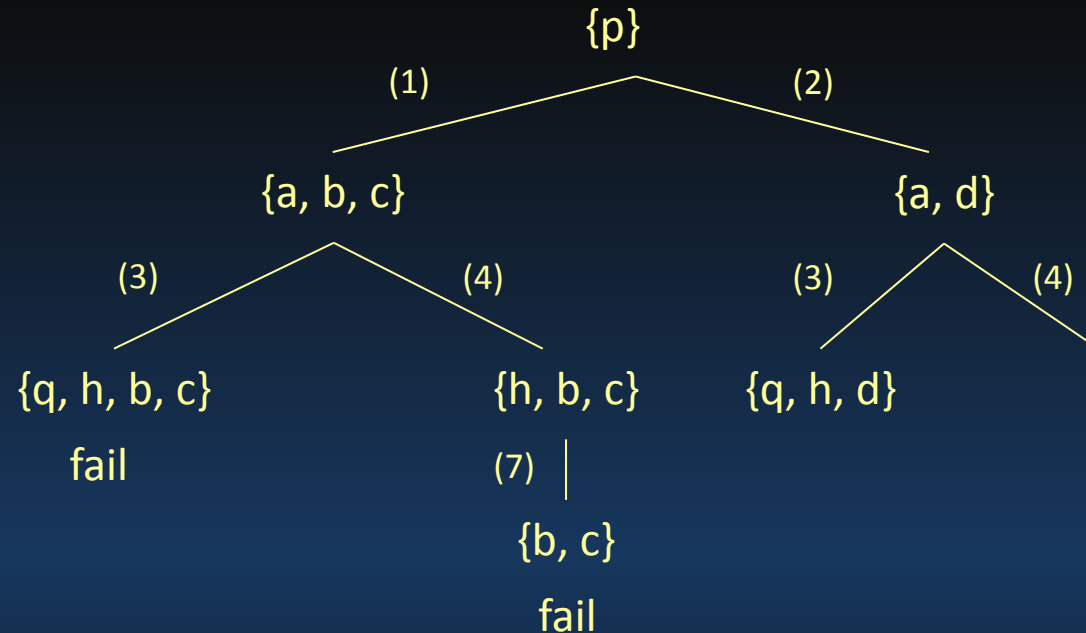
(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



$?- p.$

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

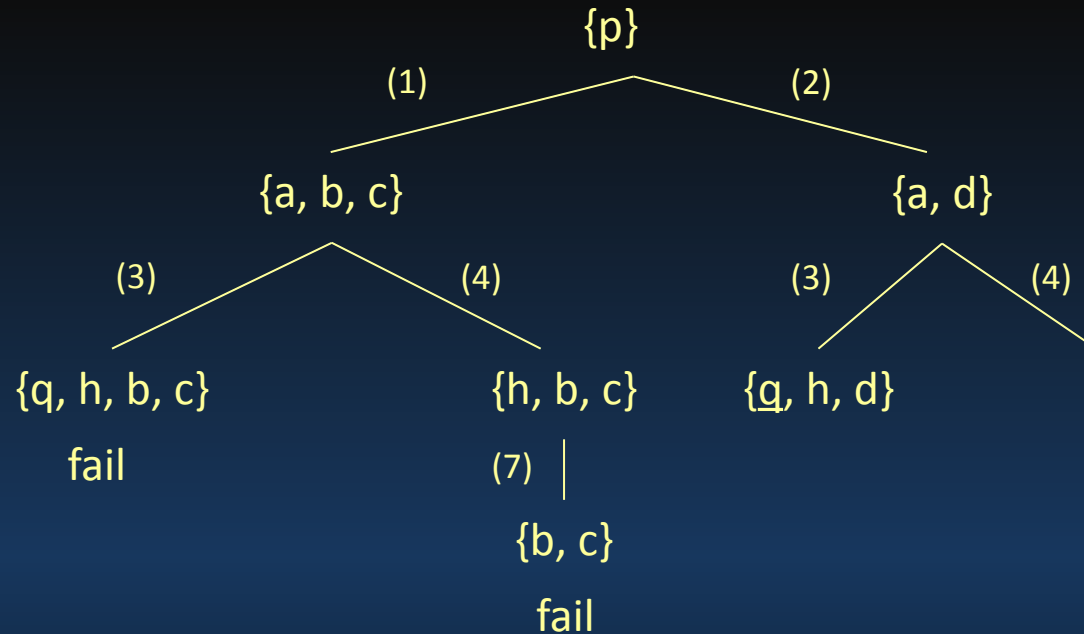
(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



?- p.

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p \text{ :- } a, b, c.$

(2)  $p \text{ :- } a, d.$

(3)  $a \text{ :- } q, h.$

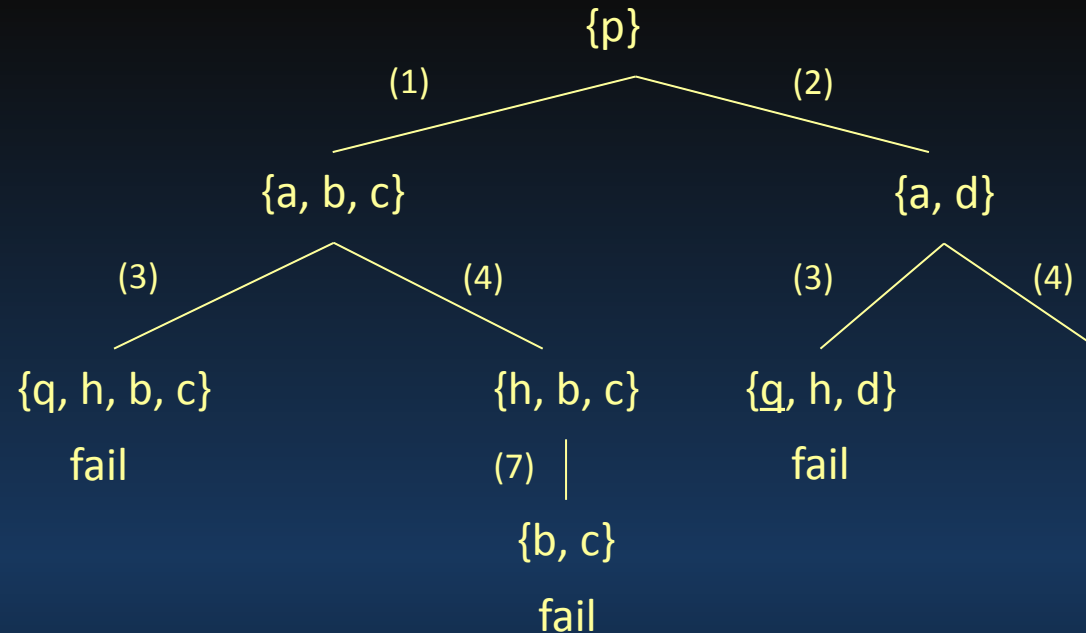
(4)  $a \text{ :- } h.$

(5)  $d \text{ :- } j.$

(6)  $d \text{ :- } i.$

(7)  $h.$

(8)  $j.$



?- p.

