Lógica

Prof. Mário Benevides
mario@cos.ufrj.br
Março de 2008

LÓGICA

Índice:

.

Motivação Prática:

- Álgebra de Boole
- Programação em lógica, p. e. PROLOG
- Sistemas especialistas
- Especificação de programas
- Banco de dados:
    - dedutivos
    - hipótese de mundo fechado
    - default / prioridades
- Sistemas distribuídos:
    - tempo
    - recursos
- Lei (Lógica deôntica, p.e.)
- Linguagens de programação

Livros:

- Lógica para a Computação - Cengage Learning - *Flávio Soares Corrêa da Silva, Marcelo Finger e Ana Cristina Vieira de Melo*
- Lógica para a Ciência da Computação – Ed. Campus
  João Nunes de Souza
- Introdução à Lógica Modal Aplicada a Computação
  Marcos Mota Costa
- Programação em Lógica e a Linguagem PROLOG
  Casanova
- Lógica
  John Nolt e Linnes Rohatyn
- A Mathematical Introduction to Logic
  Enderton
- Introduction to Mathematical Logic
  Mendelson

I - Introdução

LÓGICA é o estudo do raciocínio dedutivo.

Histórico

- Aristóteles → leis do discurso;
- Idade Média → lógica filosófica;
- Boole (1815-1864) → álgebra booleana;
- Peano (c.1865) → axiomatização da aritmética;
- Frege (1874) → investigar fundamentos da matemática;
  → lógica moderna;
- Russel-Wentehead (1910) → Princípio Matemático - lógica moderna;
- Hilbert (1925) → formalização da noção de prova;
  → mecanização da matemática;
- Gentzen (1935) → teoria da prova;
- Godel (1931-1935) → completude da lógica;
  → incompletude da aritmética
- Investigar Fundamentos da Computação.

➩ O que é Lógica?
É o estudo do raciocínio dedutivo. (informal)

Mais formalmente:
➩ Lógica é uma linguagem?

Lógica:

LINGUAGEM
+
REGRAS DE DEDUÇÃO / INFERÊNCIA
+
SEMÂNTICA

Linguagem

É usada para descrever o conhecimento que se deseja representar.

Regras de Dedução

Servem para tirar conclusões a partir do conhecimento representado na linguagem.

Semântica

Serve para dar significado aos objetos descritos na linguagem.

Tipos mais comuns de lógica:

- Lógica Clássica Proposicional
- Lógica Clássica de 1ª Ordem

Hipóteses da Lógica Clássica:

- proposicões atômicas;
- proposições mais complexas podem ser construídas (decompostas) de (em) proposições atômicas;
- proposições atômicas são <u>verdadeiras</u> ou <u>falsas</u> (2 valores);
- o valor verdade de uma proposição mais complexa somente depende dos valores das proposições atômicas que a compõe.

Ex1: João ama Maria. (V ou F)

Ex2: João é estudante <u>e</u> João é alto.

→ o valor verdade só depende de sabermos se:

João é estudante. (V ou F?)

João é alto. (V ou F?)

II-1. Linguagem da Lógica Clássica Proposicional

- A linguagem proposicional é uma linguagem formal cujo objetivo é representar trechos de discurso de uma maneira precisa e sem ambigüidades.
- Os seguintes operadores são usados para formar proposições mais complexas:

e	$\wedge$
ou	$\vee$
se <condição> então <conclusão>	$\rightarrow$
não	$\neg$

- Para representar proposições atômicas usaremos letras maiúsculas, por exemplo: A, B, C,...

Exemplos: $A \wedge B$, $A \vee B$, $A\rightarrow B$, $\neg A$

Ex: Sócrates é um homem.
Se Sócrates é um homem *então* Sócrates é mortal.

A - Sócrates é um homem.
B - Sócrates é mortal.

BD: A
$A\rightarrow B$

Definição:

Um *alfabeto proposicional* é composto por três conjuntos de símbolos:
Conectivos/operadores lógicos: $\wedge$, $\vee$, $\rightarrow$, $\neg$, $\leftrightarrow$
Símbolos auxiliares: (e)
Símbolos proposicionais: qualquer letra maiúscula é um símbolo proposicional (ex: A, B,..., Z). Podemos acrescentar um subscrito numérico a letras maiúsculas (A_1, A_2,...). Denotamos este conjunto por P.

Apresentaremos a seguir uma gramática para definirmos quais são as *fórmulas bem formadas* da linguagem:

Definição:

A noção de *fórmula bem formada*, ou simplesmente *fórmula*, é definida, indutivamente, pelas seguintes condições:
Qualquer símbolo proposicional é uma fórmula.
Se α e β são fórmulas então $(\alpha \wedge \beta)$, $(\alpha \vee \beta)$, $\neg \alpha$, $(\alpha \rightarrow \beta)$, $(\alpha \leftrightarrow \beta)$ também o são;
Nada mais é fórmula.

Exemplos:
a) $A \rightarrow B$ (não é fórmula)
b) $(A) \rightarrow \neg (B)$ (não é fórmula)
c) $(\neg A \vee B) \wedge (B \wedge C) \rightarrow D$ (não é fórmula)
d) $((A \rightarrow (B \rightarrow \neg A)) \rightarrow (A \vee B))$ (é fórmula)
e) $(A \rightarrow (B \wedge C))$ (é fórmula)

Exercícios:

1. Represente as seguintes proposições utilizando a linguagem da lógica clássica proposicional. Utilize os símbolos proposicionais C (está chovendo) e N (está nevando).

a) Está chovendo, mas não está nevando.
b) Não é o caso que está chovendo ou nevando.
c) Se não está chovendo, então está nevando.
d) Não é o caso que se está chovendo então está nevando.
e) Está chovendo se e somente se está nevando.
f) Se está nevando e chovendo, então está nevando.
g) Se não está chovendo, então não é o caso que está nevando e chovendo.

Nos exercícios seguintes, represente o texto na linguagem da lógica proposicional, especificando significado dos símbolos proposicionais utilizados.

2. Ela não está em casa ou não está atendendo ao telefone. Mas se ela não está em casa, então ela foi seqüestrada. E se ela não está atendendo ao telefone, ela está correndo algum outro perigo. Ou ela foi seqüestrada ou ela está correndo um outro perigo.

3. Hoje é fim-de-semana se e somente se hoje é sábado ou domingo. Hoje não é sábado. Hoje não é domingo. Portanto, hoje não é um fim-de-semana.

4. A proposta de auxílio está no correio. Se os juízes a receberem até sexta-feira, eles a analisarão. Portanto, eles a analisarão porque se a proposta estiver no correio, eles a receberão até sexta-feira.

Observação:
- Convenções sobre omissão de parênteses:

$$\neg \; > \; \wedge \; > \; \vee \; > \; \rightarrow$$

$$\neg A \rightarrow B \equiv (\neg A \rightarrow B) \qquad\qquad A \rightarrow A \wedge B \equiv A \rightarrow (A \wedge B)$$

- Parênteses mais externos podem ser omitidos:

$$A \rightarrow (B \rightarrow C) \equiv (A \rightarrow (B \rightarrow C))$$

II-2. Semântica da Lógica Clássica Proposicional

A semântica da lógica clássica proposicional consiste na atribuição de significado às fórmulas da linguagem.
Isto é feito através da atribuição de valor verdade.
Para cada fórmula é atribuído um valor verdadeiro ou falso.

valores-verdade: V - verdadeiro
F - falso

O valor verdade de uma fórmula depende unicamente dos valores verdade atribuídos aos seus símbolos proposicionais.

Tabela Verdade

Conjunção:

A	**B**	**$A \wedge B$**
V	V	V
V	F	F
F	V	F
F	F	F

Hoje tem aula **e** hoje é quinta-feira.

Disjunção (não-exclusiva):

A	**B**	**$A \vee B$**
V	V	V
V	F	V
F	V	V
F	F	F

Hoje tem aula **ou** hoje é quinta-feira.

Negação:

A	**$\neg A$**
V	F
F	V

Hoje **não** tem aula.

Implicação:

Exemplo: Considere o anúncio abaixo:

Para pagamento à vista, nós damos 25% de desconto na compra de qualquer TV.

<u>Em linguagem lógica</u>: "*Se* pagamento à vista *então* 25% de desconto."
Suponha agora que D. Maria deseja verificar se este anúncio é honesto ou não.

Possibilidade	Pagam$^{\underline{to}}$ à vista	25% de desconto	Anúncio
1	V	V	V
2	V	F	F
3	F	V	V
4	F	F	?

A	**B**	**A→B**
V	V	V
V	F	F
F	V	V
F	F	V (?)

◆ Existem lógicas que discordam da linha 4
Ex: 3-valores, intuicionista, relevante...

