# CSC242: Introduction to Assignment Project Exam Help Artificial/Lintelligence

WeChat: cstutorcs

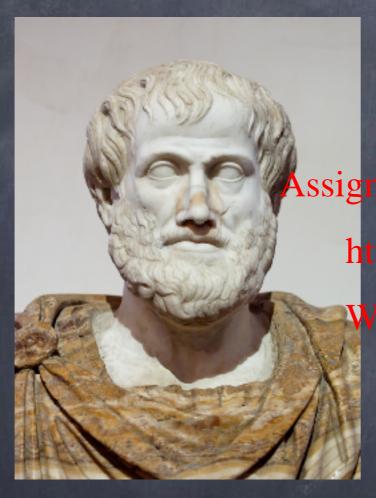
Lecture 2.3

Please put away all electronic devices

#### Boolean CSP

- All variables must be Booleans
  - Domains sall mentrive, tals elep
- Constraints: Identify possible combinations of the boolean variables

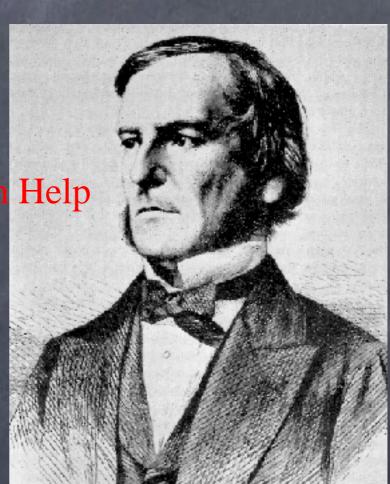
## Propositional Logic



Assignment Project Exam

https://tutorcs.com

WeChat: cstutorcs



Aristole (384BC - 332BC) George Boole (1815-1864)

### Propositional Logic

- Propositions: things that are true or false

  Assignment Project Exam Help
- Connectives: nombine opropositions into larger propositions stutores
- Sentences: statements about the world (can be true or false)
  - Boolean functions of Boolean variables

#### Truth Table

 $L_{1,1} \wedge (W_{1,2} \vee W_{2,1})$ 

$L_{1,1}$	$W_{1,2}$	$W_{2,1}$	$W_{1,2} \lor W_{2,1}$ Exam Help	$L_{1,1} \wedge (W_{1,2} \vee W_{2,1})$
F	F	nttps://tutorcs.co	m F	F
F	F	WeChat: cstutoro	T T	F
F	T	F	T	F
F	T	T	T	F
T	F	F	F	F
T	F	T	T	T
T	T	F	T	THE REAL PROPERTY.
T	T	T	T	T

# Propositional Logic

- Possible worlds
- Models Assignment Project Exam Help
- Satisfiabilityhttps://tutorcs.com
- Unsatisfiable WeChat: cstutorcs

# Background Knowledge

$$B_{1,1} \Leftrightarrow P_{1,2} \vee P_{2,1}$$

$$B_{1,2} \Leftrightarrow P_{1,1} \vee P_{2,2} \vee P_{3,1}$$

$$B_{2,2} \Leftrightarrow P_{1,2} \vee P_{2,3} \text{ in } P_{3,2} \text{ is } OK_{1,1} \Leftrightarrow \neg (P_{1,1} \vee W_{1,1})$$

$$\dots \qquad \qquad \text{https://tutores.com } OK_{1,2} \Leftrightarrow \neg (P_{1,2} \vee W_{1,2})$$

$$S_{1,1} \Leftrightarrow W_{1,2} \vee W_{2,1} \text{ we Chat: estutores } OK_{2,1} \Leftrightarrow \neg (P_{2,1} \vee W_{2,1})$$

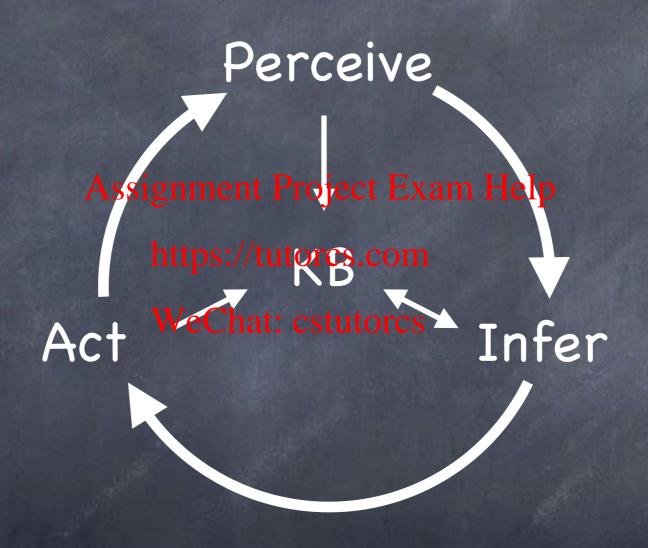
$$S_{1,2} \Leftrightarrow W_{1,1} \vee W_{2,2} \vee W_{3,1} \qquad \dots$$

$$S_{2,2} \Leftrightarrow W_{1,2} \vee W_{2,3} \vee W_{3,2} \vee W_{2,1}$$

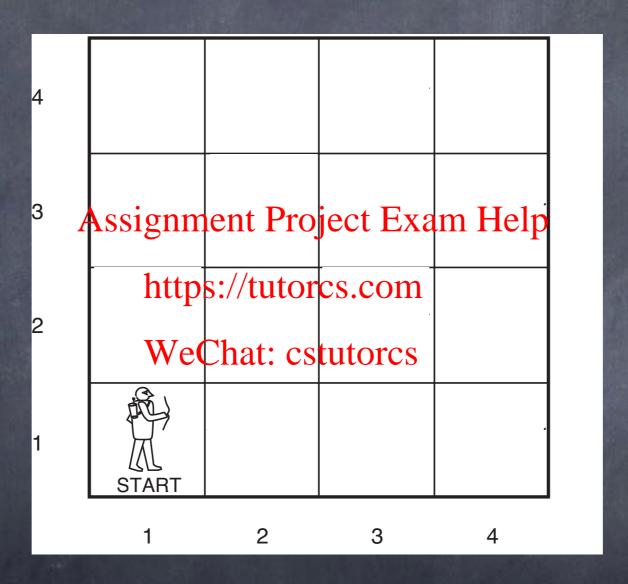
$$\dots$$

 $W_{1,1} \lor W_{1,2} \lor ... \lor W_{3,4} \lor W_{4,4}$  $\neg (W_{1,1} \land W_{1,2}), \neg (W_{1,1} \land W_{1,3}), ..., \neg (W_{3,4} \land W_{4,4})$ 

# Knowledge-Based Agents



## Perception



$$\neg B_{1,1}, \ \neg S_{1,1}$$

#### Inference

- Given what I know...
  - What shoulded pagest Exam Helpee AIMA 7.7

https://tutorcs.com

#### Inference

Given what I know... Is there no pit in room [2,1]?

