

## Modes ( $M$ ):

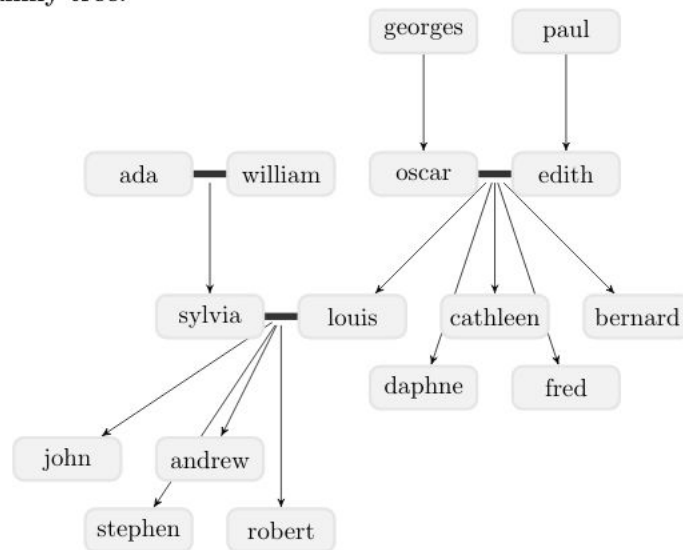
```
modeh(1,parent_of(+person,-person))  
modeh(1,grandfather_of(+person,-person))  
modeh(1,grandparent_of(+person,-person))
```

```
modeb(*,father_of(+person,-person))  
modeb(*,mother_of(+person,-person))  
modeb(*,parent_of(+person,-person))
```

## Examples:

```
parent_of(louis, stephen).  
parent_of(sylvia, robert).  
grandfather_of(william, robert).  
grandfather_of(georges, louis).  
grandparent_of(william, andrew).
```

## Family tree:



## Determinations:

parent_of:	grandparent_of:	grandfather_of:
<ul style="list-style-type: none"><li>• mother_of</li><li>• father_of</li></ul>	<ul style="list-style-type: none"><li>• parent_of</li><li>• mother_of</li><li>• father_of</li></ul>	<ul style="list-style-type: none"><li>• parent_of</li><li>• mother_of</li><li>• father_of</li></ul>

Parameters:  $i = 2$   $c = 2$

**Knowledge:** /

**Example:** `parent_of(louis, stephen).`

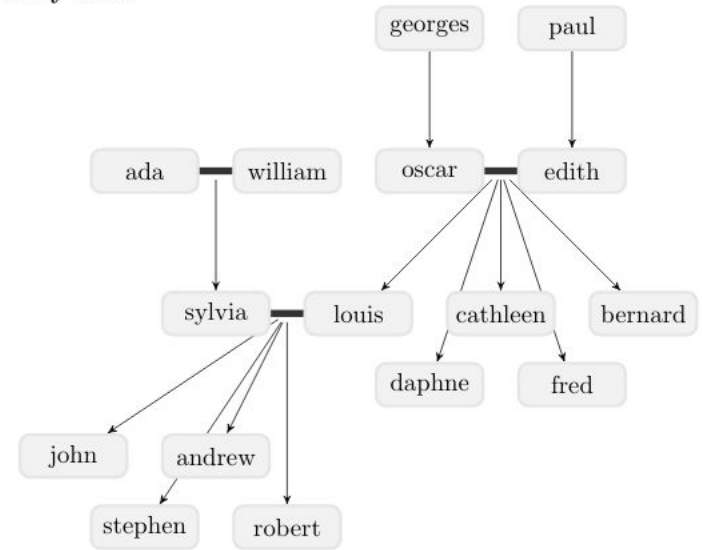
**Most specific clause:**

A `parent_of B` :-  
    A `father_of B`,  
    A `father_of C`,  
    A `father_of D`,  
    A `father_of E`.

$\perp_i$  **after reduction:**

A `parent_of B` :- A `father_of B`.