Exercício:

Construa a tabela verdade de: $(\neg A \vee B) \rightarrow C$

A	**B**	**C**	**¬ A**	**(¬ A ∨ B)**	**(¬ A∨B)→C**
V	V	V			
V	V	F			
V	F	V			
V	F	F			
F	V	V			
F	V	F			
F	F	V			
F	F	F			

↳Mais formalmente:

A cada símbolo proposicional nós queremos atribuir um valor verdadeiro ou falso. Isto é feito através de uma função **v** de atribuição de valor verdade.

v: P → {V,F}, onde P é conjunto dos símbolos proposicionais

<u>Exemplos</u>:

v(A) = F, v(B) = V, v(C) = V

- Uma vez atribuído valor verdade a cada símbolo proposicional em P, queremos estender esta atribuição para o conjunto de todas as fórmulas da linguagem proposicional LP, que denotaremos por W.
- Definimos uma função *v* de atribuição de valor verdade a fórmulas da linguagem LP como uma extensão da função v tal que:

v: W → {V,F}, onde *v* deve satisfazer as seguintes condições:

1. $v(A) = \mathrm{v}(A)$, se $A \in P$

2. $v(\neg \alpha) =$ V, se $v(\alpha) =$ F
   F, se $v(\alpha) =$ V

3. $v(\alpha \wedge \beta) =$ V, se $v(\alpha) = v(\beta) =$ V
   F, caso contrário

4. $v(\alpha \vee \beta) =$ F, se $v(\alpha) = v(\beta) =$ F
   V, caso contrário

5. $v(\alpha \rightarrow \beta) =$ F, se $v(\alpha) =$ V e $v(\beta) =$ F
   V, caso contrário

Exemplo 1: $(A \rightarrow (B \vee \neg C))$

v(A) = V v(B) = v(C) = F

$v(A \rightarrow (B \vee \neg C)) = ?$ V

$v(A) =$ V $\quad v(B \vee \neg C) = ?$ V

$v(B) = ?$ F $\quad v(\neg C) = ?$ V

v(B) = F $\quad v(C) = ?$ F

v(C) = F

Exemplo 2: $v(\,(A \wedge \neg D) \rightarrow (\neg C \vee B)\,) = ?$

$v(A) = F$, $v(C) = V$,
$v(B) = F$, $v(D) = V$

$v(\,(A \wedge \neg D) \rightarrow (\neg C \vee B)\,) = ?\ V$

$v(A \wedge \neg D) = ?\ F$ $v(\neg C \vee B) = ?\ F$

$v(A) = ?\ F$ $v(\neg D) = ?\ F$ $v(\neg C) = ?\ F$ $v(B) = ?\ F$

$v(A) = F$ $v(D) = ?\ V$ $v(C) = ?\ V$ $v(B) = F$

$v(D) = V$ $v(C) = V$

✎ Quantas linhas possui uma tabela verdade para $(A \wedge \neg D) \rightarrow (\neg C \vee B)$?

Cada linha corresponde a uma possível atribuição de valores verdade aos símbolos proposicionais que compõe a fórmula. Como esta fórmula possui 4 símbolos proposicionais (A,B,C e D), sua tabela verdade deve ter $2^4 = 16$ linhas.

- T.V. computa o valor verdade de uma fórmula para todas as possíveis atribuições v a seus símbolos proposicionais.

- Logo, o problema de se saber o valor verdade de uma fórmula na lógica clássica proposicional é DECIDÍVEL; o algoritmo é o seguinte:

passo 1: conte o número de símbolos proposicionais;
passo2: monte uma tabela com 2^n linhas e com quantas colunas for o número de subfórmulas de α;
passo 3: preencha as colunas dos símbolos proposicionais com V ou F alternando de cima para baixo TFTF para a 1ª coluna, TTFF... para a 2ª, TTTTFFFF para a 3ª e assim por diante, nas potências de 2.
passo 4: compute o valor verdade das outras colunas usando as tabelas básicas fornecidas.

Exemplo: $(\neg A \rightarrow B) \vee C$

$2^3 = 8$

A	B	C	¬ A	(¬ A → B)	(¬ A → B) ∨ C
V	V	V	F	V	V
F	V	V	V	V	V
V	F	V	F	V	V
F	F	V	V	F	V
V	V	F	F	V	V
F	V	F	V	V	V
V	F	F	F	V	V
F	F	F	V	F	F

- Existem fórmulas onde todas as linhas da T.V. dão verdade.
- Elas são verdadeiras não importando os valores verdade que atribuímos aos seus símbolos proposicionais.
- Estas fórmulas são chamadas **tautologias**.
- Da mesma forma, existem fórmulas que são sempre falsas, independente dos valores verdade atribuídos aos seus símbolos proposicionais. Estas são chamadas **contradições**.
- Além disso, existem fórmulas que, embora diferentes, têm tabelas verdade que coincidem linha a linha. Tais fórmulas são ditas **equivalentes**.

Exemplos:

A	A→A
V	V
F	V

$A \rightarrow A$ é uma tautologia.

A	**B**	**B→A**	**A → (B → A)**
V	V	V	V
V	F	V	V
F	V	F	V
F	F	V	V

A → (B→A) é uma tautologia.

A	**B**	**(A ∨ B)**	**¬ (A ∨ B)**	**A ∧ ¬ (A ∨ B)**
V	V	V	F	F
V	F	V	F	F
F	V	V	F	F
F	F	F	V	F

A ∧ (¬ (A ∨ B)) é uma contradição.

A	**B**	**A ∧ B**	**¬ (A ∧ B)**
V	V	V	F
V	F	F	V
F	V	F	V
F	F	F	V

A	**B**	**¬A**	**¬B**	**¬A ∨ ¬B**
V	V	F	F	F
V	F	F	V	V
F	V	V	F	V
F	F	V	V	V

¬ (A ∧ B) é equivalente a ¬A ∨ ¬B.

Definição:

a) Uma fórmula α é uma tautologia se e somente se, para toda atribuição v, $v(\alpha) = V$.

b) Uma fórmula α é uma contradição se e somente se, para toda atribuição v, $v(\alpha) = F$.

Exemplos de tautologias "famosas":

- $A \vee \neg A$
- $A \to A$
- $(A \to ((A \to B) \to B)$
- $A \wedge B \to A$
- $A \wedge B \to B$
- $\neg \neg A \to A$
- $A \to A \vee B$

- $B \to A \vee B$
- $((A \to B) \wedge \neg B) \to \neg A$

Exemplos de contradições:

- $A \wedge \neg A$
- $\neg (A \to A)$
- $A \wedge (A \to B) \wedge \neg B$

Exercício:

Verificar se estas fórmulas são realmente tautologias e contradições.

<u>Definição:</u>

Duas fórmulas são ditas *equivalentes* se e somente se elas têm a mesma tabela verdade, isto é, $\alpha \equiv \beta$ (ou $\alpha \Leftrightarrow \beta$) se e somente se, para toda atribuição v, $v(\alpha) = v(\beta)$.

<u>Exemplos de equivalências</u>:

- $\neg \neg A \equiv A$
- $\neg (A \vee B) \equiv \neg A \wedge \neg B$

Exercício:

Verificar se as seguintes fórmulas são equivalentes:

$\neg (A \wedge B) \equiv \neg A \vee \neg B$

$\neg (P \to Q) \equiv (P \wedge \neg Q)$

$P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$

$P \to Q \equiv \neg Q \to \neg P$

$P \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R)$

<u>Observação</u>:

Utilizando a noção de equivalência, é possível definir alguns de nossos conectivos a partir de outros. Desse modo, mostraremos que uma linguagem contendo apenas a negação ($\neg$) e mais um conectivo qualquer ($\wedge$, $\vee$ ou $\to$) permite expressar exatamente o mesmo conjunto de proposições que a linguagem da lógica proposicional completa. Assim:

- Definir $\to$ e $\wedge$ usando $\neg$ e $\vee$

  $P \to Q \equiv \neg P \vee Q$

  $P \wedge Q \equiv \neg (\neg P \vee \neg Q)$

- Definir $\to$ e $\vee$ usando $\neg$ e $\wedge$

  $P \to Q \equiv \neg (P \wedge \neg Q)$

  $P \vee Q \equiv \neg (\neg P \wedge \neg Q)$

◆ Definir $\wedge$ e $\vee$ usando $\neg$ e $\rightarrow$

$P \wedge Q \equiv \neg (P \rightarrow \neg Q)$

$P \vee Q \equiv \neg P \rightarrow Q$

Exercício:

Verificar as equivalências acima.

Exercício: (Sheffer Stroke P/Q)

Suponha P/Q é uma fórmula com a seguinte T.V.:

P	**Q**	**P/Q**
V	V	F
V	F	F
F	V	F
F	F	V

Defina $\wedge$, $\vee$, $\neg$, $\rightarrow$ usando /

Dica: $\neg P \equiv (P/P)$

P	**¬ P**	**P/P**
V	F	F
F	V	V

⇘ Dada uma tabela verdade, como saber a fórmula correspondente?

P	**Q**	**R**	**S**	**α**
V	V	V	V	V
F	V	V	V	V
V	F	V	V	V
F	F	V	V	F
V	V	F	V	F
F	V	F	V	F
V	F	F	V	F
F	F	F	V	F
V	V	V	F	F
F	V	V	F	F
V	F	V	F	F
F	F	V	F	F
V	V	F	F	F
F	V	F	F	F
V	F	F	F	F
F	F	F	F	F

⇘ Como achar α?

Linhas em que α é verdadeira:

P	**Q**	**R**	**S**	**α**	
V	V	V	V	V	$P \wedge Q \wedge R \wedge S$
F	V	V	V	V	$\neg P \wedge Q \wedge R \wedge S$
V	F	V	V	V	$P \wedge \neg Q \wedge R \wedge S$

$(P \wedge Q \wedge R \wedge S) \vee (\neg P \wedge Q \wedge R \wedge S) \vee (P \wedge \neg Q \wedge R \wedge S)$

Formalizando:

Para achar a fórmula correspondente a uma tabela verdade, procedemos da seguinte maneira:

1. Para cada linha da tabela verdade em que a fórmula α é verdadeira, escrevemos uma fórmula correspondendo à conjunção (E) das fórmulas atômicas que forem verdadeiras e das negações daquelas que forem falsas na linha considerada.
2. A fórmula α é a disjunção (OU) das fórmulas escritas no passo 1.