Assignment Project Exam Help

https://tutorcs.com

 $R_1$ :  $\neg P_{1,1}$ 

 $R_2$ :  $B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$  WeChat: cstutorcs  $\neg P_{2,1}$ ?

 $\neg B_{1,1}$ 

#### Entailment

- $\alpha$  entails  $\beta$  :  $\alpha \models \beta$ 
  - Every madelneft rajist elsa Helpmodel of  $\beta$
  - Whenever  $\alpha$  is true, so is  $\beta$
  - $\beta$  is true in every world consistent with  $\alpha$
  - $Models(\alpha) \subseteq Models(\beta)$
  - $\beta$  logically follows from  $\alpha$

- Given knowledge  $\alpha$  and query  $\beta$ 
  - For everyignossibole: Wonlderw
    - If  $\alpha$  is satisfied by w WeChat: cstutores
      - ullet If eta is not satisfied by w
        - Conclude that  $\alpha \not\models \beta$
  - Conclude that  $\alpha \models \beta$

- Given knowledge  $\alpha$  and query  $\beta$ 
  - For every ispossibile would by
    - If  $\alpha$  is satisfied by w WeChat: cstutores
      - ullet If eta is not satisfied by w
        - Conclude that  $\alpha \nvDash \beta$
  - Conclude that  $\alpha \models \beta$  AIMA Fig. 7.10

#### Possible Worlds

$P_{1,1}$	$P_{1,2}$	$P_{2,1}$	$B_{1,1}$			
F	F	Āssig	nment Pro	ject Exam	Help	
F	F	F h	ttps://tutor	cs.com		
F	F	T V	VeCl <mark>pat: cs</mark>	tutorcs		
•••						
Т	Т	F	T			
T	T	T	F			
T	T	T	T			

#### Knowledge

$P_{1,1}$	$P_{1,2}$	$P_{2,1}$	$B_{1,1}$	$R_1$	$R_2$	$\neg B_{1,1}$	$\neg P_{2,1}$
F	F	Assig	nment Pro	ject Exam	Help	T	
F	F	F h	ttps://tutor	cs.com	F	F	
F	F	T V	VeChat: cs	tutorcs	F	Т	
•••				•••			
Т	Т	F	Т	F	T	F	
Т	Т	T	F	F	F	Т	
T	T	J	T	F	T	F	

Knowledge

$P_{1,1}$	$P_{1,2}$	$P_{2,1}$	$B_{1,1}$	$R_1$	$R_2$	$\neg B_{1,1}$	$\neg P_{2,1}$
F	F	Assig	nment Pro	ject Exam	Help	Т	
	F	<mark>- F h</mark>	ttps://tutor	cs.com	Г	Г	
F	F	T	VeCl <mark>jat: cs</mark>	tutor <mark>e</mark> s—		T	
•••							
T	T	F	T	F	T	F	
T	T	-	F			T	
T	T		T	F	T	F	TALL VI

Knowledge

$P_{1,1}$	$P_{1,2}$	$P_{2,1}$	$B_{1,1}$	$R_1$	$R_2$	$\neg B_{1,1}$	$ egreen P_{2,1} egreen$
F	F	Assig	nment Pro	ject Exam	Help	T	
	F	<del>- F h</del>	ttps://tutor	cs.com	F	F	
F	F	T	VeCl <mark>rat: cs</mark>	tutorcs	F	T	
•••				•••			
T	T	F	T	F	T	F	
T	T	T	F	F	F	T	
T	T	T	T	F	Т	F	

#### Query

	$P_{1,1}$	$P_{1,2}$	$P_{2,1}$	$B_{1,1}$	$R_1$	$R_2$	$\neg B_{1,1}$	$\neg P_{2,1}$
	F	F	Assig	nment Pro	ject Exam	Help	Т	T
	F	F	<del>F h</del>	ttps://tutor	cs.com	F	F	
	F	F	T	VeCl <mark>pat: cs</mark>	tutorcs	F	T	
	•••				•••			
HASS.	T	T	F	T	F	T	F	
	T	T	T	F	F	F	T	
	T	Т	T	T	F	T	F	

#### Entailment

Given what I know... Is there no pit in room [2,1]?

 $R_1: \neg P_{1,1}$ 

 $R_2$ :  $B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$  WeChat: cstutorcs  $\neg P_{2,1}$ ?

 $\neg B_{1,1}$ 

 $KB \vDash \neg P_{2,1}$ 

#### Entailment

#### Given what I know... Is there no pit in room [1,2]?

Assignment Project Exam Help

$$R_1$$
:  $\neg P_{1,1}$  https://tutorcs.com

$$R_2$$
:  $B_{1,1} \Leftrightarrow (P_{1,2} \vee P_2 G)$  at: cstutorcs

$$R_3: B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$$

$$\neg B_{1,1}$$

$$B_{2,1}$$

$$\neg P_{1,2}$$
?

$$KB \vDash \neg P_{1,2}$$

$$KB \not\models P_{2,2}$$

$$KB \not\models \neg P_{2,2}$$

https://tutores.com WeChat: cstutores

- Given knowledge  $\alpha$  and query  $\beta$ 
  - For everyignossibole: Wonlderw
    - If  $\alpha$  is satisfied by w WeChat: cstutores
      - ullet If eta is not satisfied by w
        - Conclude that  $\alpha \not\models \beta$
  - Conclude that  $\alpha \models \beta$

$P_{1,1}$	$P_{1,2}$	4	$OK_{1,1}$	$OK_{2,1}$	$R_1$	$R_2$		$\neg B_{1,1}$	$\neg S_{1,1}$	$OK_{2,1}$
F	F	•••	<b>K</b> ssi	gnment	Project	Exam	Help	•••	Т	
				https://t	utores.c	om				
F	F	•••	Т	WeCha	t: cstuto	orcs	•••	•••	Т	Т
•••					•••					
Т	Т	•••	F	Т	F	•••	•••	•••	F	
Т	Т	•••	Т	F	T	F	•••	•••	Т	
Т	Т	•••	T	Т	T	Т		•••	F	