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p \text{ :- } a, b, c.$

(2)  $p \text{ :- } a, d.$

(3)  $a \text{ :- } q, h.$

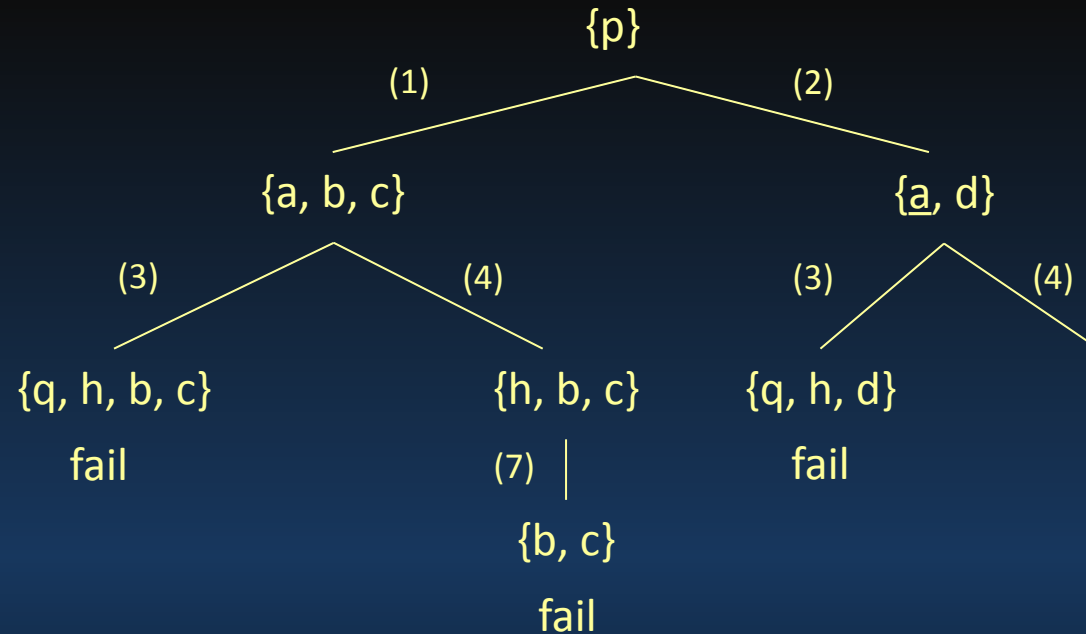
(4)  $a \text{ :- } h.$

(5)  $d \text{ :- } j.$

(6)  $d \text{ :- } i.$

(7)  $h.$

(8)  $j.$



$?- p.$

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

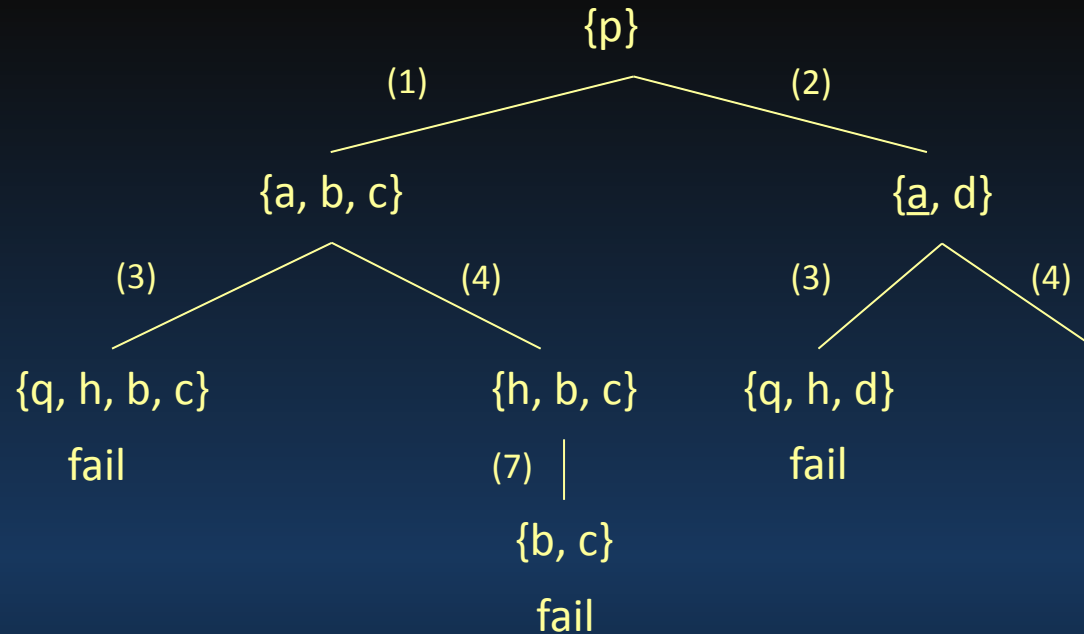
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

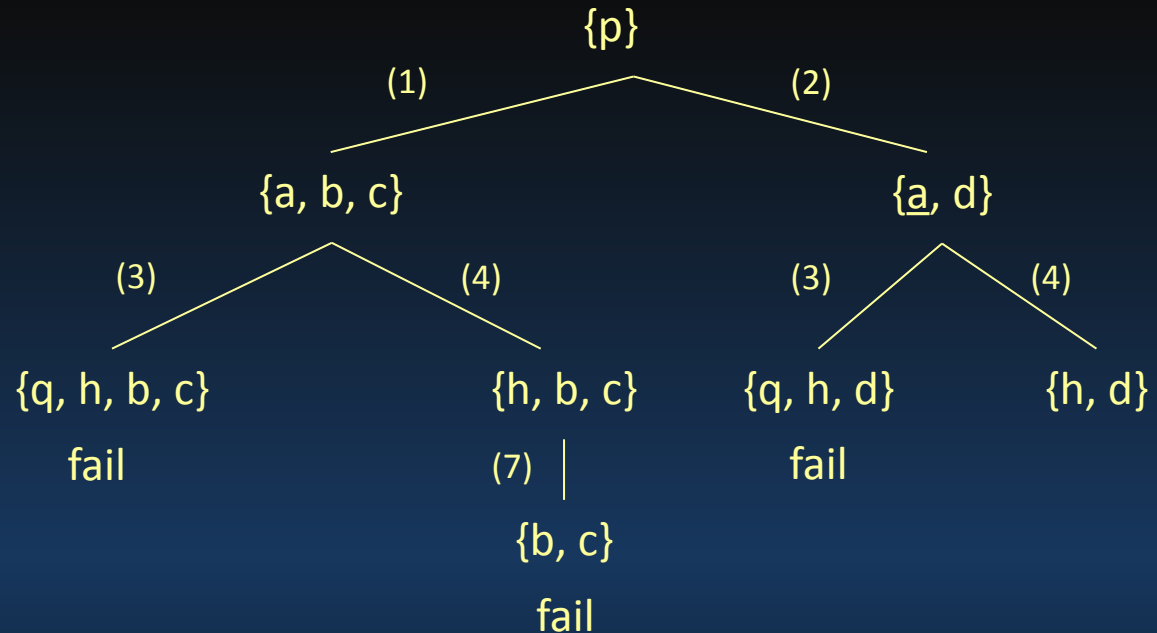
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

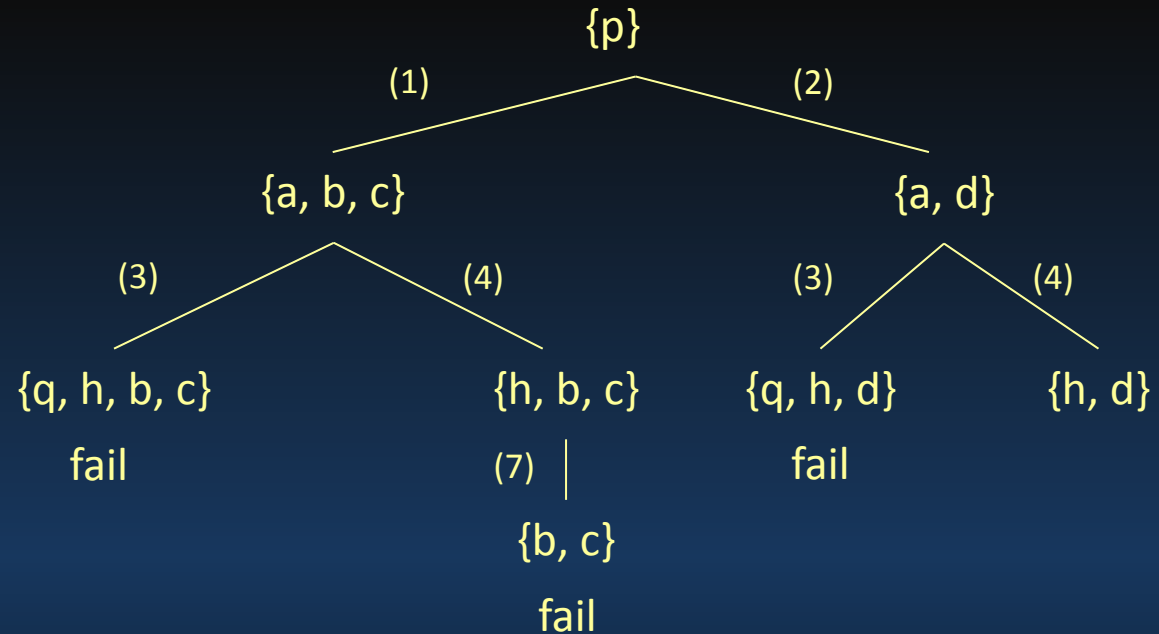
(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



?- p.



# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

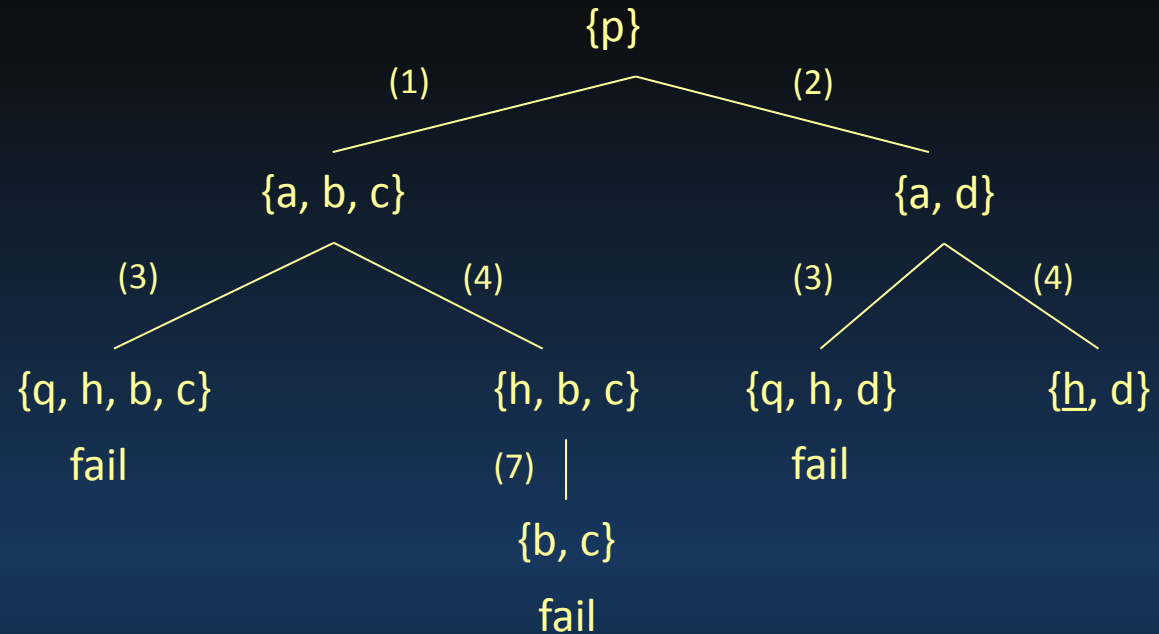
(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



?- p.

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

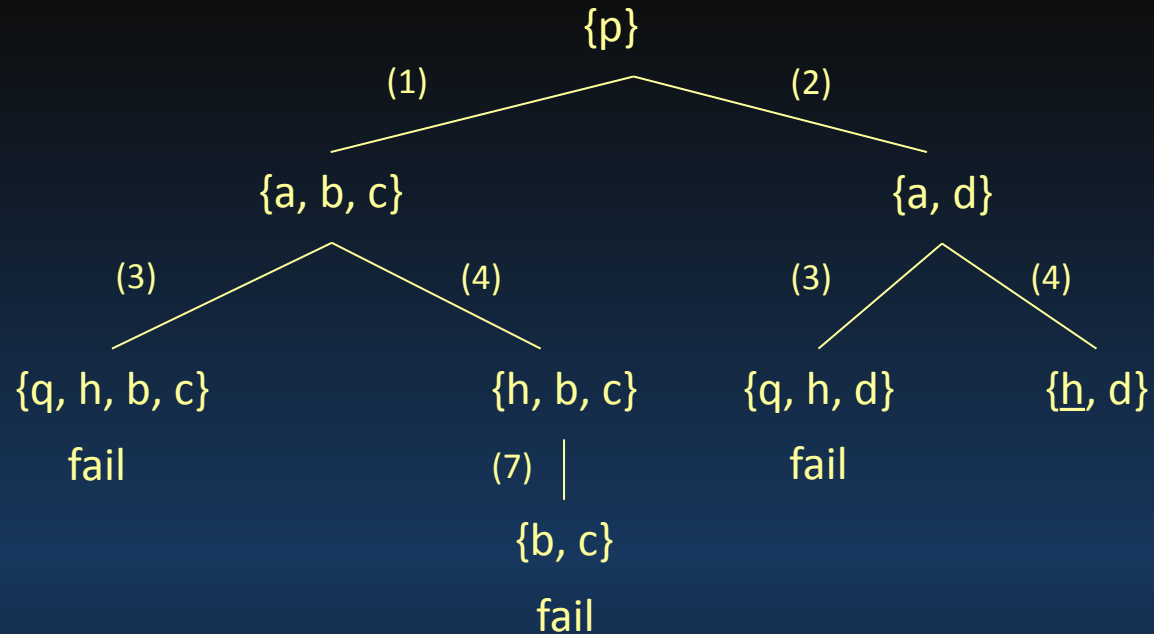
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

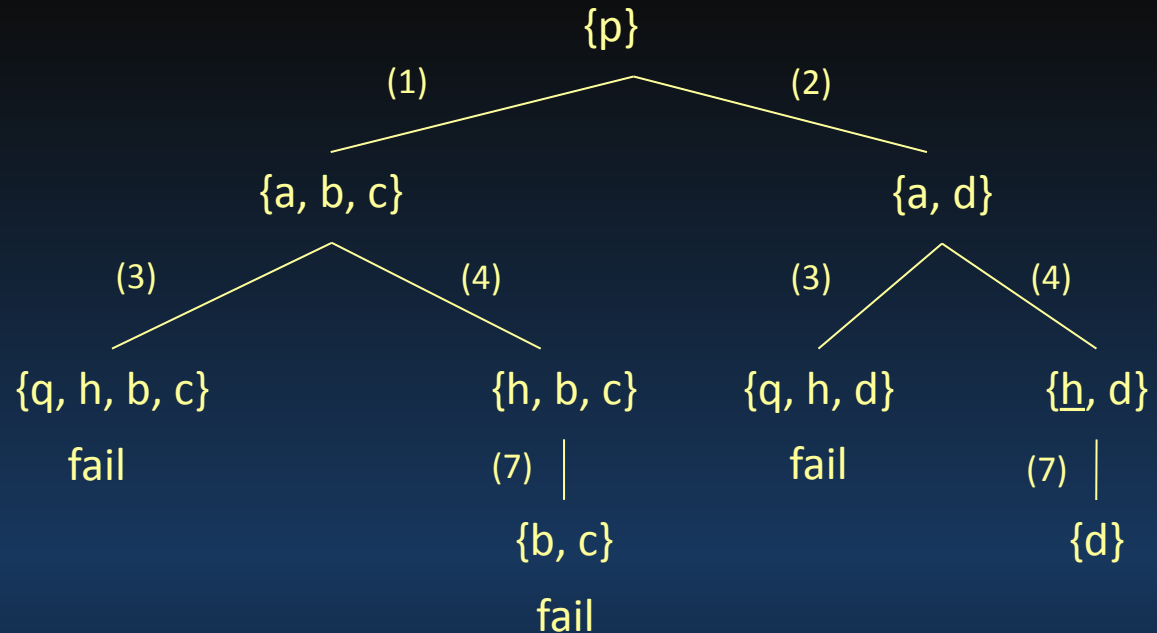
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