<u>Definição:</u>

1. Uma fórmula atômica ou átomo é qualquer símbolo proposicional.
   Ex: A, B, C.

2. Um literal é um átomo ou sua negação.
   Ex: A, $\neg$ A, B, $\neg$ C.

<u>Definição:</u>

a) <u>Forma Normal Disjuntiva</u> (FND):

Uma fórmula α está na forma normal disjuntiva se e somente se α tem a seguinte forma:

$$\alpha = C_1 \vee C_2 \vee \ldots \vee C_n$$

onde cada $C_i = (Q_1 \wedge Q_2 \wedge \ldots \wedge Q_m)$, $1 \leq i \leq n$, e Qj, $1 \leq j \leq m$, são literais, isto é, cada C_i é uma conjunção de literais.

b) <u>Forma Normal Conjuntiva</u> (FNC):

Uma fórmula α está na forma normal conjuntiva se e somente se α tem a seguinte forma:

$$\alpha = D_1 \wedge D_2 \wedge \ldots \wedge D_n$$

onde cada $D_i = (Q_1 \vee Q_2 \vee \ldots \vee Q_m)$, $1 \leq i \leq n$ e Qj, $1 \leq j \leq m$, são literais, isto é, cada D_i é uma conjunção de literais.

ALGORITMO: Converter fórmulas para a FNC:

passo 1: elimine o conectivo $\rightarrow$ usando:

$$A \rightarrow B \equiv (\neg A \vee B)$$
$$\neg (A \rightarrow B) \equiv (A \wedge \neg B)$$

passo 2: mova a negação ($\neg$) para o interior da fórmula, usando as seguintes regras:

$$\neg \neg A \equiv A$$
$$\neg (A \wedge B) \equiv (\neg A \vee \neg B)$$
$$\neg (A \vee B) \equiv (\neg A \wedge \neg B)$$

passo 3: distribua as definições sobre conjunções usando:

$$A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$$
$$(A \wedge B) \vee C \equiv (A \vee C) \wedge (B \vee C)$$

Exemplo:

$(A \wedge B) \rightarrow C$

$\equiv \neg (A \wedge B) \vee C$

$\equiv (\neg A \vee \neg B \vee C)$ ⇨ FNC

Exercício:

Fazer um algoritmo para converter fórmulas para a forma normal disjuntiva.

<u>Definição:</u>

Seja α uma fórmula e Γ um conjunto de fórmulas:

1. Uma atribuição de valor verdade v, $P \rightarrow \{V, F\}$ <u>satisfaz</u> α se e somente se $v(\alpha)$= V. E v <u>satisfaz</u> Γ se e somente se v satisfaz cada membro de Γ.
2. Γ é <u>satisfatível</u> se e somente se existe uma atribuição v que satisfaz Γ. Caso contrário, Γ é <u>insatisfatível</u>.

<u>Definição:</u>

Um ***argumento*** é uma seqüência de fórmulas $\alpha_1, \alpha_2, ..., \alpha_n, \beta$ onde α_i, $1 \leq i \leq n$, são as premissas (banco de dados) e β é a conclusão (pergunta).

$$\begin{array}{c} \alpha 1 \\ \alpha 2 \\ . \\ . \\ \underline{\alpha n} \\ \beta \end{array}$$

Exemplo:

João está em casa ou no trabalho.
João não está no trabalho.

C: João está em casa.
T: João está no trabalho.

$$\begin{array}{c} C \vee T \\ \underline{\quad \neg T \quad} \\ C \end{array}$$

Colocando numa notação horizontal:

$C \vee T, \neg T \models C$

Um argumento pode ser logicamente válido ou inválido.

Definição:

Um conjunto de fórmulas α_1, α_2, ..., α_n ***implica logicamente*** numa fórmula β**,** $\alpha 1$, $\alpha 2$, ..., $\alpha n \models \beta$, se somente se para toda valoração v se $v(\alpha 1 \wedge \alpha 2 \wedge ... \wedge \alpha n)$= V, então $v(\beta)$= V.

Definição:

Um argumento $\alpha 1$, $\alpha 2$, ..., αn, β é ***válido*** se somente se $\alpha 1$, $\alpha 2$, ..., αn ***implica logicamente*** em β, isto é, $\alpha 1, \alpha 2, ..., \alpha n \models \beta$.

TEOREMA:

Um argumento α_1, α_2, ..., α_n, β é logicamente válido, ou seja, $\alpha_1, \alpha_2, ..., \alpha_n \models \beta$ se somente se $(\alpha_1 \wedge \alpha_2 \wedge ... \wedge \alpha_n) \rightarrow \beta$ for uma tautologia.

Exemplo: (baseado no exemplo anterior)

$(C \vee T) \wedge \neg T \rightarrow C$ é tautologia $\Rightarrow$ $(C \vee T) \wedge \neg T \models C$

Complexidade

Dois problemas distintos:

Problema 1: Dada uma fórmula φ com comprimento n e uma valoração v para os símbolos proposicionais. Calcular $\nu(\varphi,v)$.

Onde o comprimento de uma fórmula é o número de símbolos da fórmula, i.e., número de símbolos proposicionais + número de conectivos lógicos.

Função $\nu(\varphi,v)$: **booleno**

caso φ = P onde P é um símbolo proposicional, **retorna** $\nu(P)$;
= $\neg\varphi_1$, **retornar NOT** $\nu(\varphi_1,v)$;
= $\varphi_1 \wedge \varphi_2$, **retornar** $\nu(\varphi_1,v)$ **AND** $\nu(\varphi_2,v)$;
= $\varphi_1 \vee \varphi_2$, **retornar** $\nu(\varphi_1,v)$ **OR** $\nu(\varphi_2,v)$;
= $\varphi_1 \rightarrow \varphi_2$, **retornar NOT** $\nu(\varphi_1,v)$ **OR** $\nu(\varphi_2,v)$;

Complexidade da função $\nu(\varphi,v)$ é O(n), pois a função é chamada uma vez para cada símbolo proposicional e uma vez para cada conectivo lógico.

Problema 2: Dada uma fórmula φ com comprimento n e m símbolos proposicionais.
Verificar se existe alguma valoração que satisfaz φ.

Função SAT(φ): **booleno**

para cada valoração v **faça**
    se ν(φ,v) **então retorna verdadeiro**
retorna falso

Complexidade da função SAT(φ)

número de valorações diferentes X complexidade de ν(φ,v) ˜ $O(2^m)$ X O(n) ˜ $O(2^m . n)$

Obs. 1) problema 1 é polinomial no comprimento da fórmula;

2) problema 2 é NP completo.

II-3. Sistemas Dedutivos

Resumo: • linguagem
• semântica

BD $\alpha_1 \ldots \alpha_n$	BD ℘ q

◆ Queremos fazer perguntas q a um banco de dados BD.
◆ Nós já temos um algoritmo para responder BD∅ q montando a tabela verdade para $\alpha_1 \wedge \alpha_2 \wedge \ldots \wedge \alpha_n \rightarrow q$. Se for uma tautologia responde SIM, senão responde NÃO.
◆ A complexidade deste algoritmo é da ordem de 2^n , onde n é o número de símbolos proposicionais em $\alpha_1 \wedge \alpha_2 \wedge \ldots \wedge \alpha_n \rightarrow q$.

II-3.1. Dedução Natural

◆ Jakowski(1924) e Gentzen(1935)
◆ Somente regras de inferência
◆ Para cada conectivo lógico temos 2 regras:
  ⇒ Regra de Introdução
  ⇒ Regra de Eliminação
◆ <u>Regra de Introdução</u>: como uma fórmula contendo o conectivo pode ser inferida
◆ <u>Regra de Eliminação</u>: que conseqüências podemos tirar de uma fórmula contendo o conectivo
◆ Queremos escrever regras de inferência que sejam válidas, isto é, que as premissas <u>impliquem logicamente</u> nas conclusões.

Regras de inferência

$$\frac{P^1, P^2, \ldots, P^n}{C}$$

Se as premissas $P^1, P^2, \ldots, P^n$ forem verdadeiras, então a conclusão C é verdadeira.
$P^1, P^2, \ldots, P^n$ ℘ C

↳ $\dfrac{\alpha}{\alpha \wedge \neg \alpha}$ é uma regra válida? Não, pois α não implica logicamente $\alpha \wedge \neg\alpha$, visto que esta fórmula é uma contradição.

Regras de Inferência:

<u>Conjunção $\wedge$</u>

$\wedge$-I $$\frac{\alpha \quad \beta}{\alpha \wedge \beta}$$

Se $\alpha \in BD$ e $\beta \in BD$, então responda sim para $\alpha \wedge \beta$. BD $\wp$ $\alpha \wedge \beta$.