$P_{1,1}$	$P_{1,2}$	4	$OK_{1,1}$	$OK_{2,1}$	$R_1$	$R_2$		$\neg B_{1,1}$	$\neg S_{1,1}$	$OK_{2,1}$
F	F	•••	Essi	gnment	Project	Exam	Help <sup>*</sup>	•••	Т	
•••				nttps://t	utores.c	om			7 10	
F	F	•••	Т	WeCha	t: cstuto	orcs	•••	•••	Т	Т
•••					•••					
Т	Т	•••	F	Т	F	•••			F	
Т	Т	•••	Т	F	Т	F	•••	•••	Т	
Т	Т	•••	T	Т	Т	Т	•••	•••	F	

$P_{1,1}$	$P_{1,2}$		$OK_{1,1}$	$OK_{2,1}$	$R_1$	$R_2$	•••	$\neg B_{1,1}$	$\neg S_{1,1}$	$OK_{2,1}$
F	F	•••	<b>E</b> ssi	gnment	Project	Exam	Help	•••	Т	
•••				https://t	utores.c	om				
F	F	•••	Т	WeCha	t: cstuto	orcs	•••	•••	Т	Т
•••					•••					
Т	Т	•••	F	Т	F	•••	•••	•••	F	
Т	Т	•••	Т	F	Т	F	•••	•••	Т	
Т	Т	•••	T	Т	Т	Т		•••	F	

$P_{1,1}$	$P_{1,2}$		$OK_{1,1}$	$OK_{2,1}$	$R_1$	$R_2$	•••	$\neg B_{1,1}$	$\neg S_{1,1}$	$OK_{2,1}$
F	F	•••	Essi	gnment	Project	Exam	Help	•••	Т	
•••				nttps://t	utores.c	om				
F	F	•••	Т	WeCha	t: cstuto	rcsT			Т	Т
•••					•••					
T	Т	•••	F	T	F	•••			F	
T	Т	•••	Т	F	T	F		•••	T	
Т	Т	•••	Т	Т	Т	Т		•••	F	

n propositions m sentences, O(k) connectives

$P_{1,1}$	$P_{1,2}$	4	$OK_{1,1}$	$OK_{2,1}$	$R_1$	$R_2$		$\neg B_{1,1}$	$\neg S_{1,1}$	$OK_{2,1}$
F	F	•••	Essi	gnment	Project	Exam	Help	•••	Т	
				nttps://t	utores.c	om				
F	F	•••	Т	WeCha	t: cstute	orcs	•••		Т	Т
•••					•••					
Т	Т	• • •	F	Т	F	•••	•••	•••	F	
Т	Т	•••	Т	F	Т	F	•••	•••	Т	
Т	Т	•••	Т	Т	Т	Т	•••	•••	F	

 $(2^n mk)$  Intractable!

#### Entailment

- $\alpha$  entails  $\beta$  :  $\alpha \models \beta$ 
  - Every madelneft rajist elso reinhodel of  $\beta$
  - Whenever  $\alpha$  is true, so is  $\beta$

WeChat: cstutorcs

- $\beta$  is true in every world consistent with  $\alpha$
- $Models(\alpha) \subseteq Models(\beta)$
- $\beta$  logically follows from  $\alpha$

co-NP-complete!

# Propositional Logic

- Programming language for knowledge
- Factored representation (Beglean CSP)
  - Propositionspsconnectives, sentences
- Possible worlds hatis hability, models
- Entailment:  $\alpha \models \beta$ 
  - Every model of  $\alpha$  is a model of  $\beta$
  - Model checking intractable!

$P_{1,1}$	$P_{1,2}$	·	$OK_{1,1}$	$OK_{2,1}$	$R_1$	$R_2$		$\neg B_{1,}$	$\neg S_{1,}$	$OK_{2,1}$
F	F		F	F	F				Т	35
•••	A STATE	337				Port	The same	100	30	
F	F		Т	T	T	T			Т	T
•••	6.5			96		1616	N S	30%		10-8
Т	T		F	Т	F				F	
Т	Т		T	F	T	F			T	Part I
Т	Т		Т	T	Т	T			F	· at

Darn model checking...
so intractable

Assignment Project Exam Help

https://tutorcs.com

#### Rule: If you know $\alpha$ , then you also know $\beta$ .

Assignment Project Exam Help

https://tutorcs.com

#### Rule: If you know $\alpha$ , then you also know $\beta$ .

Assignment Project Exam Help

No http://defrechecking!

#### Rule: If you know $\alpha$ , then you also know $\beta$ .

Assignment Project Exam Help

No http://detrecking!

WeChat: cstutorcs

Seems impossible...

 $KB = \{ Hungry \Rightarrow Cranky, \\ Hungry \}$  ignment Project Exam Help

 $KB = \{ Hungry \Rightarrow Cranky, \\ Hungry \Rightarrow Signment Project Exam Help \}$ 

WeChat: cstutorcs

Cranky

$$KB = \{ Hungry \Rightarrow Cranky, \\ Hungry \Rightarrow Signment Project Exam Help \}$$

Cranky

https://tutorcs.com

Hungry	Cranky	$Hungry \Rightarrow Cranky$
F	F	T
F	T	T
T	F	F
T	THE	T

$$KB = \{ Hungry \Rightarrow Cranky, \\ Hungry \Rightarrow Signment Project Exam Help \}$$

Cranky

https://tutorcs.com

Hungry	Cranky	$Hungry \Rightarrow Cranky$
F	F	T
F	T	T
T	F	F
T	THE	T

$$KB = \{ Hungry \Rightarrow Cranky, \\ Hungry \Rightarrow Signment Project Exam Help \}$$

Cranky

https://tutorcs.com

Hungry	Cranky	$Hungry \Rightarrow Cranky$
F	F	T
F	T	T
T	F	F
THE	T	T

 $KB = \{ Hungry \Rightarrow Cranky, \\ Hungry \Rightarrow Signment Project Exam Help \}$ 

Cranky

https://tutores.com/ky

Hungry	Cranky	$Hungry \Rightarrow Cranky$
F	F	T
F	T	T
T	F	F
T	T	T

Assignment Project Exam Help

Hungry \Rightarrow Cranky, Hungry

https://tutorcs.com

WeChat. Cstutores

Premises

(Antecedents)

Assignment Project Exam Help  $Hungry \Rightarrow Cranky, Hungry$ https://tutorcs.com

WeChar. Pankys

```
Premises
(Antecedents)
```

Assignment Project Exam Help  $Hungry \Rightarrow Cranky, Hungry$ https://tutorcs.com

WeChat Cranky

Conclusions (Consequents)

Assignment Project Exam Help  $Hungry \Rightarrow Cranky, Hungry$ https://tutores.com