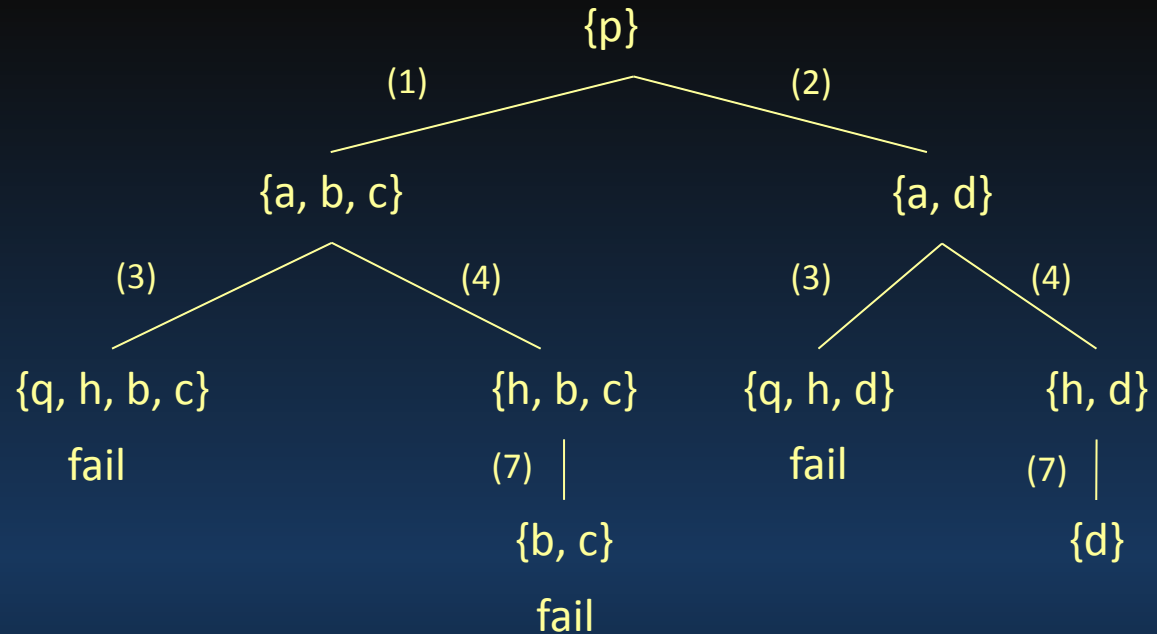
(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



?- p.

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p \text{ :- } a, b, c.$

(2)  $p \text{ :- } a, d.$

(3)  $a \text{ :- } q, h.$

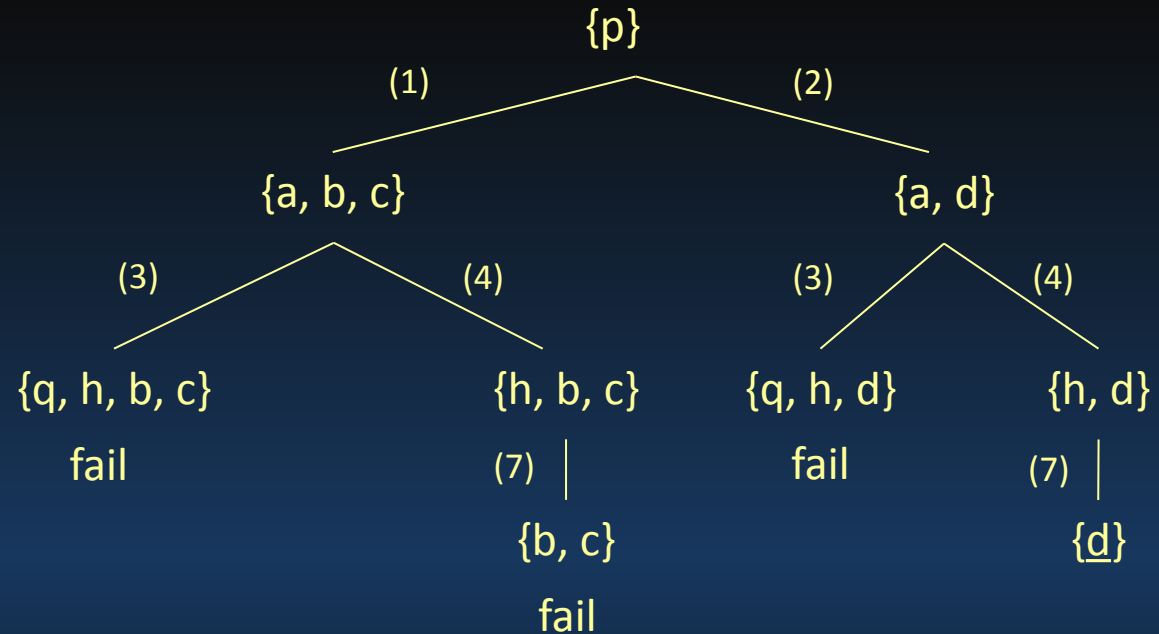
(4)  $a \text{ :- } h.$

(5)  $d \text{ :- } j.$

(6)  $d \text{ :- } i.$

(7)  $h.$

(8)  $j.$



?- p.

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

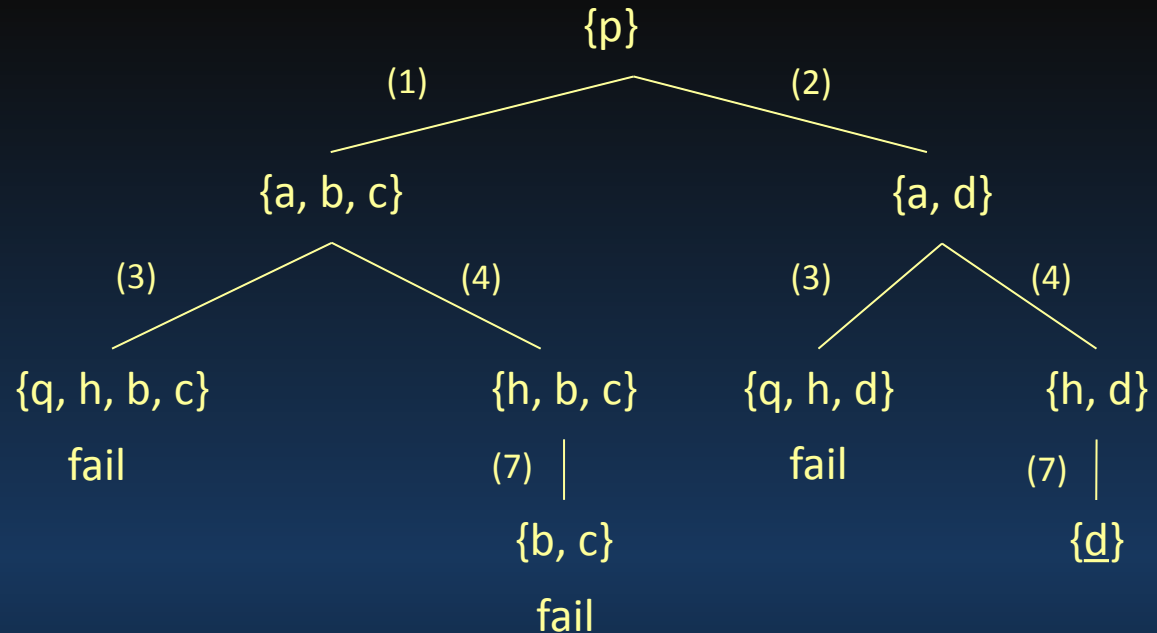
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