➩ $\wedge$ I é válida? Se $v(\alpha) = v(\beta) = V$ então $v(\alpha \wedge \beta) = V$

$\wedge$-E $$\frac{\alpha \wedge \beta}{\alpha} \qquad \frac{\alpha \wedge \beta}{\beta}$$

➩ $\wedge$ E é válida? $v(\alpha \wedge \beta) = V$ então $v(\alpha) = v(\beta) = V$

<u>Disjunção $\vee$</u>

$\vee$-I $$\frac{\alpha}{\alpha \vee \beta} \qquad \frac{\beta}{\alpha \vee \beta}$$

➩ É válida? Se $v(\alpha) = V$ então $v(\alpha \vee \beta) = V$

$\vee$-E$_1$ $$\frac{\alpha \vee \beta, \neg \alpha}{\beta}$$ **$\vee$-E$_2$** $$\frac{\alpha \vee \beta, \neg \beta}{\alpha}$$

➩ É válida? Se $v(\alpha \vee \beta) = V$ e $v(\neg \alpha) = V$ então $v(\alpha) = F$ e $v(\beta) = V$

$\vee$-E$_3$ $$\frac{\alpha \vee \beta \quad \alpha \rightarrow \sigma \quad \beta \rightarrow \sigma}{\sigma}$$

➩ É válida? Se $v(\alpha \vee \beta) = V$ então temos 2 casos:

1. $v(\alpha) = V$
   Como $\alpha \rightarrow \sigma$, então $v(\sigma) = V$
2. $v(\beta) = V$
   Como $\beta \rightarrow \sigma$, então $v(\sigma) = V$

Exemplo:

Hoje é terça-feira $\vee$ Hoje é quinta-feira.
Se Hoje é terça-feira *então* aula de lógica.
<u>*Se* Hoje é quinta-feira *então* aula de lógica.</u>
aula de lógica

Implicação →

$\to$**E** $\dfrac{\alpha \quad \alpha \to \beta}{\beta}$ Se $v(\alpha)$ = V então $v(\beta)$ = V para que $v(\alpha \to \beta)$ = V

Exemplo:

quinta-feira
se quinta-feira *então* aula de lógica
aula de lógica

$\to$**I**

$$\dfrac{\boxed{\begin{array}{c}[\alpha]^1 \\ \vdots \\ \beta\end{array}}}{\alpha \to \beta^1}$$

Onde $[\alpha]$ é uma suposição

Exemplo:

BD: eq. Newton Se solto objeto então objeto cai.

eq. Newton Suponho que solto objeto Objeto cai? SIM

Se solto objeto então objeto cai

$\to$**I**$_1$ $\dfrac{\neg \alpha}{\alpha \to \beta}$ $\to$**I**$_2$ $\dfrac{\beta}{\alpha \to \beta}$.

Exemplo 1:

1. $A \wedge B$
2. $A \to C$
3. $C \to D$

BD $\wp$ D

4. A $\wedge$E(1)
5. C $\to$E(2,4)
6. D $\to$E(3,5)

Exemplo 2:

1. $A \rightarrow C$ 2. $C \rightarrow D$	BD ℘ $A \rightarrow D$
3. $[A]^1$	
4. C	$\rightarrow$E(3,1)
5. D	$\rightarrow$E(4,2)
6. $A \rightarrow D^1$	$\rightarrow$I(3,5)

Negação ¬

¬ - I $\dfrac{\alpha}{\neg\neg\alpha}$ **¬ - E1** $\dfrac{\neg\neg\alpha}{\alpha}$ ***Redução ao Absurdo*: $[\neg\alpha]^1$**

$$\begin{array}{c} [\neg\alpha]^1 \\ \ldots \\ \underline{\alpha} \\ \alpha^1 \end{array}$$

¬ - E2 $\dfrac{\beta \quad \neg\beta}{\alpha}$

↳Resumindo:

◆ CONJUNÇÃO:

∧- I $\dfrac{\alpha \quad \beta}{\alpha \wedge \beta}$ **∨ - E** $\dfrac{\alpha \wedge \beta \quad \alpha \wedge \beta}{\alpha \qquad\qquad \alpha}$

◆ DISJUNÇÃO:

∨ - I $\dfrac{\alpha}{\alpha\vee\beta}$ $\dfrac{\beta}{\alpha\vee\beta}$

∨ - E1 $\dfrac{\alpha\vee\beta, \neg\alpha}{\beta}$

∨ - E2 $\dfrac{\alpha\vee\beta, \neg\beta}{\alpha}$

∨ - E3

$$\begin{array}{ccc} & [\alpha]^1 & [\beta]^2 \\ & . & . \\ & . & . \\ & . & . \\ \hline \alpha\vee\beta & \sigma & \sigma \\ \multicolumn{3}{c}{\sigma^{1,2}} \end{array}$$

◆ IMPLICAÇÃO:

→E $\dfrac{\alpha \quad \alpha\rightarrow\beta}{\beta}$ **→I**

$$\begin{array}{c} [\alpha]^1 \\ \vdots \\ \beta \ . \\ \hline \alpha\rightarrow\beta^1 \end{array}$$

→I1 $\underline{\neg\alpha}$ **→I2** $\underline{\beta}$.

$\alpha \rightarrow \beta$ $\qquad\qquad\qquad\qquad$ $\alpha \rightarrow \beta$

◆ NEGAÇÃO:

¬ - I $\dfrac{\alpha}{\neg\neg\alpha}$ $\qquad$ **¬ - E₁** → **¬ - E**$_1$ $\dfrac{\neg\neg\alpha}{\alpha}$ $\qquad$ ***Redução ao Absurdo***: $\begin{array}{c}[\neg\alpha]^1 \\ \ldots \\ \alpha \\ \hline \alpha^1\end{array}$

¬ - E$_2$ $\dfrac{\beta \;\neg\beta}{\alpha}$

Exemplo:

BD = $\{A \wedge B\}$ $\qquad$ BD $\wp$ $A \vee B$?

1. $A \wedge B$
2. A $\qquad \wedge E_1$ (1)
3. $A \vee B$ $\quad \vee I$ (2)

Definição:

1. Uma ***prova*** em dedução natural de uma formula φ a partir de um conjunto de fórmulas BD=$\{\varphi 1, \varphi 2, \ldots, \varphi k\}$ é uma sequência de fórmulas rotuladas da seguinte forma:

   i as fórmulas do BD formam o prefixo da sequência e são rotuladas 1: $\varphi 1$, 2: $\varphi 2$, ..., k: φk;

   ii se a última fórmula da sequência é ρ.r: φi (onde ρ pode ser a sequência vazia), então a próxima fórmula rotulada será:

   *ii.*1 ρ.r.1: φj se φi é uma suposição;

   *ii.*2 ρ.r+1: φj se φj foi obtida pela aplicação de regras do grupo I a fórmulas no escopo igual superior a σ, onde $\rho = \sigma$.r;

   *ii.*3 σ.s+1: φj se φj foi obtida pela aplicação das regras do grupo II e $\rho = \sigma$.s;

   iv n: φ é a última fórmula da seqüência.

1. A fórmula φ é dita um ***teorema do conjunto de fórmulas*** BD, BD $\wp$ φ.

4. A fórmula φ é dita um teorema lógico se BD é vazio.

OBS: Uma prova pode ser chamada algumas vezes de derivação.

Definição:

1. Um conjunto de fórmulas $\Gamma = \{\alpha 1, \alpha 2, \ldots, \alpha n\}$ é **in*consistente*** se somente se $\Gamma \wp$ $\beta \wedge \neg\beta$ para alguma fórmula β.
2. Um conjunto de fórmulas $\Gamma = \{\alpha 1, \alpha 2, \ldots, \alpha n\}$ é ***consistente*** se somente se ele não é inconsistente.

Notação:

Nós abreviaremos:

$\Gamma \cup \alpha \wp \beta$ por $\Gamma, \alpha \wp \beta$

$\varnothing \wp \alpha$ por $\wp \alpha$

Um teorema muito importante da lógica proposicional é o chamado teorema da dedução.

<u>(META) TEOREMA DA DEDUÇÃO:</u>

Seja Γ um conjunto de fórmulas e α e β, fórmulas. Então:

Se $\Gamma, \alpha \wp \beta$ então $\Gamma \wp \alpha \rightarrow \beta$.

<u>Prova do (Meta)Teorema da Dedução:</u>

Suponha $\Gamma = \{\alpha_1, \alpha_2, \ldots, \alpha_n\}$.

BD [Γ [α_1, α_2, ..., α_n]]

Prova Π [α, ...]

β

Reescrever

$\Rightarrow$

BD [α_1, Γ [α_2, ..., α_n]]

Prova Π: [$[\alpha]^1$, ...]

$$\frac{\beta}{\alpha \rightarrow \beta^1}$$

Se Γ (conjunto de fórmulas) e α (fórmula) pertencem ao um banco de dados (BD) e a partir desse BD temos uma prova Π e chegamos a β aplicando as regras do Cálculo Dedutivo. Então se dado um BD = Γ, consegue-se provar que a partir de Γ , α implica em β. Para esta prova , basta supor A e aplicar as mesmas regras de dedução da prova anterior (Prova Π). Dessa forma chega-se em β, e fechamos a suposição feita fazendo $\alpha \rightarrow \beta$.

META TEOREMA 2:

Se $\Gamma \wp A \rightarrow B$ então $\Gamma, A \wp B$.

BD [Γ [A_1, A_2, ..., A_n]

Prova Π [...

$\overline{}$
$A \rightarrow B$

Reescrever $\Rightarrow$

BD [Γ [A_1, A_2, ..., A_n] A]

Prova Π [...

$\underline{A \rightarrow B}$
B

Se a partir de um BD formado por um conjunto de fórmulas Γ chegamos em $\alpha \rightarrow \beta$ através de uma prova Π, siginifica que se supormos α dentro desse BD, podemos chegar até β através do Cálculo Dedutivo.

Portanto, se α pertencer ao BD e usarmos o Cálculo Dedutivo, conseguimos chegar até β.