WeChar. Cranky

Hungry	Cranky	$Hungry \Rightarrow Cranky$
F	F	T
F	T	T
T	F	F
T	T	T

Assignment Project Exam Help  $\varphi \Rightarrow \psi, \varphi$  https://tutorcs.com\_\_\_\_

$\varphi$	$\psi$	$\varphi \Rightarrow \psi$
F	F	T
F	T	T
T	F	F
THE	T	T

# Modus Ponens

"mode that affirms"

Assignment Project Exam Help  $\varphi \Rightarrow \psi, \varphi$ Nttps://tutorcs.com
WeChat: cstutorcs

# Derivation

 $\{ Hungry \Rightarrow Cranky, Hungry \} \vdash_{MP} Cranky$ Assignment Project Exam Help

https://tutorcs.com

# Derivation

```
\{ Hungry \Rightarrow Cranky, Hungry \} \vdash Cranky
Assignment Project Exam Help
```

https://tutorcs.com

# Derivation

$$\alpha \vDash \beta$$

$$\alpha \vdash \beta$$

 $\beta$  logically follows from  $\alpha$  Easier to compute https://tutorcs.com

Intractable to compute cstutores

But does  $\alpha$  really follow from  $\beta$ ?

# Soundness

Assignment Project Exam Help

If determinen  $\alpha \models \beta$ We Chat: estutores

# Soundness

Assignment Project Exam Help

If  $\alpha = \beta$  We Chat: cstutores

Modus Ponens is sound

Hungry
Assignment Project Exam Help

https://tutorcs.com

$$\frac{\varphi \Rightarrow \psi, \ \psi}{\varphi}$$

 $\varphi \Rightarrow \psi, \psi$ 

Hungry

Assignment Project Exam Help

https://tutorcs.com

	$\varphi$	$\psi$	$\varphi \Rightarrow \psi$
	F	F	T
<b>-</b>	F	T	T
	T	F	F
<b>+</b>	T	T	T

 $\varphi \Rightarrow \psi, \psi$ 

Hungry

Assignment Project Exam Help

https://tutorcs.com

	$\varphi$	$\psi$	$\varphi \Rightarrow \psi$
	F	F	T
	F	T	T
	T	F	F
-	T	T	T

 $\varphi \Rightarrow \psi, \psi$ 

Hungry

Assignment Project Exam Help

$$\{\psi^{\text{ps://tut/pres-point}} \neq \varphi$$

$\varphi$	$\psi$	$\varphi \Rightarrow \psi$
F	F	T
F	T	T
T	F	F
T	T	T

Hungry

Assignment Project Exam Help

$$\{\psi^{\text{ps://tut/pres-point}} \neq \varphi$$

WeChat: cstutorcs

$\varphi$	$\Rightarrow \psi$ ,	$\psi$
	$\varphi$	

Unsound!

$\varphi$	$\psi$	$\varphi \Rightarrow \psi$
F	F	T
F	T	T
T	F	F
T	T	T

# Affirming the Consequent

$$Hungry \Rightarrow Cranky, Cranky$$

Assignment Project Exam Help

$$\{\psi^{\text{tut/pres-ton}} \neq \varphi$$

WeChat: cstutorcs

$\varphi$	$\Rightarrow$	y	b,	$\psi$
250	1	1111111	THE REAL PROPERTY.	3115

4

Unsound!

$\varphi$	$\psi$	$\varphi \Rightarrow \psi$
F	F	T
F	T	T
T	F	F
T	E THE	T

#### $Hungry \Rightarrow Cranky, \neg Cranky$

 $\neg Hungry$ 

Assignment Project Exam Help

https://tutorcs.com

$$\frac{\varphi \Rightarrow \psi, \ \neg \psi}{\neg \varphi}$$

$$\frac{Hungry \Rightarrow Cranky, \neg Cranky}{\neg Hungry}$$

 $\frac{\varphi \Rightarrow \psi, \, \neg \psi}{\neg \varphi}$ 

https://tutorcs.com

Assignment Project Exam Help

$\varphi$	$\psi$	$\varphi \Rightarrow \psi$	$\neg \varphi$	$ eg \psi$
F	F	T	T	T
F	Т	T	Т	F
T	F	F	F	T
T	T	T	F	F

# $\frac{Hungry \Rightarrow Cranky, \neg Cranky}{\neg Hungry}$ $\frac{\neg Hungry}{Assignment Project Exam Help}$

 $\frac{\varphi \Rightarrow \psi, \, \neg \psi}{\neg \varphi}$ 

https://tutorcs.com

STATE OF	$\varphi$	$\psi$	$\varphi \Rightarrow \psi$	$\neg \varphi$	$ eg \psi$
	F	F	T	T	Ţ
	F	Т	T	Т	F
	T	F	F	F	T
	T	T	T	F	F

# $\frac{Hungry \Rightarrow Cranky, \neg Cranky}{\neg Hungry}$

 $\frac{\varphi \Rightarrow \psi, \, \neg \psi}{\neg \varphi}$ 

https://tutorcs.com

Assignment Project Exam Help

STATE OF	$\varphi$	$\psi$	$\varphi \Rightarrow \psi$	$\neg \varphi$	$ eg \psi$
	F	F	T	T	T
	F	Т	T	Т	F
	T	F	F	F	T
	T	T	T	F	F

# $Hungry \Rightarrow Cranky, \neg Cranky$ $\neg Hungry$ Assignment Project Exam Heli

$$\frac{\tau}{\neg \varphi}$$

$$\{ \varphi^{\text{https://thtores/point}} \models \neg \varphi$$

Sound!

$\varphi$	$\psi$	$\varphi \Rightarrow \psi$	$\neg \varphi$	$ eg \psi$
F	F	T	T	T
F	Т	T	Т	F
T	F	F	F	T
Т	T	Т	F	F

# Modus Tollens

$$\text{MT:} \frac{Hungry \Rightarrow Cranky, \neg Cranky}{\neg Hungry} \qquad \text{MT:} \frac{\varphi \Rightarrow \psi, \neg \psi}{\neg \varphi}$$

$$\text{Assignment Project Exam Help}$$

$$\{ \varphi^{\text{https://thtorcs/ton}} \models \neg \varphi$$

Sound!