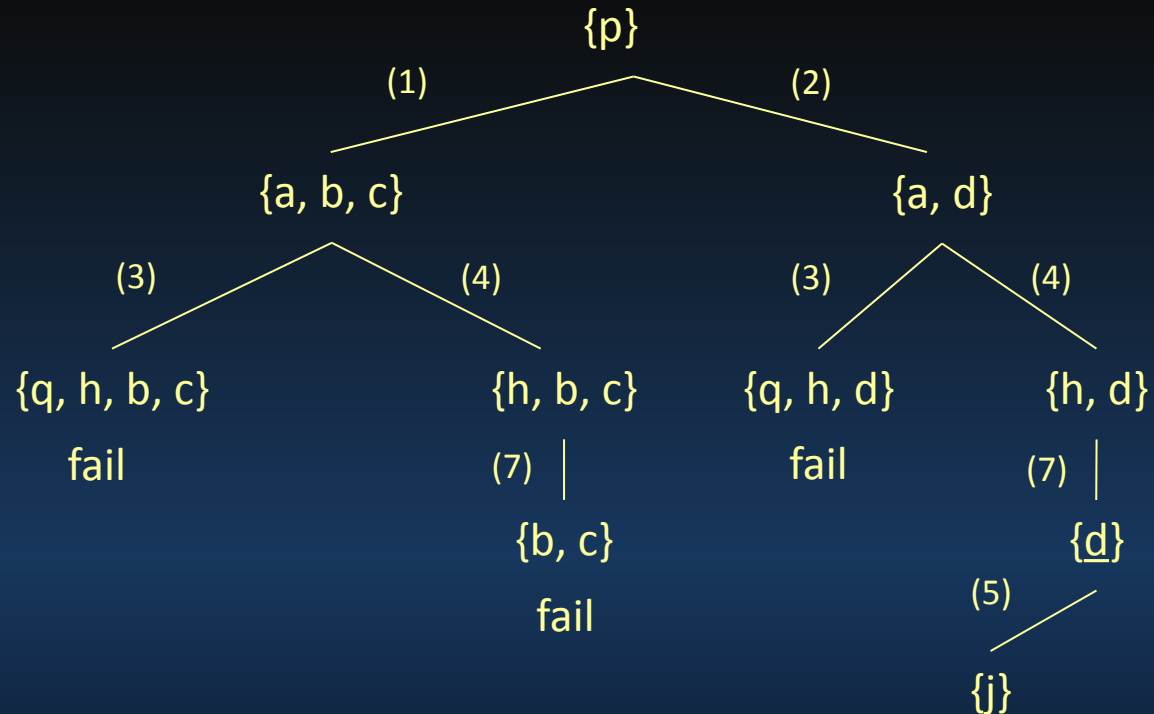
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?-  $p.$



# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

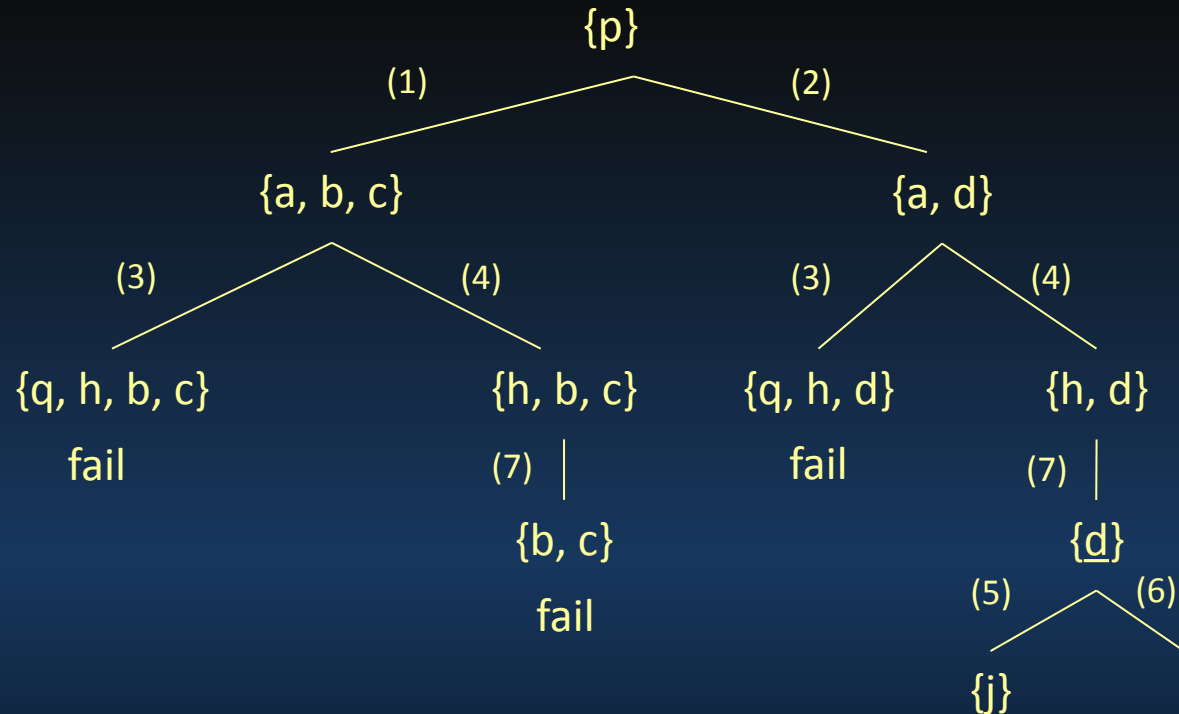
(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$

?- p.





# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

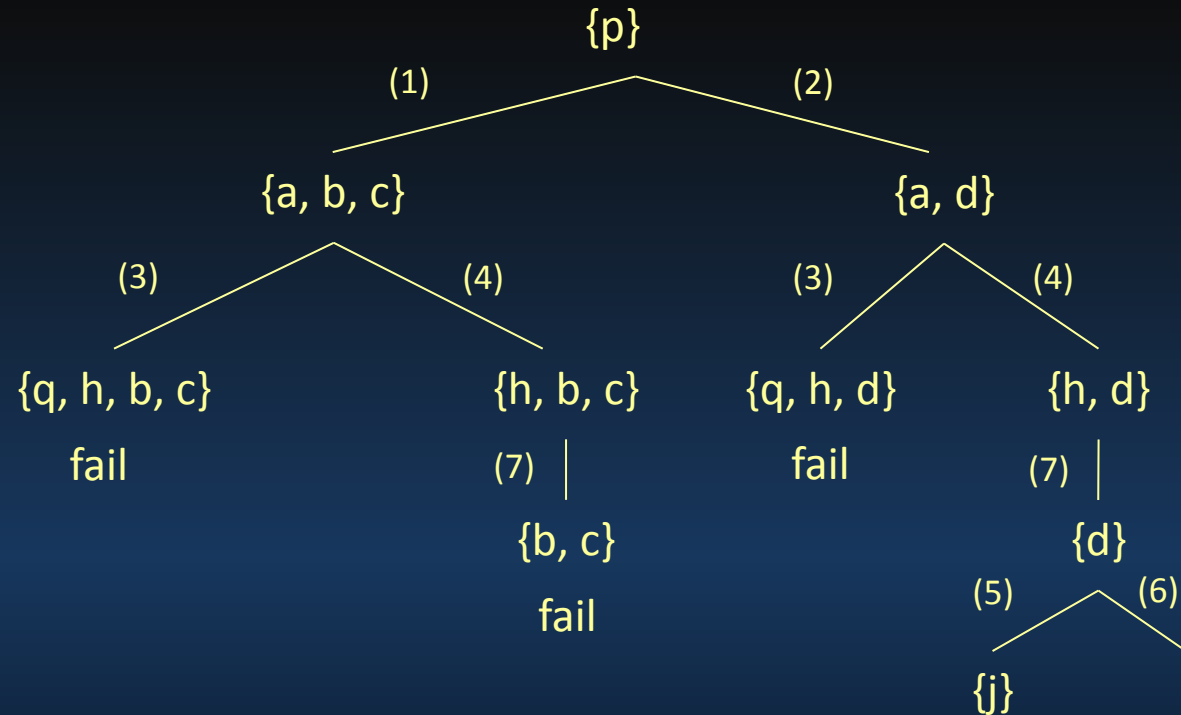
(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



?- p.