Exercícios:

Prove por dedução natural as seguintes afirmações:

1. BD = $\{A \rightarrow B \vee C, A \rightarrow \neg B, C \rightarrow \neg D\}$; BD $\wp$ $A \rightarrow \neg D$?
2. $P \wedge Q \wp Q \wedge P$?
3. $\wp (P \wedge Q) \rightarrow (Q \wedge P)$?
4. $Q \wp P \rightarrow Q$?
5. $P \rightarrow Q \wedge R \wp (P \rightarrow Q) \wedge (P \rightarrow R)$?
6. $P \rightarrow (Q \wedge R) \wp (P \wedge Q) \rightarrow R$?
7. $\neg \alpha \wedge \neg \beta \wp \neg (\alpha \vee \beta)$?
8. $P \vee Q \rightarrow R \wp P \rightarrow (Q \rightarrow R)$?
9. $B, R \vee S \rightarrow A, R \vee S, A \wedge R \rightarrow C, B \wedge S \rightarrow C \wp C$?
10. $(P \rightarrow Q) \wedge (Q \rightarrow R) \wp P \rightarrow R$?
11. $P \rightarrow R, Q \rightarrow S \wp P \wedge Q \rightarrow R \wedge S$?
12. $A \wedge B, (A \vee B) \rightarrow (R \rightarrow S), (P \rightarrow Q) \rightarrow R, A \wedge P \rightarrow Q \wp S$?
13. $P \rightarrow Q, \neg Q \wp \neg P$?
14. $(P \rightarrow Q) \rightarrow Q, Q \rightarrow P \wp P$?
15. $\wp \neg (\alpha \vee \beta) \rightarrow \neg \alpha \wedge \neg \beta$?

II-3.2. - Provador de Dov Gabbay

Um provador automático de teoremas dirigido pelo objetivo.

- **Objetivo:** Dado um banco de dados e uma pergunta (objetivo) p, nós queremos responder SIM ou NÃO para a pergunta p.

- **2 Métodos:**
  - Tabela Verdade BD|= p (semanticamente);
  - Regras de Dedução Natural BD $\wp$ p (sintaticamente).

  Tabela Verdade é mecânico mas **pouco eficiente**.
  Dedução Natural requer **criatividade**.

- **Nosso objetivo:** Responder BD $\wp$ p automaticamente.

Provador Automático de Teoremas Dirijido pelo Objetivo (pergunta) :

- **Linguagem:**
  Linguagem da lógica clássica proposicional + o símbolo $\perp$ (absurdo)

Definição: *Cláusula, Objetivo e Banco de Dados.*

(*i*) Qualquer átomo (símbolo proposicional) , incluindo $\perp$, é uma ***cláusula*** e também um ***objetivo.***
(*ii*) Se G é um objetivo e q um átomo, então $G \rightarrow q$ é uma ***cláusula*** e também um ***objetivo.***
(*iii*) Se G_1 e G_2 são objetivos, então $G_1 \wedge G_2$ é também um ***objetivo.***
(*vi*) Um banco de dados BD é um conjunto de cláusulas.
(*v*) Nenhuma fórmula é uma cláusula a não ser as definidas em (i),(ii),(iii) e (vi).

Transformação de fórmulas em Cláusulas e Objetivos:

1. $\neg\alpha \equiv (\alpha \rightarrow \perp)$
2. $\alpha \vee \beta \equiv (\alpha \rightarrow \perp) \rightarrow \beta$
3. $\alpha \rightarrow (\beta \rightarrow \sigma) \equiv (\alpha \wedge \beta) \rightarrow \sigma$
4. $\alpha \rightarrow (\beta \wedge \sigma) \equiv (\alpha \rightarrow \beta) \wedge (\alpha \rightarrow \sigma)$
5. Se na transformação do BD obtenho $\alpha_1 \wedge \alpha_2 \wedge ... \wedge \alpha_n$, nós colocamos α_1, α_2, ..., α_n no BD.

Exemplo:

$\neg A \vee B \rightarrow B \wedge C$

$(A\rightarrow\bot) \vee B \rightarrow B \wedge C$ (Regra 1)

$(A\rightarrow\bot)\rightarrow\bot) \rightarrow B \rightarrow B\wedge C$ (Regra 2)

$(((A\rightarrow\bot)\rightarrow\bot) \rightarrow B \rightarrow B) \wedge (((A\rightarrow\bot)\rightarrow\bot) \rightarrow B \rightarrow C)$ (Regra 4)

TEOREMA:

Toda fórmula da lógica proposicional é equivalente (pode ser traduzida) a uma conjunção de cláusulas.

Problema: BD $\wp$ G?

Problema Transformado: BDo $\wp$ Go?

Regras de Computação:

1. Provar BD $\wp$ $\alpha\wedge\beta$, prove BD $\wp$ α e BDo $\wp$ β.

2. Provar BD $\wp$ $\alpha \rightarrow\beta$, prove BD,α $\wp$ β. Se α for uma conjunção $\alpha=\alpha_1 \wedge \alpha_2 \wedge\ldots\wedge \alpha_n$, adicione a BD $\alpha_1, \alpha_2,\ldots, \alpha_n$.

3. Provar BD $\wp$ q, onde q é um átomo, temos 4 casos:
   - 3.1. $q \in$ BD, responda SIM.
   - 3.2. $\bot \in$ BD, responda SIM.
   - 3.3. $\sigma \rightarrow q \in$ BD, pergunte BD $\wp$ σ.
   - 3.4. $\sigma \rightarrow \bot \in$ BD, pergunte BD $\wp$ σ.

4. Regra do Reinício (*restart*)

   Nosso problema inicial era mostrar BD $\wp$ A. No meio da computação, nós queremos perguntar BD' $\wp$ C e continuar a prova. BD' $\wp$ C se BD' $\wp$ A, onde A é o objetivo inicial. (BD': banco de dados modificado.)

5. Checagem de LOOP:

   Nunca pergunte BD $\wp$ A pela 2ª vez para ao mesmo BD e mesmo A.

Exemplos:

(1) $(A \rightarrow B),(B \rightarrow C),A \ \wp\ C$

$(A \rightarrow B),(B \rightarrow C),A \ \wp\ B$ Regra (3.3)

$(A \rightarrow B),(B \rightarrow C),A \ \wp\ A$ Regra (3.3)

SIM ! Regra (3.1)

(2) $A \rightarrow (B \wedge C),\ D \wedge F \rightarrow A,\ D \wedge F \ \wp\ B \wedge F$

$A \rightarrow B, A \rightarrow C, (D \wedge F) \rightarrow A, D, F \ \wp\ B \wedge F$ Transformação do BD

BD $\wp$ B Regra (1)	BD $\wp$ F Regra (1)
BD $\wp$ A Regra (3.3)	SIM ! Regra(3.1)
BD $\wp$ $D \wedge F$ Regra (3.3)	

BD $\wp$ D Regra (1)	BD $\wp$ F Regra (1)
SIM ! Regra (3.1)	SIM ! Regra (3.1)

(3) BD={ (A→B) → B   C ?
A → C
B → C }

BD ℘ C

(3.3)

BD ℘ A   BD ℘ B

(3.3)

FALHA

BD ℘ A→B

(2)

BD,A ℘ B

(3.3)

BD,A ℘ A→B

(2)

BD,A ℘ B ➔ ***ENTRA EM LOOP.***
Usar a Regra do Reinício.

(4)

BD,A ℘ C

(3.3)

BD,A ℘ A

(3.1)

SUCESSO

(4) BD=$\{ (P\rightarrow Q) \rightarrow P\}$ P ?

BD $\wp$ P

(3.3)

BD $\wp$ $(P\rightarrow Q)$

(2)

BD,P $\wp$ Q ➔***FALHA***

Usar a Regra do Reinício.

(4)

BD,P $\wp$ P

(3.1)

SUCESSO

Execícios:

(1) $(A\rightarrow B)\rightarrow B \wp (B\rightarrow A)\rightarrow A$
(2) $(A\rightarrow B)\rightarrow B \wp A\vee B$
(3) $(\neg A\rightarrow A) \wp A$
(4) $(A\rightarrow B) \wp (C\vee A)\rightarrow (C\vee B)$
(5) $A \wp A\vee B$
(6) $A\rightarrow B,\neg B \wp \neg A$
(7) $\wp A\vee (A\rightarrow B)$
(8) $(A\vee B)\vee C \wp A\vee (B\vee C)$

II-3.3. Método de Tableau

As deduções são feitas por refutação, i.e., se queremos deduzir α a partir de um banco de fórmulas BD, BD $\vdash \alpha$, partimos da negação de α e tentamos chegar no absurdo.

As deduções têm forma de árvore.

Regras:

R_1 $\alpha \wedge \beta$

α

β

R_2 $\alpha \vee \beta$

α β

R_3 $\alpha \rightarrow \beta$

$\neg\alpha$ β

R_4 $\neg\neg\alpha$

α

R_5 $\neg(\alpha \wedge \beta)$

$\neg\alpha$ $\neg\beta$

R_6 $\neg(\alpha \vee \beta)$

$\neg\alpha$

$\neg\beta$

R_7 $\neg(\alpha \rightarrow \beta)$

α

$\neg\beta$

exemplo 1: $\{A \rightarrow B, B \rightarrow C\} \vdash A \rightarrow C$

$A \rightarrow B$

$B \rightarrow C$

$\neg (A \rightarrow C)$

A

$\neg C$

$\neg B$ $\quad\quad$ C

$\neg A$ $\quad\quad$ B

Cada ramo corresponde a uma valoração que satisfaz a fórmula, por isso é chamado Tableau Semântico.

Não é completo, mas é refutacionalmente completo.

Definição: Um **ramo** θ de um tableau τ é dito **fechado** se ele contiver α e $\neg\alpha$ para qualquer fórmula α.

Definição: Um **tableau** τ é dito **fechado** se cada um dos seus ramos for fechado. E aberto caso contrário.

exemplo: $A \vee B \vdash \neg(\neg A \wedge \neg B)$

$$\mathbf{A \vee B}$$

$$\neg\neg(\neg A \wedge \neg B)$$

$$(\neg A \wedge \neg B)$$

$$\neg A$$

$$\neg B$$

$$\mathbf{A} \qquad\qquad \mathbf{B}$$

Este tableau é fechado, pois todas as valorações são contraditórias, logo

$A \vee B$ e $\neg(\neg A \wedge \neg B)$ não é satisfatível.