	$\varphi$	$\psi$	$\varphi \Rightarrow \psi$	$\neg \varphi$	$ eg \psi$
<b>+</b>	F	F	T	T	T
	F	T	T	T	F
	T	F	F	F	T
	T	T	T	F	F

# Equivalences

For any sentences 
$$\varphi$$
 and  $\psi$  If  $\phi$  the side with the properties  $\phi$  and  $\phi$  and  $\phi$ 

are inference rules

# Equivalences AIMA Fig. 7.11

$$(\alpha \wedge \beta) = (\beta \wedge \alpha) \qquad \qquad \text{Commutativity of } \wedge \\ (\alpha \vee \beta) = (\beta \vee \alpha) \qquad \qquad \text{Commutativity of } \vee \\ ((\alpha \wedge \beta) \wedge \gamma) = (\alpha \wedge (\beta \wedge \gamma)) \qquad \qquad \text{Associativity of } \wedge \\ ((\alpha \vee \beta) \wedge \gamma) = (\alpha \vee (\beta \vee \gamma)) \qquad \qquad \text{Associativity of } \vee \\ ((\alpha \vee \beta) \wedge \gamma) = (\alpha \vee (\beta \vee \gamma)) \qquad \qquad \text{Associativity of } \vee \\ ((\alpha \vee \beta) \wedge \gamma) = (\alpha \vee (\beta \vee \gamma)) \qquad \qquad \text{Double-negation elimination} \\ (\alpha \Rightarrow \beta) = (\neg \alpha) \qquad \qquad \text{Double-negation elimination} \\ (\alpha \Rightarrow \beta) = (\neg \alpha \vee \beta) \qquad \qquad \text{Implication elimination} \\ (\alpha \Leftrightarrow \beta) = ((\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)) \qquad \qquad \text{Biconditional elimination} \\ (\alpha \Leftrightarrow \beta) = ((\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)) \qquad \qquad \text{De Morgan's law} \\ (\alpha \wedge (\beta \vee \gamma)) = ((\alpha \wedge \beta) \vee (\alpha \wedge \gamma)) \qquad \text{Distributivity of } \wedge \text{ over } \vee \\ (\alpha \vee (\beta \wedge \gamma)) = ((\alpha \vee \beta) \wedge (\alpha \vee \gamma)) \qquad \text{Distributivity of } \vee \text{ over } \wedge \text{ over }$$

$$\frac{\alpha \wedge \beta}{\alpha}$$

$$\frac{\neg \neg \alpha}{\alpha}$$

$$\frac{\neg(\alpha \land \beta)}{\neg \alpha \lor \neg \beta} \quad \frac{\neg(\alpha \lor \beta)}{\neg \alpha \land \neg \beta}$$

$$\frac{\neg(\alpha \lor \beta)}{\neg \alpha \land \neg \beta}$$

And-eliminationignm Double Exam Help De Morgan's

httpsegation

Laws

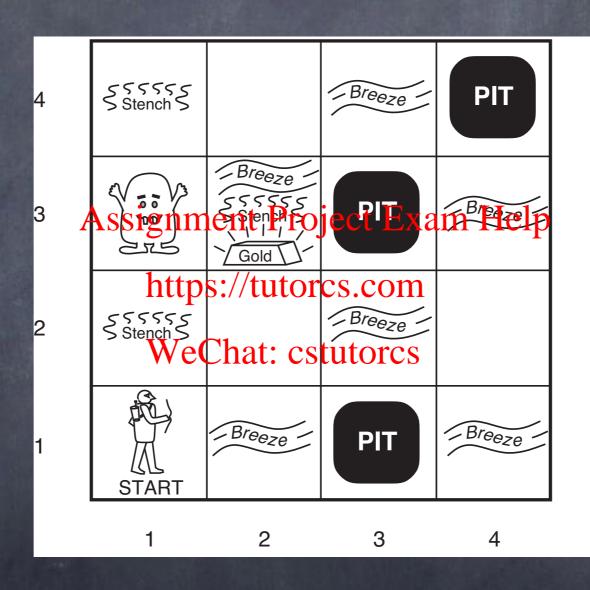
$$\frac{\alpha \Rightarrow \beta, \alpha}{\beta}$$

$$\frac{\alpha \Leftrightarrow \beta}{(\alpha \Rightarrow \beta) \land (\beta \Rightarrow \alpha)}$$

$$\frac{\alpha \Leftrightarrow \beta}{(\alpha \Rightarrow \beta) \land (\beta \Rightarrow \alpha)} \qquad \frac{(\alpha \Rightarrow \beta) \land (\beta \Rightarrow \alpha)}{\alpha \Leftrightarrow \beta}$$

Modus Ponens

Definition of biconditional



### Proof

- Inference rules produce theorems derived from other sentences Assignment Project Exam Help
- The sequence of the rule applications used in whe derivation constitutes a proof of the theorem

# Proof

#### Given what I know... Is there no pit in room [1,2]?

 $\neg P_{1,2}$ ?

Assignment Project Exam Help

1: 
$$\neg P_{1,1}$$
 https://tutorcs.com

2: 
$$B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,p})$$
 that: cstutores

$$3: B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$$

4: 
$$\neg B_{1,1}$$

5: 
$$B_{2,1}$$

1:  $\neg P_{1,1}$ 

2:  $B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$ 

3:  $B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$ 

 $4: \ \overline{\neg B_{1,1}}$ 

5:  $B_{2,1}$ 

Rule	Premises	Genclusion Help
Bicond elim	2 h	$\begin{array}{c} \bullet \\ \bullet \\ \bullet \\ B_{1,1} \end{array} \stackrel{\text{com}}{\Rightarrow} (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}) \end{array}$
And elim	6	$7 = Chat: cstutorcs \\ 7: ((P_{1,2} \lor P_{2,1}) \Rightarrow B_{1,1})$
Contrapositive	7	8: $(\neg B_{1,1} \Rightarrow \neg (P_{1,2} \lor P_{2,1}))$
MP	8, 4	9: $\neg (P_{1,2} \lor P_{2,1})$
De Morgan	9	10: $\neg P_{1,2} \wedge \neg P_{2,1}$
And elim	10	11: $\neg P_{1,2}$

# Proof

- Each step's premises must be in the KB already
   Assignment Project Exam Help
- Each step's conclusionois added to the KB
   WeChat: cstutorcs
- The last step derives the query

$$KB \vdash \neg P_{1,2}$$

## Proof

- Each step's premises must be in the KB already
   Assignment Project Exam Help
- Each step's conclusionois added to the KB WeChat: cstutorcs
- The last step derives the query

$$KB \vdash \neg P_{1,2}$$

• If all the inference rules are sound

$$KB \vDash \neg P_{1,2}$$

## Proof as Search

- States are sets of sentences (KBs)
- Actions are applying interence rules
  - Actions(s) = ttps r/tutor Match(Premises(r), KB) = m
  - $Result(r_m, s) \stackrel{\mathsf{VeChat:}}{=} s \ Usubst(m, Conclusions(r))$
- $\bullet$  Initial state: initial KB
- Goal test:  $query \in KB$

- Searching for proofs is an alternative to enumerating models.