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

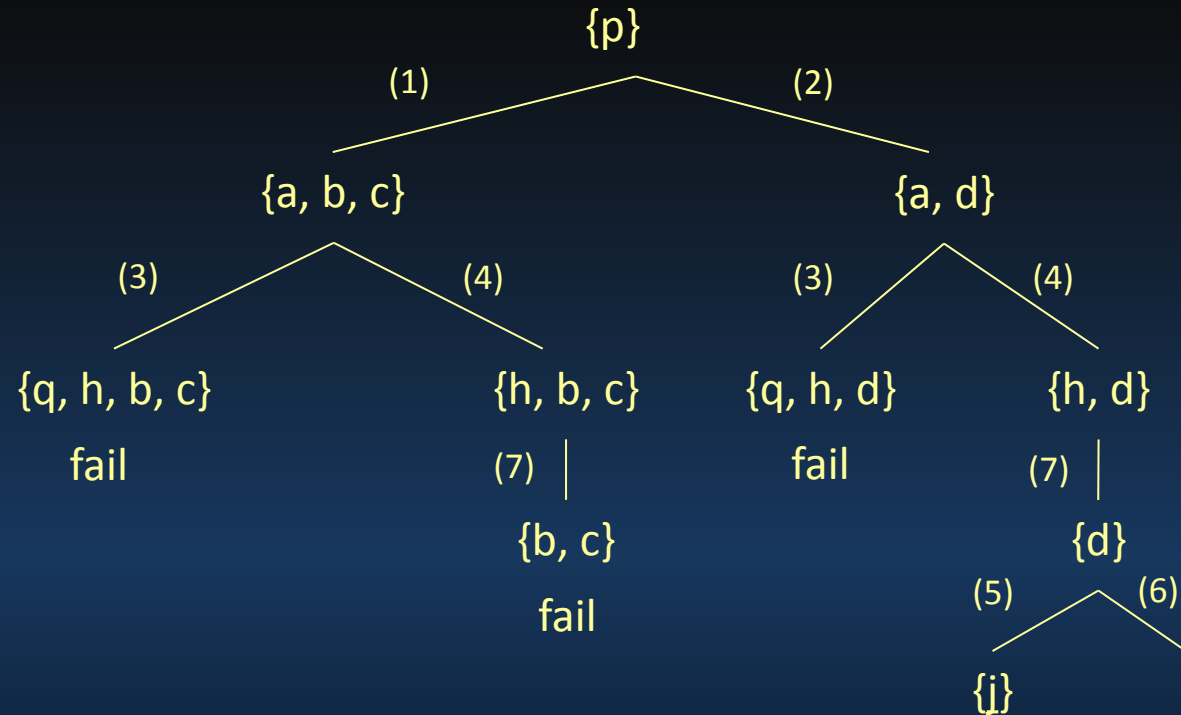
(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



$?- p.$

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p \text{ :- } a, b, c.$

(2)  $p \text{ :- } a, d.$

(3)  $a \text{ :- } q, h.$

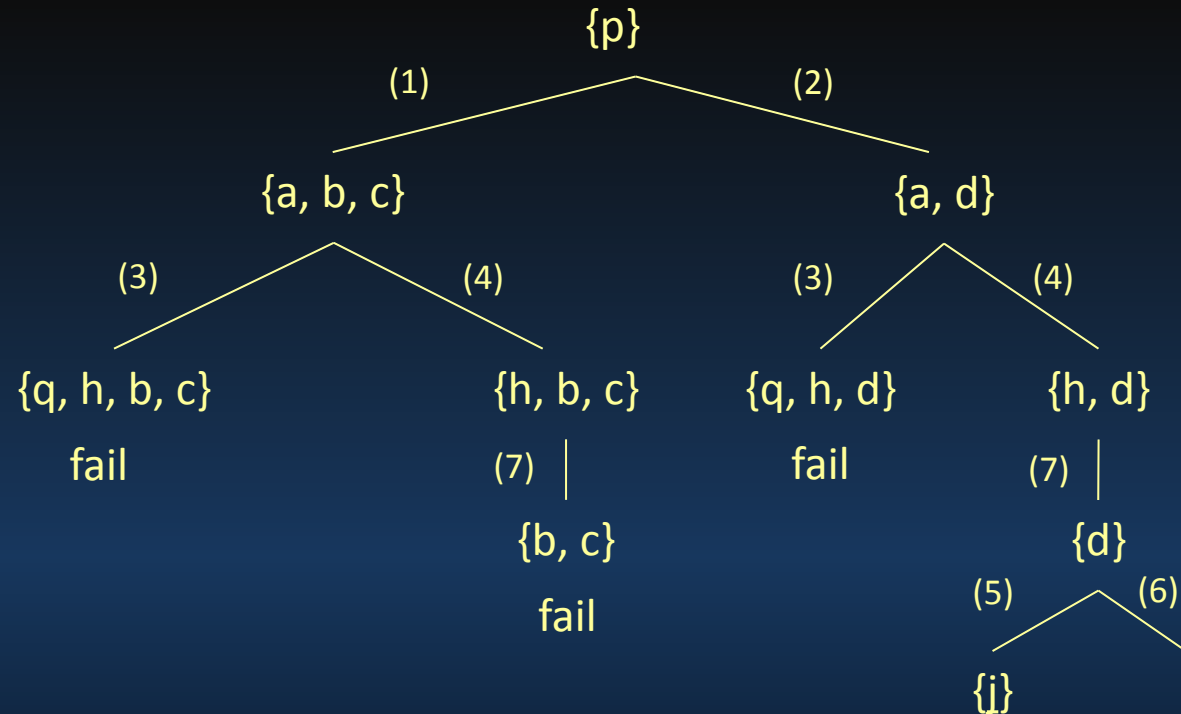
(4)  $a \text{ :- } h.$

(5)  $d \text{ :- } j.$

(6)  $d \text{ :- } i.$

(7)  $h.$

(8)  $j.$



?- p.

