



# *Artificial intelligence*

---

Rajorshi Bhattacharya, Lecturer  
(CSE), EU

- 
- 1. <https://youtu.be/tFYngFwDuJQ>
  - 2. <https://youtu.be/Z0vTGofo2n8>
  - 3. [https://youtu.be/Ggle\\_tNpLjk](https://youtu.be/Ggle_tNpLjk)

# Predicate Logic in AI

1. Marcus was a man.
2. Marcus was a Pompeian.
3. All Pompeian were Romans.
4. Caesar was a ruler.
5. All Romans were either loyal to Caesar or hated him.
6. Everyone is loyal to someone.
7. People only try to assassinate rulers they are not loyal to.
8. Marcus tried to assassinate Caesar.

**Sentences as  
Well formed  
Formulas**

# Sentences to Well formed Formulas in Predicate Logic

1. Marcus was a man.
2. Marcus was a Pompeian.
3. All Pompeian were Romans.
4. Caesar was a ruler.
5. All Romans were either loyal to Caesar or hated him.
6. Everyone is loyal to someone.
7. People only try to assassinate rulers they are not loyal to.
8. Marcus tried to assassinate Caesar.
9. All men are people

# Sentences to Well formed Formulas in Predicate Logic

- |   |  |
|---|--|
| 1. Marcus was a man.  | 1. <i>man(Marcus)</i>  |
| 2. Marcus was a Pompeian.                                       | 2. <i>Pompeian(Marcus)</i>   |
| 3. All Pompeian were Romans.                                    | 3. $\forall x : Pompeian(x) \rightarrow Roman(x)$  |
| 4. Caesar was a ruler.  | 4. <i>ruler(Caesar)</i>  |
| 5. All Romans were either loyal to Caesar or hated him.         | 5. $\forall x : Roman(x) \rightarrow \text{loyalto}(x, Caesar)$<br>$\vee \text{hate}(x, Caesar)$   |
| 6. Everyone is loyal to someone.                                | 6. $\forall x : \exists y : \text{loyalto}(x, y)$  |
| 7. People only try to assassinate rulers they are not loyal to. | 7. $\forall x : \forall y : \text{person}(x) \wedge \text{ruler}(y) \wedge \text{tryassassinate}(x, y)$<br>$\rightarrow \neg \text{loyalto}(x, y)$ |
| 8. Marcus tried to assassinate Caesar.                          | 8. <i>tryassassinate(Marcus, Caesar)</i>   |
| 9. All men are people   | 9. $\forall x : \text{man}(x) \rightarrow \text{person}(x)$  |





## Resolution using Predicate Logic in Artificial Intelligence

1.  $\text{man}(\text{Marcus})$
2.  $\text{Pompeian}(\text{Marcus})$
3.  $\forall x : \text{Pompeian}(x) \rightarrow \text{Roman}(x)$
4.  $\text{ruler}(\text{Caesar})$
5.  $\forall x : \text{Roman}(x) \rightarrow \text{loyalto}(x, \text{Caesar})$   
 $\quad \vee \text{hate}(x, \text{Caesar})$
6.  $\forall x : \exists y : \text{loyalto}(x, y)$
7.  $\forall x : \forall y : \text{person}(x) \wedge \text{ruler}(y) \wedge \text{tryassassinate}(x, y)$   
 $\rightarrow \neg \text{loyalto}(x, y)$
8.  $\text{tryassassinate}(\text{Marcus}, \text{Caesar})$
9.  $\forall x : \text{man}(x) \rightarrow \text{person}(x)$