  Assignment Project Exam Help
- "In many practical cases, finding a proof can be more afficient because the proof can ignore irrelevant propositions, no matter how many of them there are."

- States are sets of sentences (KBs)
- Actions are applying interence rules
  - Actions(s) = ttps r/tutor Match(Premises(r), KB) = m
  - $Result(r_m, s) \stackrel{\mathsf{VeChat:}}{=} s \overset{\mathsf{cstattores}}{=} t(m, Conclusions(r))$
- $\bullet$  Initial state: initial KB
- Goal test:  $query \in KB$

Need a complete search strategy

Assignment Project Exam Help

https://tutorcs.com

- Need a complete search strategy
- · Need a complete set cétainference rules

https://tutorcs.com

# Completeness

Assignment Project Exam Help

If  $\alpha \vdash \beta$ 

## Completeness

Assignment Project Exam Help

If determine 
$$\alpha \vdash \beta$$
 We Chat: cstutores

MP: 
$$\frac{\varphi \Rightarrow \psi, \varphi}{\psi}$$

Modus Ponens is not complete

# Proof

Rule	Premises	Conclusion
Bicond elim	Assign	$G_{\bullet}^{\text{pent-Project Exam Help}}(B_{1,1}^{\bullet}) \wedge ((P_{1,2} \vee P_{2,1})) \rightarrow B_{1,1})$
And elim	6	$7: ((P_{1,2} \lor P_{2,1}) \Rightarrow B_{1,1})$
Contrapositive	7	8: $(\neg B_{1,1} \Rightarrow \neg (P_{1,2} \lor P_{2,1}))$
MP	8, 4	9: $\neg (P_{1,2} \lor P_{2,1})$
De Morgan	9	10: $\neg P_{1,2} \wedge \neg P_{2,1}$
And elim	10	11: $\neg P_{1,2}$

- Need a complete search strategy
- · Need a complete set cétainference rules

https://tutorcs.com

- Need a complete search strategy
- · Need a complete set refainference rules
  - Or a singlette of the inference rule



 $Hungry \lor Cranky$ 

 $\neg Hungry$ 

Cranky

Assignment Project Exam Help

https://tutorcs.com

 $B_{2,1}$ 

1,4	2,4	3,4	4,4	$P_{1,1} \vee P_{2,2} \vee P_{3,1}$
1,3	2,3	3,3	4,3	$\neg P_{1,1}$
1,2	2,2 P?	<b>A</b> 3,2	ssignme <sub>4,2</sub>	nt Project Exam Help
OK 1,1	2,1 p	3,1		//tutorcs.com $P_{2,2} \lor P_{3,1}$ nat: cstutorcs
V OK	B A OK	3,1 P?		$ eg P_{2,2}$

 $P_{3,1}$ 

## Reasoning By Cases

If A or B is true and you know it's not A, https://tutorcs.com

thereit: musts be B

 $l_1 \lor \dots \lor l_i \lor \dots \lor l_k$   $\neg l_i$  https://tutorcs.com  $l_1 \lor \dots \lor l_{i-1} \lor v \lor l_{i-1} \lor v \lor l_k$ 

 $l_1 \lor \dots \lor l_i \lor \dots \lor l_k - l_i$   $l_1 \lor \dots \lor l_i \lor \dots \lor l_k$ https://tutorcs.com

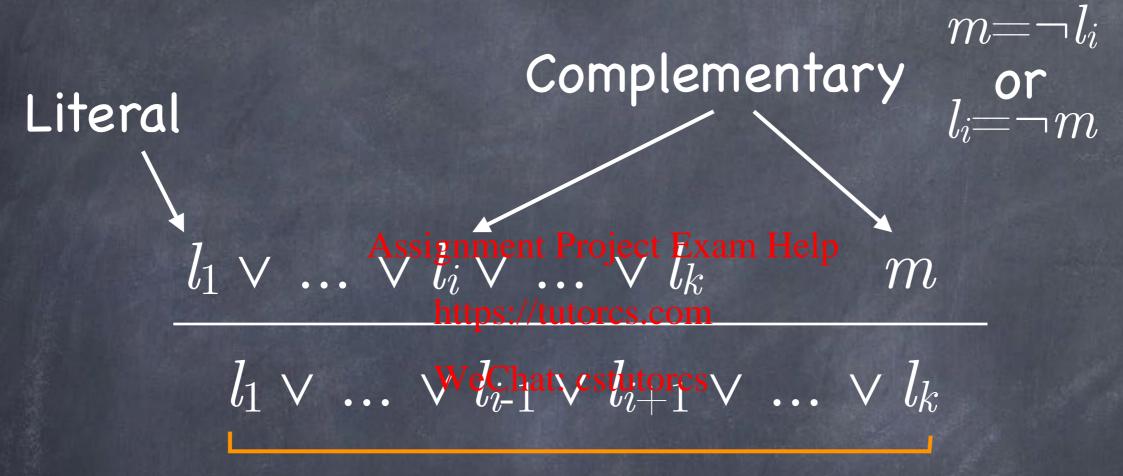
 $l_1 \lor \dots \lor l_i \lor \dots \lor l_k$   $\neg l_i$  https://tutores.com  $l_1 \lor \dots \lor l_k$   $\lor l_i \lor \dots \lor l_k$ 

 $l_i$  is gone

#### Literals

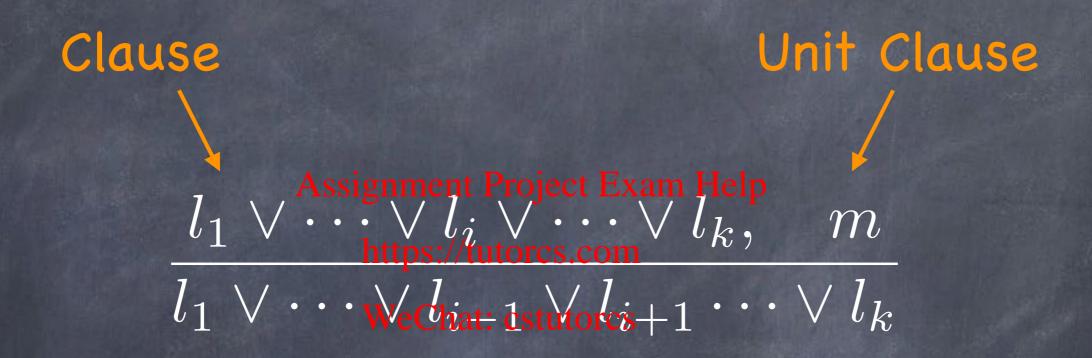
$$l_1 \lor \ldots \bigvee_{l_i}^{\mathsf{Assignment Project Exam Help}} \lnot l_i$$
 $l_1 \lor \ldots \bigvee_{\mathsf{https://tutores.com}}^{\mathsf{Valify}} \lnot l_i$ 

Literals  $l_1 \lor \ldots \lor l_i \lor \ldots \lor l_k$  Complementary  $l_i = \neg m$   $l_i \lor \ldots \lor l_k \lor \ldots \lor l_k$ 



Clause

### Unit Resolution



 $l_1, \ ..., \ l_k$  and m are literals  $l_i$  and m are complementary



- 1.  $\overline{Hungry} \vee \overline{Cranky}$
- $2. \neg Hungry$

Q: Cranky?