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

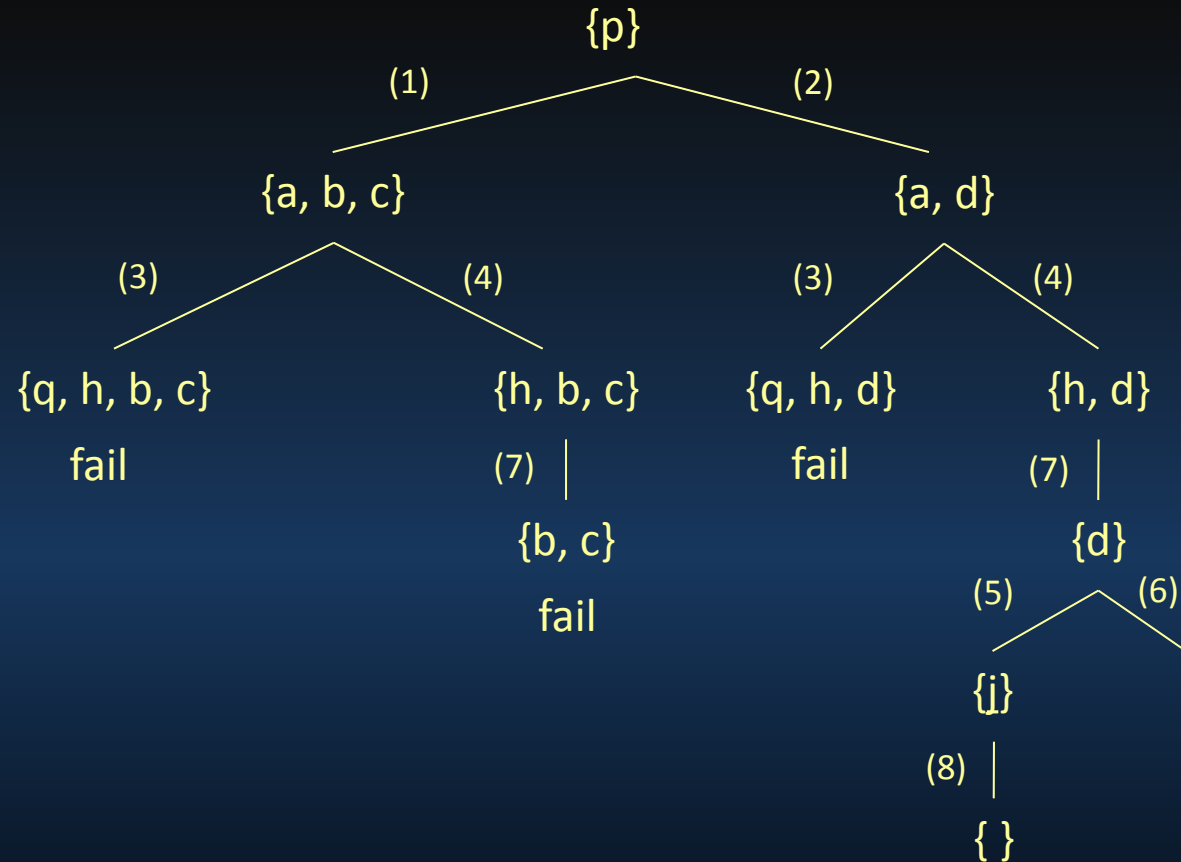
(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



?- p.

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

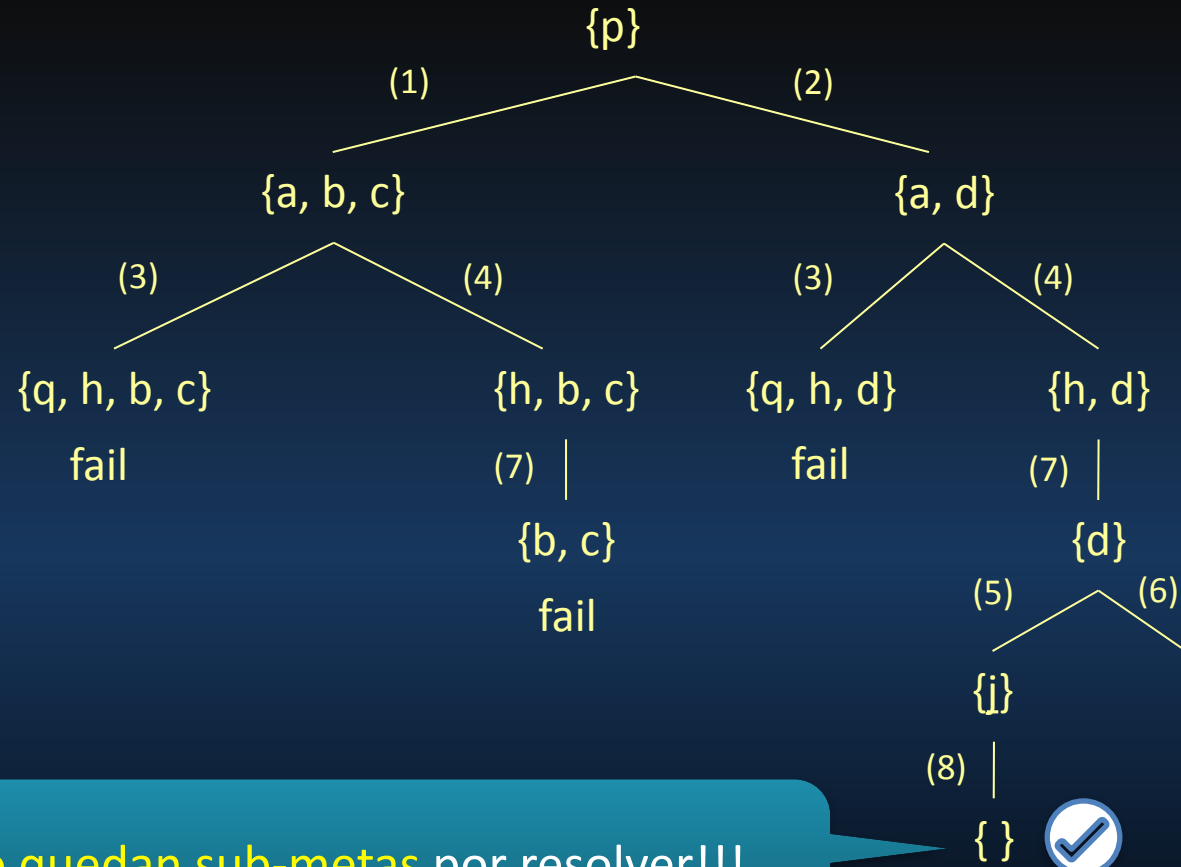
(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



No quedan sub-metas por resolver!!!

?- p.

# Ejemplo – Resolución de Consultas

## Árbol de Derivación

(1)  $p :- a, b, c.$

(2)  $p :- a, d.$

(3)  $a :- q, h.$

(4)  $a :- h.$

(5)  $d :- j.$

(6)  $d :- i.$

(7)  $h.$

(8)  $j.$



Respuesta del intérprete: **existe una derivación para p**

?- p.  
true

# *Ejemplo – Resolución de Consultas*

## *Recursión*

$p :- a, b.$

$b :- p.$

$b :- a.$

$a.$

Consideremos el siguiente programa Prolog, en el que algunas cláusulas están definidas de manera tal que:

- Para derivar  $p$  es necesario derivar  $b$
- Para derivar  $b$  es necesario derivar  $p$

# *Ejemplo – Resolución de Consultas*

## *Recursión*

(1)  $p :- a, b.$

(2)  $b :- p.$

(3)  $b :- a.$

(4)  $a.$

Consideremos el siguiente programa Prolog, en el que algunas cláusulas están definidas de manera tal que:

- Para derivar  $p$  es necesario derivar  $b$
- Para derivar  $b$  es necesario derivar  $p$



# Ejemplo – Resolución de Consultas

## Recursión

(1)  $p :- a, b.$

(2)  $b :- p.$

(3)  $b :- a.$

(4)  $a.$

Consulta que queremos  
resolver: **determinar si p es  
derivable** a partir del programa

?- p.