Teorema (Completude): se BD $\models \alpha$ então existe tableau fechado para BD, $\neg\alpha$.

Teorema (Completude): se existe um tableau fechado para BD, $\neg\alpha$., então BD $\models \alpha$.

Exercícios:

1. $A \to B, \neg (A \vee B) \vdash \neg(C \to A)$
2. $(P \to Q), \neg(P \leftrightarrow Q) \vdash \neg P$
3. Guga é inteligente. Guga é determinado. Se Guga é determinado e atleta então ele não é um perdedor. Guga é atleta se ele é amante do tênis. Guga é amante do tênis se é inteligente. Guga não é um perdedor?

4. $(P \to (R \to Q)), (P \to R) \vdash (P \to Q)$

5. $\neg A \vee B, \neg(B \vee \neg C), C \to D \vdash \neg A \vee D$

II-3.4. Sistema Axiomático

- Outro sistema dedutivo.
- Mais antigo e mais utilizado para fins teóricos.
- Frege, Russel, Hulbert.

Sejam α, β e γ fórmulas quaisquer da linguagem proposicional.

Axiomas Lógicos:

- ***Implicação***:
  (1) $\alpha \to (\beta \to \alpha)$
  (2) $(\alpha \to (\beta \to \gamma)) \to ((\alpha \to \beta) \to (\alpha \to \gamma))$

- ***Conjunção:***
  (3) $(\alpha \wedge \beta) \to \alpha$
  (4) $(\alpha \wedge \beta) \to \beta$
  (5) $\alpha \to (\beta \to (\alpha \wedge \beta))$

- ***Disjunção:***
  (6) $\alpha \to \alpha \vee \beta$
  (7) $\beta \to \alpha \vee \beta$
  (8) $((\alpha \to \gamma) \wedge (\beta \to \gamma)) \to ((\alpha \vee \beta) \to \gamma)$

- ***Negação***:
  (9) $\alpha \to \neg\neg\alpha$
  (10) $\neg\neg\alpha \to \alpha$
  (11) $(\alpha \to \beta) \to ((\alpha \to \neg\beta) \to \neg\alpha)$

- ***Regra de Inferência:***
  Modus Pones (M.P.) $\dfrac{\alpha \quad \alpha \to \beta}{\beta}$

- Nosso cálculo dedutivo possui um conjunto infinito de axiomas lógicos. Para cada fórmula α, β e γ, nós temos axiomas diferentes.
- (1),...,(11) são chamadas de axiomas esquema.
- A única regra é a Modus Pones (M.P.).

Definição:

Uma fórmula α é dita um ***teorema*** de um conjunto de fórmulas Γ ($\Gamma \wp \alpha$) se e somente se existe uma seqüência de fórmulas $\alpha_1,...,\alpha_n$ tal que $\alpha_n = \alpha$ e cada α_i é:

(*i*) uma instância de um axioma esquema;

(*ii*) ou for obtida por M.P. aplicada a α_l e α_k e $l, k < i$.

(*iii*) ou um membro de Γ.

A seqüência de fórmulas $\alpha_1,...,\alpha_n$ é chamada de uma prova de α a partir de Γ.

Exemplos:

(1) $\Gamma = \{A \wedge B, A \rightarrow C\}$ $C \vee D$?

1. $A \wedge B \rightarrow A$	axioma 3
2. $A \wedge B$	Γ
3. A	M.P.(1,2)
4. $A \rightarrow C$	Γ
5. C	M.P.(3,4)
6. $C \rightarrow (C \vee D)$	axioma 6
7. $C \vee D$	M.P.(5,6)

(2) $\wp\ A \rightarrow A$

1. $A \rightarrow ((A \rightarrow A) \rightarrow A)$	axioma 1
2. $(A \rightarrow ((A \rightarrow A) \rightarrow A)) \rightarrow ((A \rightarrow (A \rightarrow A)) \rightarrow (A \rightarrow A))$	axioma 2
3. $(A \rightarrow (A \rightarrow A)) \rightarrow (A \rightarrow A)$	M.P.(1,2)
4. $A \rightarrow (A \rightarrow A)$	axioma 1
5. $A \rightarrow A$	M.P.(4,3)

Exercícios:

Provar usando o Método Axiomático:

1) $A \rightarrow B, B \rightarrow C \mid\!- A \rightarrow C$

2) $(A \vee B) \rightarrow C \mid\!- A \rightarrow (B \vee C)$

3) $A \rightarrow (B \vee C) \mid\!- (A \rightarrow B) \vee (A \rightarrow C)$

Observação:

É importante notar (e possível provar) que os três métodos dedutivos estudados para a lógica clássica proposicional são equivalentes, ou seja, uma afirmação que pode ser provada utilizando um deles, sempre poderá ser provada utilizando qualquer dos outros. Isso é importante, na medida em que nos permite provar uma determinada propriedade dos sistemas dedutivos em geral, provando-a apenas para o método axiomático, que embora difícil de ser usado na prática para provar um teorema, é bastante simples no que diz respeito à sua construção, o que facilita a demonstração de propriedades teóricas, como a completude e a corretude, que veremos na próxima seção.

II-4. Relações entre Sintaxe e Semântica

Uma das aspectos mais importantes da lógica proposicional é a maneira como a sintaxe se relaciona com a semântica.

SEMÂNTICA	**SINTAXE**
Valor verdade	prova/dedução
valida	teorema
implica logicamente $\Gamma \models \alpha$	$\Gamma \wp \alpha$
tabela verdade	cálculo dedutivo

Nós queremos relacionar o fato de uma fórmula α ser um teorema de um conjunto de fórmulas Γ ($\Gamma \wp \alpha$) com a propriedade de Γ implicar logicamente em α ($\Gamma \varnothing \alpha$).

Teorema da Corretude

“Tudo que o cálculo dedutivo prova é semanticamente válido.”

Se $\Gamma \wp \alpha$ **então** $\Gamma \varnothing \alpha$

Se uma fórmula é provada a partir de um conjunto de fórmulas então ela é logicamente implicada por este conjunto de fórmulas.
Este teorema nos assegura que tudo que provamos no sistema dedutivo é ***correto*** em relação à semântica. Isto é, nosso sistema dedutivo só prova teoremas que semanticamente estão ***corretos.***

Como se prova:
1) Prova-se que os axiomas do cálculo dedutivo são semânticamente válidos, isto é, são tautologias;
2) Prova-se que as regras de inferência sempre derivam conclusões verdadeiras a partir de premissas verdadeiras.

Teorema da Completude

“Tudo que é semânticamente válido é provado pelo cálculo dedutivo.”

Se $\Gamma \varnothing \alpha$ **então** $\Gamma \wp \alpha$

Se Γ implica logicamente em α então existe uma prova de α a partir de Γ no sistema dedutivo.
O sistema dedutivo é ***completo*** em relação à semântica pois para toda fórmula α que é logicamente implicada por Γ existe uma prova α a partir de Γ no sistema dedutivo.
Tudo que é semanticamente obtido pode ser também obtido no sistema dedutivo.

$\Gamma \varnothing \alpha \therefore \alpha$ é conseqüência lógica de Γ
$\Rightarrow$ Toda atribuição de valores verdade que satisfaz Γ também satisfaz α.

Sendo $\Gamma = \{ \alpha_1, ..., \alpha_n \}$
Quero provar que se $\emptyset\ (\alpha_1 \wedge ... \wedge \alpha_n) \rightarrow \alpha$ então $\wp\ (\alpha_1 \wedge ... \wedge \alpha_n) \rightarrow \alpha$

Como se prova:

1ª solução : Basta provar que o nosso sistema dedutivo é capaz de provar todas as tautologias.
Problema: O número de tautologias é infinito.

2ª solução: *Modelo Canônico*

Se $\Gamma\emptyset\ \alpha$ **então** $\Gamma \wp\ \alpha$

$\Leftrightarrow$

Se $\Gamma \cup \{\alpha\}$ é consistente **então** $\Gamma \cup \{\alpha\}$ é satisfatível

Prova do Modelo Canônico:

1. Suponha que se $\Gamma\emptyset\ \alpha$ então $\Gamma \wp\ \alpha$.
2. Suponha que Γ é consistente.
3. Suponha, por contradição, que Γ é insatisfatível.
4. Se Γ é insatisfatível, então, por definição, não existe nenhuma atribuição de valores verdade que seja tal que $\forall\alpha\in\Gamma$, $\upsilon(\alpha)$=V. Sendo assim, podemos dizer que $\Gamma\emptyset\ \alpha$ e $\Gamma\emptyset\ \neg\alpha$.
5. Mas isso nos permite escrever, pela suposição em (1) que $\Gamma \wp\ \alpha$ e $\Gamma \wp\ \neg\alpha$
6. A partir de (5) podemos escrever que $\Gamma \wp\ \alpha \wedge \neg\alpha$
7. Ora, (6) contradiz o fato, que havíamos suposto verdadeiro, de que Γ é consistente.

Assim, por contradição, podemos afirmar que:

Se $\Gamma\emptyset\ \alpha$ **então** $\Gamma \wp\ \alpha$

$\Rightarrow$

Se $\Gamma \cup \{\alpha\}$ é consistente **então** $\Gamma \cup \{\alpha\}$ é satisfatível

Vamos agora provar a implicação contrária:

1. Suponha que se Γ é consistente então Γ é satisfatível.
2. Suponha $\Gamma\emptyset\ \alpha$
3. Suponha, por contradição, que $\neg(\Gamma \wp\ \alpha)$
4. Então $\Gamma \cup \{\neg\alpha\}$ é consistente (já que $\neg(\Gamma \wp\ \alpha)$, não poderá $\Gamma \wp\ \alpha \wedge\neg\alpha$)
5. Então, pela suposição (1), $\Gamma \cup \{\neg\alpha\}$ é satisfatível.
6. Logo, existe uma valoração υ que satisfaz $\Gamma \cup \{\neg\alpha\}$.
7. Mas isto é uma contradição, pois por (2) υ satisfaz α também.