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs

Unit Res: Hungry

1,2

3. Cranky

 $KB \vdash_{UR} Cranky$ 

1,4	2,4	3,4	4,4
1.0	0.0	2.2	4.0
1,3	2,3	3,3	4,3
1,2	2,2 P?	3,2	4,2
	1:		
OK			
1,1	2,1 B	3,1 P?	4,1
V	A	1 :	Ass
OK	OK		

1. 
$$P_{1,1} \vee P_{2,2} \vee P_{3,1}$$

2. 
$$\neg P_{1,1}$$

3. 
$$\neg P_{2,2}$$

ignment Project Exam Help

https://tutorcs.com

Unit Res: $P_{1,1}$	1,2	4. $P_{2,2} \vee P_{3,1}$
Unit Res: $P_{2,2}$	4,3	5. $P_{3,1}$

$$KB \vdash_{UR} P_{3,1}$$

### Unit Resolution

- Sound: if  $\alpha \vdash \beta$  then  $\alpha \models \beta$ 
  - Easy to shown Project Exam Help
- Not completeles:/ifftoras.com eta then  $\alpha \vdash eta$ 
  - Give a counterexample



 $Hungry \lor Cranky$ 

 $\neg Hungry \lor Sleepy$ 

Assignment Project Exam HelpSleepy

https://tutorcs.com

### Resolution

 $\frac{l_1 \vee \cdots \vee l_i \vee \cdots \vee l_k, \quad m_1 \vee \cdots \vee m_j \vee \cdots \vee m_n}{l_1 \vee \cdots \vee l_{i-1} \vee l_{i+1} \cdots \vee l_k \vee m_1 \vee \cdots \vee m_{j-1} \vee m_{j+1} \cdots \vee m_n}$ 

Assignment Project Exam Help  $l_1, \ ..., \ l_k$  ,  $m_{1/tutores.com}$  are literals

 $l_i$  and  $m_j^{\text{regree}}$  ctomplementary

 $l_i$  and  $m_j$  are gone

Technical note: Resulting clause must be <u>factored</u> to contain only one copy of each literal.

(See AIMA)



- 1.  $Hungry \lor Cranky$
- 2.  $\neg Sleepy \lor \neg Hungry$
- 3.  $Cranky \lor Sleepy$

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs

Rule	Premises	Conclusion
Resolution: Hungry	1,2	4. $Cranky \lor \neg Sleepy$
Resolution: Sleepy	3,4	5. $Cranky \vee Cranky$
Factoring	5	6. Cranky

 $KB \vdash Cranky$ 

## Resolution

• Sound:

if 
$$\alpha \vdash \beta$$
 then  $\alpha \models \beta$ 

• Easy to shown Project Exam Help

https://tutorcs.com

### Resolution

- Sound: if  $\alpha \vdash \beta$  then  $\alpha \models \beta$ 
  - Easy to shown Project Exam Help
- Not complete  $\beta$  then  $\alpha \vdash \beta$ 
  - Give a counterexample

# Resolution is Refutation-Complete

Assignment Project Exam Help

If a set of clauses is unsatisfiable, then resolution can derive the empty clause
 (□)

AIMA p. 255

### Resolution

$$\frac{l_1 \vee \cdots \vee l_i \vee \cdots \vee l_k, \quad m_1 \vee \cdots \vee m_j \vee \cdots \vee m_n}{l_1 \vee \cdots \vee l_{i-1} \vee l_{i+1} \cdots \vee l_k \vee m_1 \vee \cdots \vee m_{j-1} \vee m_{j+1} \cdots \vee m_n}$$

Assignment Project Exam Help  $l_1, \ldots, l_k$ ,  $m_1$ , tutores.  $m_n$  are literals

 $l_i$  and  $m_j^{\text{regree}}$  ctomplementary

Technical note: Resulting clause must be <u>factored</u> to contain only one copy of each literal.

(See AIMA)

# Challenges for Using Resolution

Only works on clauses
 https://tutorcs.com

WeChat: cstutorcs

Only refutation-complete

# Conjunctive Normal Form (CNF)

#### Assignment Project Exam Help

• Any sentence of propositional logic can be converted into an equivalent conjunction (set) of clauses

# Conjunctive Normal Form (CNF)

- Eliminate  $\Leftrightarrow$ :  $\alpha \Leftrightarrow \beta \rightarrow \alpha \Rightarrow \beta \land \beta \Rightarrow \alpha$
- Eliminate Assignment Project Exam Help
- Move negationhttps://tutorcs.com
  - $\neg \neg \alpha \rightarrow \alpha$  WeChat: cstutorcs
  - $\bullet \neg(\alpha \lor \beta) \rightarrow (\neg\alpha \land \neg\beta)$
  - $\bullet \neg(\alpha \land \beta) \rightarrow (\neg\alpha \lor \neg\beta)$
- Distribute \( \) over \( \):
  - $(\alpha \lor (\beta \land \gamma)) \rightarrow ((\alpha \lor \beta) \land (\alpha \lor \gamma))$

AIMA p. 253-254

# Challenges for Using Resolution

- Only works on clauses
   https://tutorcs.com
  - Convert Kille & Convert & Co

Only refutation-complete

# Challenges for Using Resolution

- Only works on clauses
   https://tutorcs.com
  - · Convert KB & query to clauses (CNF)

• Only refutation-complete

# Resolution is Refutation-Complete

Assignment Project Exam Help

• If a set of clauses is unsatisfiable, then resolution can derive the empty clause (□)

a clause with zero literals

# Entailment and Satisfiability

- If a set of clauses is unsatisfiable, then resolution can derive the empty clause (□) Assignment Project Exam Help
- $KB \models \beta$  https://tutorcs.com
  - iff every model of  $\beta$
  - ullet iff no model of  $K\!B$  is a model of  $\neg eta$
  - iff there are no models of  $KB \cup \{ \neg \beta \}$
  - iff  $KB \cup \{ \neg \beta \}$  is unsatisfiable

- Convert  $KB \cup \{ \neg \beta \}$  to CNF
- Apply resalution rujet Entitlesp
  - No new clauses can be added
    - $KB \nvDash \beta$
  - Two clauses resolve to yield the empty clause (contradiction)
    - $leftharpoonup KB \models \beta$

Given what I know... Is there no pit in room [1,2]?