# *Ejemplo – Resolución de Consultas Recursión*

(1)  $p :- a, b.$

$\{p\}$

(2)  $b :- p.$

(3)  $b :- a.$

(4)  $a.$

Construimos el árbol de derivación  
para la consulta

$?- p.$

# *Ejemplo – Resolución de Consultas*

## *Recursión*

(1)  $p :- a, b.$

$\{p\}$

(2)  $b :- p.$

(3)  $b :- a.$

(4)  $a.$

$?- p.$

# *Ejemplo – Resolución de Consultas*

## *Recursión*

(1)  $p :- a, b.$

$\{p\}$

(2)  $b :- p.$

(3)  $b :- a.$

(4)  $a.$

$?- p.$

# Ejemplo – Resolución de Consultas

## Recursión

(1)  $p :- a, b.$

(2)  $b :- p.$

(3)  $b :- a.$

(4)  $a.$

$\{p\}$   
|  
(1)  
|  
 $\{a, b\}$

$?- p.$

# *Ejemplo – Resolución de Consultas*

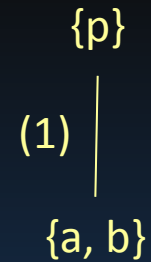
## *Recursión*

(1)  $p :- a, b.$

(2)  $b :- p.$

(3)  $b :- a.$

(4)  $a.$



$?- p.$

# *Ejemplo – Resolución de Consultas*

## *Recursión*

(1)  $p :- a, b.$

(2)  $b :- p.$

(3)  $b :- a.$

(4)  $a.$

$\{p\}$   
(1) |  
 $\{\underline{a}, b\}$

$?- p.$

# Ejemplo – Resolución de Consultas

## Recursión

(1)  $p :- a, b.$

(2)  $b :- p.$

(3)  $b :- a.$

(4)  $a.$

$\{p\}$   
(1) |  
 $\{\underline{a}, b\}$

?-  $p.$



# Ejemplo – Resolución de Consultas

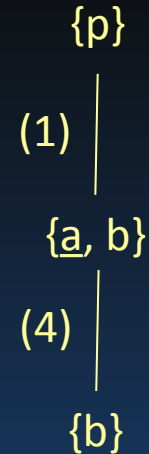
## Recursión

(1)  $p :- a, b.$

(2)  $b :- p.$

(3)  $b :- a.$

(4)  $a.$



$?- p.$

# *Ejemplo – Resolución de Consultas*

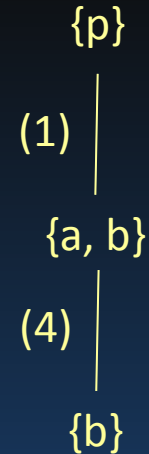
## *Recursión*

(1)  $p :- a, b.$

(2)  $b :- p.$

(3)  $b :- a.$

(4)  $a.$



$?- p.$

# Ejemplo – Resolución de Consultas

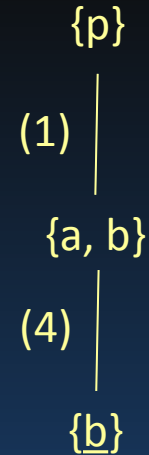
## Recursión

(1)  $p :- a, b.$

(2)  $b :- p.$

(3)  $b :- a.$

(4)  $a.$



$?- p.$

# Ejemplo – Resolución de Consultas

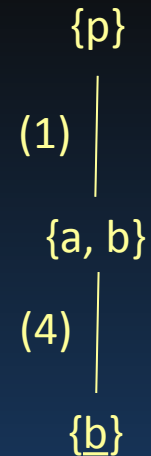
## Recursión

(1)  $p :- a, b.$

(2)  $b :- p.$

(3)  $b :- a.$

(4)  $a.$



?- p.

# Ejemplo – Resolución de Consultas

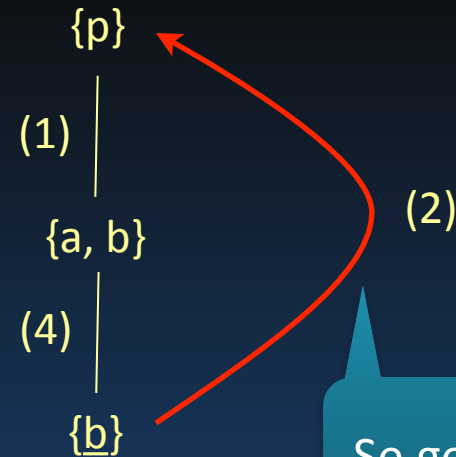
## Recursión

(1)  $p :- a, b.$

(2)  $b :- p.$

(3)  $b :- a.$

(4)  $a.$



Se genera un **ciclo!!!**

?- p.

# Ejemplo – Resolución de Consultas

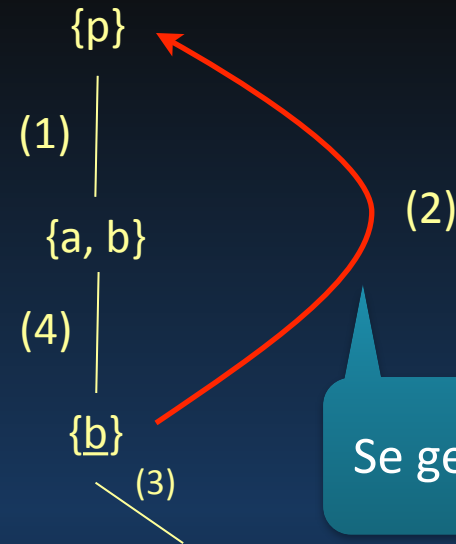
## Recursión

(1)  $p :- a, b.$

(2)  $b :- p.$

(3)  $b :- a.$  ←

(4)  $a.$



Se genera un **ciclo!!!**

?- p.

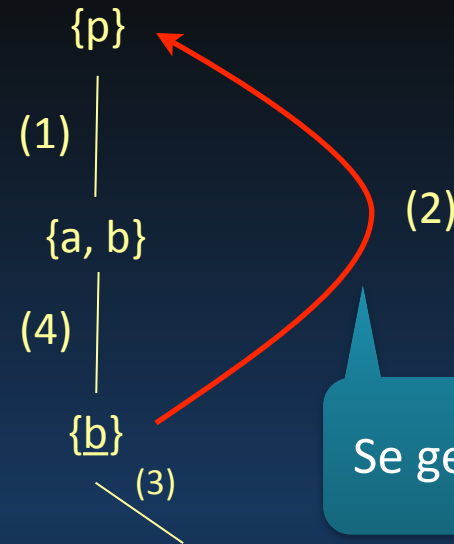
# Ejemplo – Resolución de Consultas Recursión

(1)  $p :- a, b.$

(2)  $b :- p.$

(3)  $b :- a.$

(4)  $a.$



Se genera un **ciclo!!!**

El intérprete selecciona la **regla (2)** por lo que **no hallará** una solución a la consulta **p**, sino que se quedará **ciclando** en la **resolución de las metas** correspondientes a **p** y **b**.

En cambio, si el intérprete seleccionara la **regla (3)**, la consulta sería resuelta en forma **exitosa**.

?- p.

# Ejemplo – Resolución de Consultas

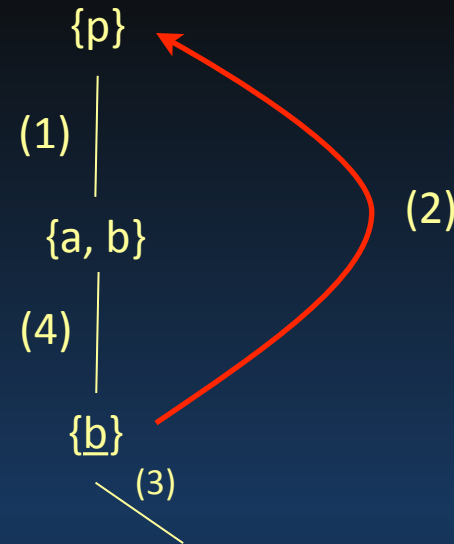
## Recursión

(1)  $p :- a, b.$

(2)  $b :- p.$

(3)  $b :- a.$

(4)  $a.$



**RECORDAR:** El intérprete de **SWI-Prolog** realiza la **construcción** del árbol de derivación **en profundidad**, seleccionando las **sub-metas de izquierda a derecha** y recorriendo las **cláusulas** del programa **de arriba hacia abajo**.

El programador debe tener en cuenta el **orden** en que especifica las **cláusulas** de un programa PROLOG para mejorar su **performance** y, especialmente, para **garantizar su correctitud** al no comprometer su terminación.