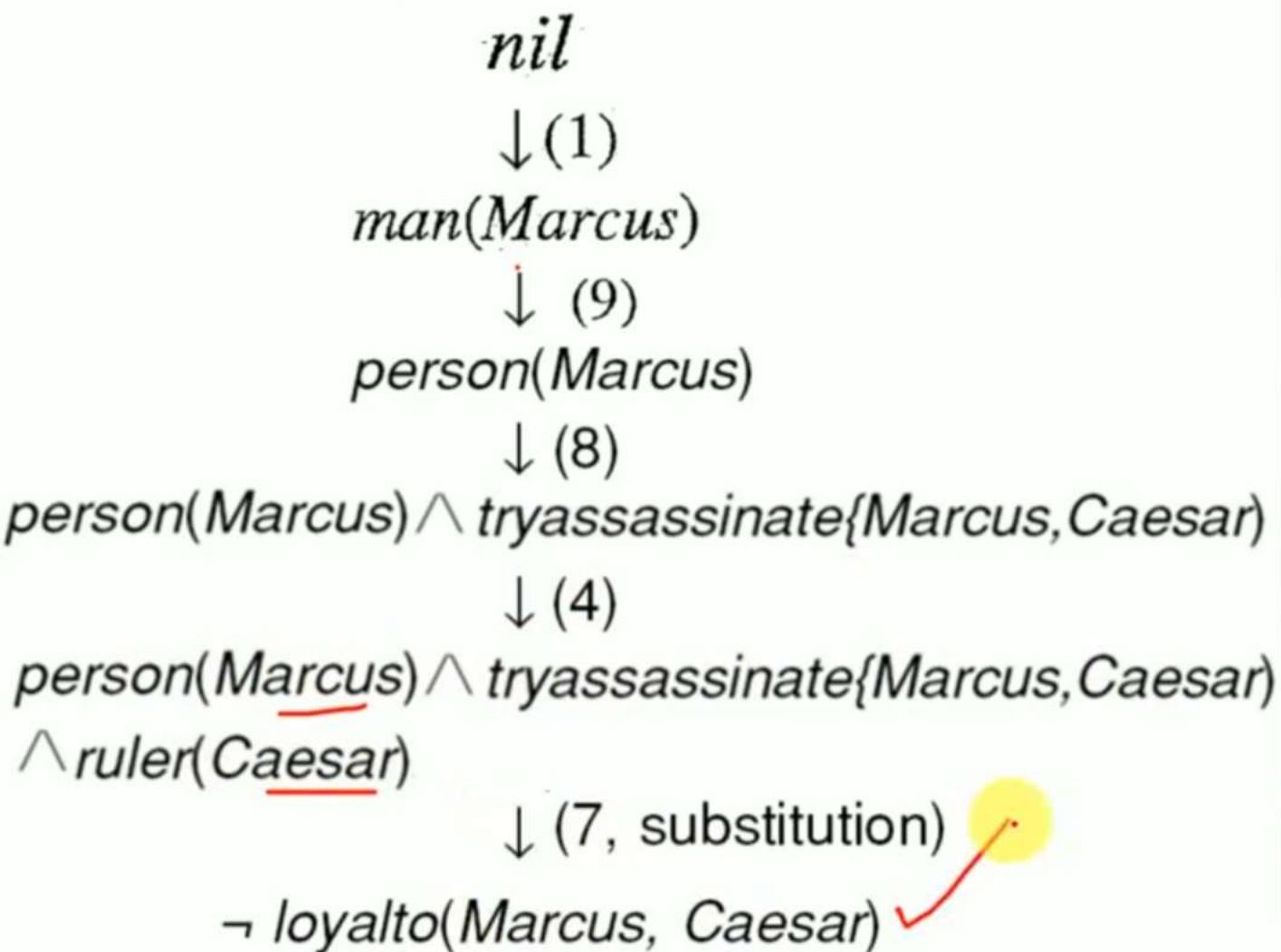
# Resolution using Predicate Logic in Artificial Intelligence

1. Was Marcus Loyal to Caesar?

1.  $\text{man}(\text{Marcus})$
2.  $\text{Pompeian}(\text{Marcus})$
3.  $\forall x : \text{Pompeian}(x) \rightarrow \text{Roman}(x)$
4.  $\text{ruler}(\text{Caesar})$
5.  $\forall x : \text{Roman}(x) \rightarrow \text{loyalto}(x, \text{Caesar})$   
 $\quad \vee \text{hate}(x, \text{Caesar})$
6.  $\forall x : \exists y : \text{loyalto}(x, y)$
7.  $\forall x : \forall y : \text{person}(x) \wedge \text{ruler}(y)$   
 $\quad \wedge \text{tryassassinate}(x, y) \rightarrow \neg \text{loyalto}(x, y)$
8.  $\text{tryassassinate}(\text{Marcus}, \text{Caesar})$
9.  $\forall x : \text{man}(x) \rightarrow \text{person}(x)$

# Resolution using Predicate Logic in Artificial Intelligence

1. Was Marcus Loyal to Caesar?



1. *man(Marcus)*
2. *Pompeian(Marcus)*
3.  $\forall x : \text{Pompeian}(x) \rightarrow \text{Roman}(x)$
4. *ruler(Caesar)*
5.  $\forall x : \text{Roman}(x) \rightarrow \text{loyalto}(x, \text{Caesar})$   
 $\vee \text{hate}(x, \text{Caesar})$
6.  $\forall x : \exists y : \text{loyalto}(x, y)$
7.  $\forall x : \forall y : \text{person}(x) \wedge \text{ruler}(y)$   
 $\wedge \text{tryassassinate}(x, y) \rightarrow \neg \text{loyalto}(x, y)$
8. *tryassassinate(Marcus, Caesar)*
9.  *$\forall x : \text{man}(x) \rightarrow \text{person}(x)$*

# Resolution using Predicate Logic in Artificial Intelligence

2. Was Marcus hates Caesar?

1.  $\text{man}(\text{Marcus})$
2.  $\text{Pompeian}(\text{Marcus})$
3.  $\forall x : \text{Pompeian}(x) \rightarrow \text{Roman}(x)$
4.  $\text{ruler}(\text{Caesar})$
5.  $\forall x : \text{Roman}(x) \rightarrow \text{loyalto}(x, \text{Caesar})$   
 $\quad \vee \text{hate}(x, \text{Caesar})$
6.  $\forall x : \exists y : \text{loyalto}(x, y)$
7.  $\forall x : \forall y : \text{person}(x) \wedge \text{ruler}(y)$   
 $\quad \wedge \text{tryassassinate}(x, y) \rightarrow \neg \text{loyalto}(x, y)$
8.  $\text{tryassassinate}(\text{Marcus}, \text{Caesar})$
9.  $\forall x : \text{man}(x) \rightarrow \text{person}(x)$