**Family tree:**



**Determinations:**

parent_of:	grandparent_of:	grandfather_of:
<ul style="list-style-type: none"><li>• mother_of</li><li>• father_of</li></ul>	<ul style="list-style-type: none"><li>• parent_of</li><li>• mother_of</li><li>• father_of</li></ul>	<ul style="list-style-type: none"><li>• parent_of</li><li>• mother_of</li><li>• father_of</li></ul>

**Parameters:**  $i = 2$   $c = 2$

**Knowledge:** /

**Example:** `parent_of(louis, stephen).`

$\perp_i$  **after reduction:**

`A parent_of B :- A father_of B.`

**Hypotheses:**

C	[ f, p, n, h]
A parent_of B	[-1, 8, 7, 1]
A parent_of B :- A father_of B.	[ 2, 4, 1, 0]
A parent_of B :- A father_of C.	[-6, 4, 4, 1]

**Result of search:**

`A parent_of B :- A father_of B.`

**Numbers:**

f = Number of positive examples covered -  
Number of negative examples covered -  
Number of literals in body of clause -  
Optimistic estimate of literals needed  
p = Number of positive examples covered  
n = Number of negative examples covered  
h = Optimistic estimate of literals needed

**Determinations:**

parent_of:	grandparent_of:	grandfather_of:
• mother_of	• parent_of	• parent_of
• father_of	• mother_of	• mother_of
	• father_of	• father_of

**Parameters:**  $i = 2$   $c = 2$

**Knowledge:**  $\text{parent\_of}(A, B) \text{ :- father\_of}(A, B).$

**Example:**  $\text{parent\_of}(\textit{sylvia}, \textit{robert}).$

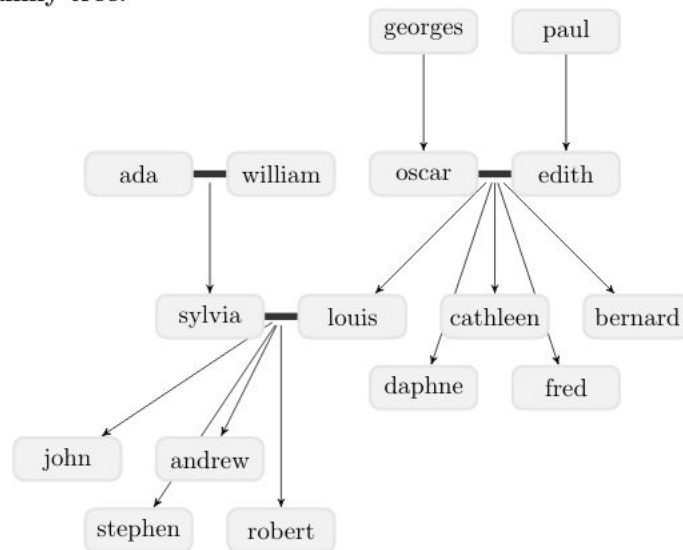
**Most specific clause:**

$A \text{ parent\_of } B \text{ :- } A \text{ mother\_of } B,$   
 $A \text{ mother\_of } C,$   
 $A \text{ mother\_of } D,$   
 $A \text{ mother\_of } E.$

$\perp_i$  **after reduction:**

$A \text{ parent\_of } B \text{ :- } A \text{ mother\_of } B.$

**Family tree:**



**Determinations:**

parent_of:	grandparent_of:	grandfather_of:
<ul style="list-style-type: none"><li>• mother_of</li><li>• father_of</li></ul>	<ul style="list-style-type: none"><li>• parent_of</li><li>• mother_of</li><li>• father_of</li></ul>	<ul style="list-style-type: none"><li>• parent_of</li><li>• mother_of</li><li>• father_of</li></ul>

**Parameters:**  $i = 2$   $c = 2$

## Knowledge:

parent\_of(A, B) :- father\_of(A, B).  
parent\_of(A, B) :- mother\_of(A, B).