Assignment Project Exam Help

https://tutorcs.com

2: 
$$B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,p})$$
 that: cstutorcs

4: 
$$\neg B_{1,1}$$

$$\neg P_{1,2}$$
?

$$KB = \{ B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1}), \neg B_{1,1} \}$$

Query:  $\neg P_{1,2}$ ?

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs

# Add -Query to KB

 $B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1}), \neg B_{1,1}, \neg \neg P_{1,2}$ 

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutorcs

## Convert to CNF

$$B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1}), \neg B_{1,1}, \neg \neg P_{1,2}$$

$$B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1}), \stackrel{\text{Assignment Project Exam Help}}{(P_{1,2} \vee P_{2,1})} \Rightarrow B_{1,1}, \neg B_{1,1}, \neg \neg P_{1,2}$$

$$\neg B_{1,1} \vee (P_{1,2} \vee P_{2,1}), \neg \stackrel{\text{Plust}}{(P_{1,2} \vee P_{2,1})} \vee B_{1,1}, \neg B_{1,1}, \neg \neg P_{1,2}$$

$$\neg B_{1,1} \vee (P_{1,2} \vee P_{2,1}), (\neg P_{1,2} \wedge \neg P_{2,1}) \vee B_{1,1}, \neg B_{1,1}, P_{1,2}$$

$$\neg B_{1,1} \vee (P_{1,2} \vee P_{2,1}), (\neg P_{1,2} \vee B_{1,1}), (\neg P_{2,1} \vee B_{1,1}), \neg B_{1,1}, P_{1,2}$$

$$\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}, \neg P_{1,2} \vee B_{1,1}, \neg P_{2,1} \vee B_{1,1}, \neg B_{1,1}, P_{1,2}$$

1: 
$$\neg B_{1,1} \lor P_{1,2} \lor P_{2,1}$$

$$2: \neg P_{1,2} \lor B_{1,1}$$

$$3: \neg P_{2,1} \lor B_{1,1}$$

4: 
$$\neg B_{1,1}$$

$$5: P_{1,2}$$

Assignment Project Exam Help

https://tutorcs.com

WeChat: cstutores

Premises	Literal	Result
2,4	$B_{1,1}$	6: $\neg P_{1,2}$
5,6	$P_{1,2}$	7:

1: 
$$\neg B_{1,1} \lor P_{1,2} \lor P_{2,1}$$

$$2: \neg P_{1,2} \lor B_{1,1}$$

$$3: \neg P_{2,1} \lor B_{1,1}$$

4: 
$$\neg B_{1,1}$$

 $5: P_{1,2}$ 

Assignment Project Exam Help

https://tutorcs.com

$$2: \neg P_{1,2} \land B_{1,1}$$
 $4: \neg B_{1,1}$ 
 $6: \neg P_{1,2}$ 
 $5: P_{1,2}$ 
 $7: \square$ 

- Convert  $KB \cup \{ \neg \beta \}$  to CNF
- Apply resalution rujet Entitlesp
  - No new clauses can be added
    - $KB \nvDash \beta$
  - Two clauses resolve to yield the empty clause (contradiction)
    - $leftharpoonup KB \models \beta$

### Resolution

- Complete when used in a refutation (proof by contradiction) Assignment Project Exam Help
- Search challengesoremaining:
  - Which clauses to resolve?
  - On which complementary literals?

### Resolution

- Complete when used in a refutation (proof by contradiction) Assignment Project Exam Help
- Search challengesoremaining:
  - Which clauses to resolve?
  - On which complementary literals?

Intractable!

Assignment Project Exam Help

# But waiththere's more...

# Theorem Proving

- States are sets of sentences (KBs)
- Actions are applying interence rules
  - Actions(s) = ttps r/tutor Match(Premises(r), KB) = m
  - $Result(r_m, s) \stackrel{\mathsf{VeChat:}}{=} s \overset{\mathsf{cstattores}}{=} t(m, Conclusions(r))$
- $\bullet$  Initial state: initial KB
- Goal test:  $query \in KB$

# Inference for Knowledge-Based Agents

Assignment Project Exam Help

Data-driven (for ward whaining)

WeChat: cstutores

Goal-directed (backward chaining)

## Forward Chaining

- Given new fact  $\varphi$  (often perception)
  - Add φ to agenda Assignment Project Exam Help
  - · While agendais not empty
    - · Remove sentence sentence sentence
    - Add α to KB
    - Apply inference using only rules whose premises include  $\alpha$ 
      - ullet Add conclusion eta to agenda

# Forward Chaining

- Reasons forward from new facts
  - Data-drivenment Project Exam Help
- Done by humans tutos some extent
  - When to stop?

## Forward Chaining

- Reasons forward from new facts
  - Data-drivenment Project Exam Help
- Done by humans tutos some extent
  - When to stop?
- For KBs using only definite clauses
  - · Sound, complete, linear time

AIMA 7.5.4

# Backward Chaining

- In order to prove  $\beta$ 
  - Find assigimplitagtionsawhose conclusions: isutocs.com
  - Try to prove its premises α (recursively)

# Backward Chaining

- Reasons backward from query
  - Goal-dizected Project Exam Help
- Useful for attswering specific questions

# Backward Chaining

- Reasons backward from query
  - Goal-dizected Project Exam Help
- Useful for ahttwerting oppecific questions WeChat: cstutorcs
- For KBs using only definite clauses
  - · Sound, complete, linear time

AIMA 7.5.4

# Propositional Theorem Proving

- Inference rules: Soundness, Completeness
- Proof:  $\alpha \vdash \beta$ Assignment Project Exam Help
  - Searching for proofs is an alternative to enumerating models; "can be more efficient"
- Resolution is a sound and complete inference rule
  - Works on clauses (CNF); requires refutation
- Forward and backward chaining

### For next time:

Assignment Project Exam Help

https://tutorcs.com

AIMMechat: Stutore -8.3;

8.1.1-8.1.2 fyi