Observações:

- Um conjunto de fórmulas Γ é consistente se e somente se $\neg(\Gamma \;\wp\; \alpha \wedge \neg\alpha)$
- Uma valoração s é um modelo para $\Gamma = \{ \alpha_1, \dots , \alpha_n \}$ se e somente se $s(\alpha_i)=V$
  para todo $\alpha_i \in \Gamma$.
- Pelo modelo canônico, para provar a completude basta mostrar que todo conjunto consistente de fórmulas é satisfatível.

Prova do Teorema da Completude:

Dado um conjunto de fórmulas consistente Γ, nós precisamos construir uma valoração s’ e mostrar que s’ satisfaz Γ.

1º passo: Estender o conjunto consistente Γ para um conjunto Δ satisfazendo as seguintes condições:
a) $\Gamma \subseteq \Delta$
b) Δ é maximal e consistente, isto é, para toda fórmula α da linguagem, ou $\alpha \in \Delta$ ou $\neg\alpha \in \Delta$.

2º passo: Seja L a linguagem proposicional e Φ o conjunto dos símbolos proposicionais .

Vamos construir uma valoração /modelo que satisfaz Γ a partir de Δ.

$s : \Phi \rightarrow \{V,F\}$ para todo símbolo proposicional $A \in \Phi$.
$s(A) = V$ se $A \in \Delta$
$s(A) = F$ se $A \notin \Delta$

Nós podemos estender s para um s’ que valorize fórmulas,
$s’: L \rightarrow \{V,F\}$. (Como definido em aulas anteriores)

Lema da Verdade
Seja Γ um conjunto de fórmulas e α uma fórmula. $s’(\alpha) = V \Leftrightarrow \alpha \in \Delta$

Prova do Lema da Verdade:

Por indução, sobre o comprimento da fórmula, isto é, no número de símbolos lógicos nela contidos ($\neg,\vee$, etc).
a) Caso base:

$|\alpha|=0$ (Fórmula Atômica)
$\alpha \in \Phi$, $\alpha = A$
$s’(A) = s(A) = V \Leftrightarrow A \in \Delta$ (pela definição de s)

b) Hipótese de Indução: o lema vale para fórmula de tamanho $|\alpha| \leq n$.

c) Queremos mostrar que vale para $|\alpha|= n+1$

Considere $|\alpha|= n+1$
Temos então vários casos:

i) $\alpha = \neg\beta$

s'($\neg\beta$) = V sse s'(β) = F (definição de s')

sse $\beta \notin \Delta$ (pela hipótese de indução)

Logo, $\neg\beta \in \Delta$ (pois α é maximal)

Logo α é V sse $\alpha \in \Delta$.

ii) $\alpha = \beta \to \gamma$

s'($\beta \to \gamma$)= V sse s'(β)= F $\vee$ s'(γ)= V

sse $\beta \notin \Delta \vee \gamma \in \Delta$

sse $\neg\beta \in \Delta \vee \gamma \in \Delta$

sse $\Delta \wp \neg\beta \vee \gamma$

sse $\Delta \wp \beta \to \gamma$, pois $\Delta \wp (\neg\beta \vee \gamma) \leftrightarrow (\beta \to \gamma)$

sse $(\beta \to \gamma) \in \Delta$, pois Δ é consistente.

Observação: os casos iii e iv são similares e ficam como exercício.

iii) $\alpha = \beta \wedge \gamma$

iv) $\alpha = \beta \vee \gamma$

Vamos agora, a partir do lema demonstrado, provar o Teorema da Completude:

1. Suponha Γ é consistente.
2. Estenda Γ para Δ maximal e consistente.
3. Construir s e s'
4. Seja $\Gamma = \{ \alpha_1, ..., \alpha_n \}$. Como $\Gamma \subseteq \Delta$, $\alpha_i \in \Delta \Leftrightarrow$ s'(α_i)= V, para todo i (pelo "lema da verdade").
5. Logo, s' satisfaz Γ e portanto Γ é satisfatível.

III. Lógica Clássica de Primeira Ordem

$\forall x$, Emp_UFRJ(x) $\rightarrow$ Func_Pub(x)
$\forall x \, \exists y$, Suc(y,x)

- Formaliza o raciocínio dedutivo.
- Lógica Proposicional é um modelo muito restrito:

  Não podemos descrever propriedades sobre os elementos do universo de discurso: todos os elementos do domínio têm propriedade P; existem elementos do domínio que têm a propriedade P; não podemos representar relações e funções.
  Ex: Teoria dos Números < 0, suc, <, +, . , -,... >

III.1 Linguagem da Lógica Clássica de Primeira Ordem

Linguagem: alfabeto + regras gramaticais

Definição:
Um ***alfabeto de 1ª ordem*** consiste dos seguintes conjuntos de símbolos:

➔**Símbolos Lógicos:**
1. Conectivos lógicos: $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$, $\neg$, $\forall$, $\exists$.
2. Símbolos auxiliares: (e).
3. Conjunto enumerável de ***variáveis***: V = { v1,v2,...}

➔**Símbolos não Lógicos:**
4. Conjunto enumerável de ***constantes***: C = {c1,c2...}
5. Conjunto enumerável de ***símbolos de função***: F = {f1,f2,...}
   A cada símbolo funcional está associado um número inteiro n > 0, chamado de aridade.
6. Conjunto enumerável de ***símbolos predicativos*** (Predicados):
   P= {P1,P2,...}
   A cada símbolo predicativo está associado um número inteiro n > 0, chamado aridade.

Variáveis: representam elementos quaisquer do domínio.
Constantes: dão nome a elementos particulares do domínio.
Funções: representam operações sobre elementos do domínio.
Predicados: representam propriedades ou relações entre elementos do domínio.
$\forall$ ***Quantificador universal:*** "para todo elemento do domínio".
$\exists$ ***Quantificador existencial:*** "existe ao menos um indivíduo".

Exemplo:
$\forall x$ ($\exists y$ ANCESTRAL(y,x) $\wedge$ ANCESTRAL(João,José))

<u>Definição:</u>

Os **termos** da linguagem de 1ª ordem são definidos recursivamente como:

(*i*) toda variável e constante é um termo;

(*ii*) se $t_1, t_2, ..., t_n$ são termos e f um símbolo funcional de aridade n, $f(t_1,t_2,..., t_n)$ é um termo;

(*iii*) nada mais é termo.

<u>Definição:</u>

As **fórmulas** da lógica de 1ª ordem são definidas recursivamente como:

(*i*) Se P é um predicado de aridade n e $t_1, t_2, ..., t_n$ são termos, então $P(t_1,t_2,..., t_n)$ é uma fórmula chamada **fórmula atômica**;

(*ii*) Se α e β são fórmulas, então $(\neg \alpha)$, $(\alpha \wedge \beta)$, $(\alpha \vee \beta)$, $(\alpha \rightarrow \beta)$ também são fórmulas;

(*iii*) Se α é uma fórmula e x uma variável, então $\forall x\ \alpha$ e $\exists x\ \alpha$ também são fórmulas;

(*iv*) Nada mais é fórmula.

<u>Observações</u>:

1. $\alpha \leftrightarrow \beta \equiv (\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$
2. Convenções:

   (*i*) x, y, z, ... Variáveis;

   (*ii*) a, b, c, ... Constantes;

   (*iii*) f, g, h, ... Funções;

   (*iv*) A, B, C, P, U... Predicados.

3. $\neg \exists x$ GOSTA(x, collor) $\equiv \forall x \neg$ GOSTA(x, collor)

Exercícios: Nos exercícios seguintes, represente cada proposição na linguagem da lógica de primeira ordem, especificando em cada caso o significado dos símbolos não lógicos utilizados.

1. Todo cachorro é um animal.
   Todo animal morre.
   Rex é um cachorro.

2. Qualquer pessoa passando nos exames de história e ganhando na loteria é feliz. Porém qualquer pessoa que estuda ou tem sorte pode passar em todos os exames. João não estudou, mas ele tem sorte. Qualquer pessoa que tem sorte ganha na loteria. João é feliz?

3. Toda pessoa que não é pobre e é esperta é feliz. Pessoas que lêem não são burras. João sabe ler e é rico. Pessoas felizes têm vidas excitantes. Existe alguém que tem vida excitante?

4. Ninguém conquistou o mundo. Portanto, todo mundo é livre.

5. Todo elétron tem carga negativa. Nenhum pósitron tem carga negativa. Portanto, nenhum pósitron é um elétron.
6. Se Jane está doente, ela não virá trabalhar. Se ela não vier trabalhar, nenhum de nós terá nada para fazer. Assim, se Jane está doente, nenhum de nós terá nada para fazer.