**Example:** grandfather\_of(*william*, *robert*).

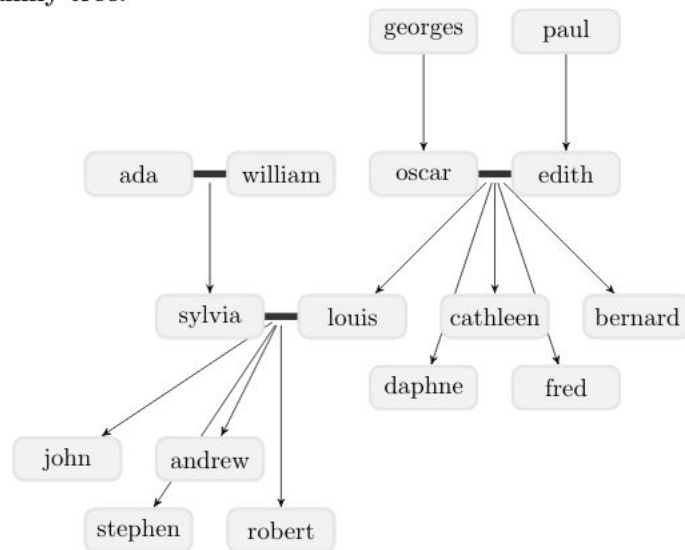
## Most specific clause:

A grandfather\_of B :-  
A parent\_of C,  
A father\_of C,  
C parent\_of B,  
C parent\_of D,  
C parent\_of E,  
C parent\_of F,  
C mother\_of B,  
C mother\_of D,  
C mother\_of E,  
C mother\_of F,

## $\perp_i$ after reduction:

A grandfather\_of B :-  
A parent\_of C,  
A father\_of C,  
C parent\_of B,  
C mother\_of B,

## Family tree:



## Determinations:

parent_of:	grandparent_of:	grandfather_of:
<ul style="list-style-type: none"><li>• mother_of</li><li>• father_of</li></ul>	<ul style="list-style-type: none"><li>• parent_of</li><li>• mother_of</li><li>• father_of</li></ul>	<ul style="list-style-type: none"><li>• parent_of</li><li>• mother_of</li><li>• father_of</li></ul>

Parameters:  $i = 2$   $c = 2$

## Knowledge:

parent\_of(A, B) :- father\_of(A, B).  
parent\_of(A, B) :- mother\_of(A, B).  
grandfather\_of(A, B) :- parent\_of(A, C), parent\_of(C, B).

**Example:** grandparent\_of(*william*, *andrew*).

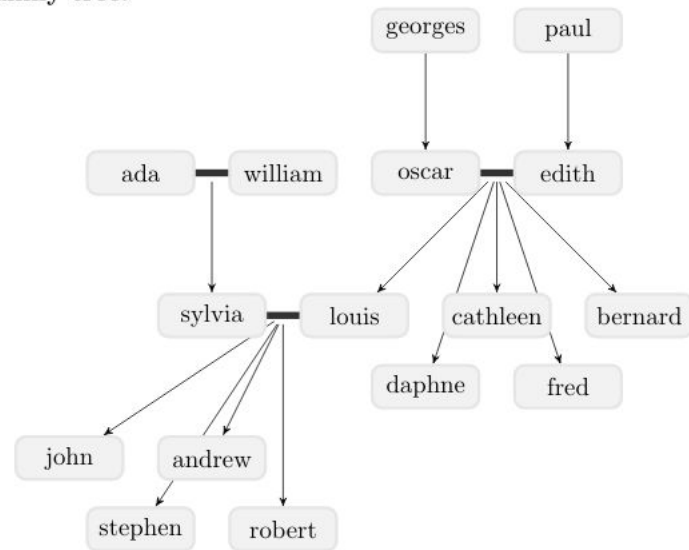
## Most specific clause:

A grandparent\_of B :-  
A parent\_of C,  
C parent\_of B,  
C parent\_of D,  
C parent\_of E,  
C parent\_of F,  
C mother\_of B,  
C mother\_of D,  
C mother\_of E,  
C mother\_of F,

$\perp_i$  after reduction:

A grandparent\_of B :-  
A parent\_of C,  
C parent\_of B,  
C mother\_of B,

## Family tree:



## Determinations:

parent_of:	grandparent_of:	grandfather_of:
• mother_of	• parent_of	• parent_of
• father_of	• mother_of	• mother_of
	• father_of	• father_of

Parameters:  $i = 2$   $c = 2$

**Knowledge:**

parent\_of(A, B) :- father\_of(A, B).  
parent\_of(A, B) :- mother\_of(A, B).  
grandfather\_of(A, B) :- parent\_of(A, C), parent\_of(C, B).