# Resolution using Predicate Logic in Artificial Intelligence

2. Was Marcus hates Caesar?

$\neg \text{loyalto}(\text{Marcus}, \text{Caesar})$

↓ (2)

*Pompeian(Marcus)*

$\neg \text{loyalto}(\text{Marcus}, \text{Caesar})$

↓ (3)

*Roman(Marcus)*

$\neg \text{loyalto}(\text{Marcus}, \text{Caesar})$

↓ (5)

*hate(Marcus, Caesar)*

1.  $\text{man}(\text{Marcus})$
2.  $\text{Pompeian}(\text{Marcus})$
3.  $\forall x : \text{Pompeian}(x) \rightarrow \text{Roman}(x)$
4.  $\text{ruler}(\text{Caesar})$
5.  $\forall x : \text{Roman}(x) \rightarrow \text{loyalto}(x, \text{Caesar})$   
     $\vee \text{hate}(x, \text{Caesar})$
6.  $\forall x : \exists y : \text{loyalto}(x, y)$
7.  $\forall x : \forall y : \text{person}(x) \wedge \text{ruler}(y)$   
     $\wedge \text{tryassassinate}(x, y) \rightarrow \neg \text{loyalto}(x, y)$
8.  $\text{tryassassinate}(\text{Marcus}, \text{Caesar})$
9.  $\forall x : \text{man}(x) \rightarrow \text{person}(x)$



# Resolution using Predicate Logic in Artificial Intelligence

1. Marcus was a man.
2. Marcus was a Pompeian.
3. Marcus was born in 40 A.D.
4. All men are mortal.
5. All Pompeians died when the volcano erupted in 79 A.D.
6. No mortal lives longer than 150 years.
7. It is now 1991.
8. Alive means not dead.
9. If someone dies, then he is dead at all later times.

**Was Marcus Dead?**

## Resolution using Predicate Logic in Artificial Intelligence

1.  $\text{man}(\text{Marcus})$
2.  $\text{Pompeian}(\text{Marcus})$
3.  $\text{born}(\text{Marcus}, 40)$
4.  $\forall x : \text{man}(x) \rightarrow \text{mortal}(x)$
5.  $\forall : \text{Pompeian}(x) \rightarrow \text{died}(x, 79)$   
 $\text{erupted}(\text{volcano}, 79)$
6.  $\forall_x : \forall t_1 : \forall t_2 : \text{mortal}(x) \wedge \text{born}(x, t_1) \wedge \text{gt}(t_2 - t_1, 150) \rightarrow \text{dead}(x, t_2)$
7.  $\text{now} = 1991$
8.  $\forall x : \forall t : [\text{alive}(x, t) \rightarrow \neg \text{dead}(x, t)] \wedge [\neg \text{dead}(x, t) \rightarrow \text{alive}(x, t)]$
9.  $\forall x : \forall t_1 : \forall t_2 : \text{died}(x, t_1) \wedge \text{gt}(t_2, t_1) \rightarrow \text{dead}(x, t_2)$

# Resolution using Predicate Logic in Artificial Intelligence

1.  $man(Marcus)$
  2.  $Pompeian(Marcus)$
  3.  $born(Marcus, 40)$
  4.  $\forall x : man(x) \rightarrow mortal(x)$
  5.  $\forall : Pompeian(x) \rightarrow died(x, 79)$
  6.  $erupted(yolcano, 79)$
  7.  $\forall_x : \forall t_1 : \forall t_2 : mortal(x) \wedge born(x, t_1) \wedge gt(t_2 - t_1, 150) \rightarrow dead(x, t_2)$
  8.  $now = 1991$
  9.  $\forall x : \forall t : [alive(x, t) \rightarrow \neg dead(x, t)] \wedge [\neg dead(x, t) \rightarrow alive(x, t)]$
  10.  $\forall x : \forall t_1 : \forall t_2 : died(x, t_1) \wedge gt(t_2, t_1) \rightarrow dead(x, t_2)$
- nil*  
 $\downarrow$  (compute  $gt$ )  
 $gt(1991, 79)$   
 $\downarrow$  (8, substitute equals)  
 $gt(now, 79)$   
 $\downarrow$  (2)  
 $Pompeian(Marcus) \wedge gt(now, 79)$   
 $\downarrow$  (5, substitution)  
 $died(Marcus, t_1) \wedge gt(now, t_1)$   
 $\downarrow$  (10, substitution)  
 $dead(Marcus, now)$   
 $\downarrow$  (9, substitution)  
 $\neg alive(Marcus, now)$