**Example:** grandparent\_of(*william, andrew*).

**$\perp_i$  after reduction:**

A grandparent\_of B :-  
    A parent\_of C,  
    C parent\_of B,  
    C mother\_of B,

**Hypotheses:**

C	[ f, p, n, h]
A grandparent_of B.	[-2, 4, 3, 2]
A grandparent_of B :- A parent_of C.	[-2, 4, 3, 1]
A grandparent_of B :- A parent_of C, C parent_of B.	[ 0, 4, 1, 0]
A grandparent_of B :- A parent_of C, C parent_of D.	[-3, 4, 3, 1]
A grandparent_of B :- A parent_of C, C mother_of B.	[ 0, 4, 1, 0]
A grandparent_of B :- A parent_of C, C mother_of D.	[-1, 4, 1, 1]

**Result of search:**

A grandparent\_of B :- A parent\_of C, C parent\_of B.

**Numbers:**

f = Number of positive examples covered -  
    Number of negative examples covered -  
    Number of literals in body of clause -  
    Optimistic estimate of literals needed  
p = Number of positive examples covered  
n = Number of negative examples covered  
h = Optimistic estimate of literals needed

**Determinations:**

parent_of:	grandparent_of:	grandfather_of:
• mother_of	• parent_of	• parent_of
• father_of	• mother_of	• mother_of
	• father_of	• father_of

Parameters:  $i = 2$   $c = 2$

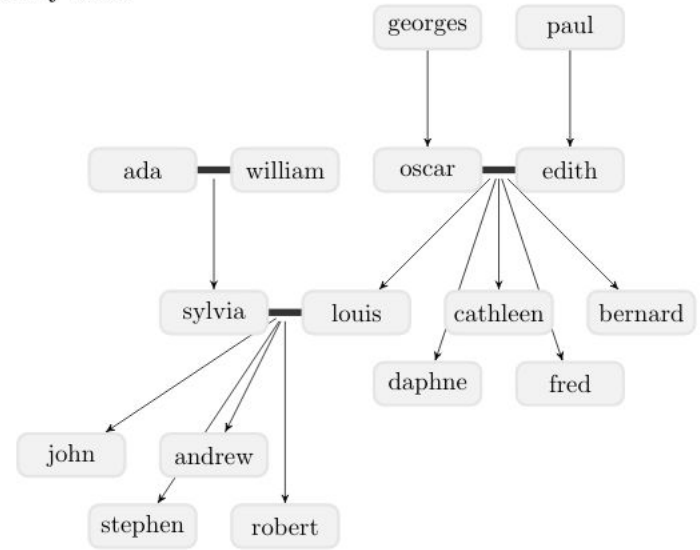
## Examples:

parent\_of(*louis*, *stephen*).  
parent\_of(*sylvia*, *robert*).  
grandfather\_of(*william*, *robert*).  
grandfather\_of(*georges*, *louis*).  
grandparent\_of(*william*, *andrew*).

## Knowledge:

parent\_of(A, B) :- father\_of(A, B).  
parent\_of(A, B) :- mother\_of(A, B).  
grandfather\_of(A, B) :- parent\_of(A, C), parent\_of(C, B).  
grandparent\_of(A, B) :- parent\_of(A, C), parent\_of(C, B).

## Family tree:



## Determinations:

parent_of:	grandparent_of:	grandfather_of:
<ul style="list-style-type: none"><li>• mother_of</li><li>• father_of</li></ul>	<ul style="list-style-type: none"><li>• parent_of</li><li>• mother_of</li><li>• father_of</li></ul>	<ul style="list-style-type: none"><li>• parent_of</li><li>• mother_of</li><li>• father_of</li></ul>

Parameters:  $i = 2$   $c = 2$