Exemplo: Conselheiro Financeiro:

1. Função: ajuda a decidir se devemos investir em investimentos tipo poupança ou no mercado de ações.
2. O investimento recomendado depende do ganho mensal da pessoa e quanto ela tem em poupança.
3. Pessoas com valor inadequado de poupança devem sempre aumentar o valor da poupança como 1ª prioridade, independente do seu ganho.
4. Pessoas com valor adequado de poupança e um ganho adequado devem considerar um investimento mais arriscado, porém mais lucrativo no mercado de ações.
5. Pessoas com ganho inadequado e com poupança adequada podem considerar em dividir o valor a ser investido entre poupança e mercado de ações. Isto aumenta a poupança e tenta aumentar o ganho.
6. Poupança adequada se valor poupado maior do que 5.000 x nº de dependentes.
7. Ganho adequado se valor ganho maior do que 15.000 + (4.000 x nº de dependentes) e é estável. Um ganho instável, mesmo que grande, é sempre inadequado.

TOTAL_POUP(x): x é total na poupança.
DEPEND(y): número de dependente
minpoup(y): função que retorna o valor da poupança mínima (5.000) vezes o número de dependentes
TOTAL_GANHO(x, estável): ganho total é x e este é estável.
minganho(y): função que retorna o valor do ganho mínimo mais o acréscimo por dependente.
MAIOR(x,y): x > y
POUP(status): status é uma variável que indica a situação (adequada/inadequada) da poupança.
GANHO(status): aqui, status indica a situação (adequada/inadequada) do ganho.
INVEST(tipo): tipo é uma variável que guarda o tipo de investimento recomendado (poupança/combinado/ações)

Especificação:

1. POUP(inadequado) $\rightarrow$ INVEST(poupança)
2. POUP(adequado) $\wedge$ GANHO(inadequado) $\rightarrow$ INVEST(combinado)
3. POUP(adequado) $\wedge$ GANHO(adequado) $\rightarrow$ INVEST(ações)
4. $\forall$x TOTAL_POUP(x) $\wedge$ $\exists$y (DEPEND(y) $\wedge$ MAIOR(x, minpoup(y))) $\rightarrow$ POUP(adequado) onde minpoup(x) = 5.000x

5. $\forall x$ TOTAL_POUP(x) $\wedge$ $\exists y$ (DEPEND(y) $\wedge$ $\neg$ MAIOR(x, minpoup(y)) $\rightarrow$ POUP(inadequado)
6. $\forall x$ TOTAL_GANHO(x, estável) $\wedge$ $\exists y$ (DEPEND(y) $\wedge$ MAIOR(x, minganho(y))) $\rightarrow$ GANHO(adequado)
7. $\forall x$ TOTAL_GANHO(x, estável) $\wedge$ $\exists y$ (DEPEND(y) $\wedge$ $\neg$ MAIOR(x, minganho(y))) $\rightarrow$ GANHO(inadequado)
8. $\forall x$ TOTAL_GANHO(x, instável) $\rightarrow$ GANHO(inadequado)

<u>Entrada:</u>
9. TOTAL_POUP(22.000)
10. TOTAL_GANHO(25.000, estável)
11. DEPEND(3)

Exercício: Qual a saída produzida pela especificação acima para a entrada dada?

III.2 - Sistemas Dedutivos:

◆ Sistemas dedutivos para lógica de primeira ordem.

Definição:

Dizemos que uma variável x ocorre livre em uma fórmula α se somente se:

(*i*) α é uma fórmula atômica e x ocorre em α;

(*ii*) α é uma fórmula da forma $\beta \wedge \gamma$, $\beta \vee \gamma$, $\beta \rightarrow \gamma$ e x ocorre livre em β ou γ;

(*iii*) α é uma fórmula da forma $\neg\ \beta$ e x ocorre livre em β;

(*iv*) α é uma fórmula da forma $\forall y\beta$ ou $\exists y\beta$ e x ocorre livre em β e $x \neq y$.

Exemplos:

➯ x ocorre livre?

1. $P(x,y)$ SIM
2. $\forall y(P(x,y) \wedge Q(y,x) \rightarrow R(y))$ SIM
3. $\forall y(\forall x(P(x) \rightarrow Q(y)) \rightarrow R(x))$ SIM
4. $\forall y\ \forall z ((\forall xP(x,y) \rightarrow Q(z)) \wedge (Q(x) \rightarrow R(x,y)))$ SIM
5. $P(z,y)$ NÃO
6. $\forall y\ \exists x (P(x,y) \rightarrow Q(y))$ NÃO

Definição:

Uma fórmula α é uma sentença (ou uma fórmula fechada) se somente se α não tem nenhuma variável ocorrendo livre.

Definição:

Seja α uma fórmula, x uma variável e t um termo. Pela substituição de x por t em α ($\alpha(x/t)$) entendemos a expressão resultante da troca de todas as ocorrências livres de x por t.

Exemplos:

1. $\forall y(P(x, y,f(x, y))) \rightarrow Q(g(x), h(g(x))$ $\quad x/h(a)$
   $\forall y(P(h(a), y, f(h(a), y)) \rightarrow Q(g(h(a), h(g(h(a))))$

2. $\forall y(\forall x(Q(x, y, g(z)) \rightarrow P(f(x), y))) \rightarrow R(y(g(x)))$ $\quad x/f(z)$
   $\forall y(\forall x(Q(x, y, g(z)) \rightarrow P(f(x), y))) \rightarrow R(y(g(f(z))))$

3. $[\ \forall y(P(x, y, f(x,y))) \rightarrow Q(y,z)$ $\quad x/g(z)\ ]$ $\quad z/a$
   $\forall y\ (P(g(z), y, f(g(z), y)) \rightarrow Q(y, z)$ $\quad z/a$
   $\forall y\ (P(g(a), y, f(g(a), y)) \rightarrow Q(y, a)$

Definição:

Denotaremos por:

$$\alpha(x_1, x_2, ..., x_n) \ (x_1/t_1, x_2/t_2, ..., x_n/t_n)$$

a substituição simultânea (paralelo) de todas as ocorrências livres de $x_1, ..., x_n$ por $t_1, ..., t_n$ respectivamente.

OBS: $\alpha(x_1/t_1) ... (x_n/t_n)$ é em série da esquerda para direita.
$\alpha(x_1/t_1, x_2/t_2, ..., x_n/t_n)$ é em paralelo.

Exemplos:

1. $P(x, f(y), g(x,y)) \quad (x/a, y/b)$
   $P(a, f(b), g(a,b))$

2. $(\forall y \forall x P(x,y)) \rightarrow Q(g(x)) \wedge R(h(y)) \quad (x/f(a), y/z)$
   $(\forall y \forall x P(x,y)) \rightarrow Q(g(f(a))) \wedge R(h(z))$

3. $P(x,y,f(x,z)) \quad (x/g(z), z/y, y/a)$
   $P(g(z), a, f(g(z),y)$

4. $P(x,y,f(x,z)) \quad (x/g(z))\ (z/y)\ (y/a)$
   $P(g(z),y,f(g(z),z)) \quad (z/y)\ (y/a)$
   $P(g(y),y,f(g(y),y)) \quad (y/a)$
   $P(g(a),a,f(g(a),a))$

Definição:

Uma variável x é substituível em uma fórmula α por um termo t se, para cada variável y ocorrendo em t, não existe nenhuma subfórmula de α da forma $\forall y\beta$ ou $\exists y\beta$ onde x ocorre livre em β.

O que queremos evitar com esta condição é que o quantificador $\forall y$ ou $\exists y$ capture alguma variável de t.

Exemplo:

$(\forall y\ CHEFE(x,y) \rightarrow GERENTE(x)) \quad x/y$
$(\forall y\ CHEFE(y,y) \rightarrow GERENTE(y))$

III-2.1. Dedução Natural:

◆Regras para $\wedge,\vee,\rightarrow,\neg$ são as mesmas do caso proposicional.

◆ ***Regras do Quantificador Universal***:

$\forall$elim - $\dfrac{\forall x\alpha}{\alpha(x/t)}$ **Condição:** **x** é substituível em $\boldsymbol{\alpha}$ por **t**

$\forall$int - $\dfrac{\alpha(a)}{\forall x\alpha(x)}$ **Condição:** **a** não ocorre em nenhuma fórmula do BD e nem em nenhuma suposição em aberto.

◆***Regras para o Quantificador Existencial:***

$\exists$ int - $\dfrac{\alpha(a)}{\exists x\alpha(a/x)}$ **Leitura:** Se temos $\alpha(a)$ para um certo elemento a, então existe um x (o próprio a) tal que $\alpha(x)$, isto é, $\exists x\alpha(x)$.

$\exists$ elim -

$$\dfrac{\exists x\alpha(x) \qquad \begin{array}{c}[\alpha(x/a)] \\ \dots \\ \beta\end{array}}{\beta}$$

Condição: a não ocorre em $\exists x\alpha(x)$, β ou em qualquer suposição na qual β depende, a não ser $\alpha(x/a)$.

Leitura: Se temos $\exists x\alpha(x)$, então seja a o elemento que satisfaz $\exists x\alpha(x)$. Se supondo um a genérico nós chegamos em β é porque de $\exists x\alpha(x)$ nós temos β.

Exemplo 1: $\forall x\forall yP(x,y)\rightarrow\forall y\forall xP(x,y)$

1.$[\forall x\forall yP(x,y)]^1$	
2.$\forall yP(a,y)$	$\forall$elim1
3.P(a,b)	$\forall$elim2
4.$\forall xP(x,b)$	$\forall$int3
5.$\forall y\forall xP(x,y)$	$\forall$int4
6.$\forall x\forall yP(x,y)\rightarrow\forall y\forall xP(x,y)^1$	$\rightarrow$I(1,5)

Exemplo 2: $\forall x(Q(y)\rightarrow P(x))\rightarrow(Q(y)\rightarrow\forall xP(x))$

1. $[\forall x(Q(y)\rightarrow P(x))]^1$	
2. $Q(y)\rightarrow P(a)$	$\forall$elim1
3.$[Q(y)]^2$	
4. P(a)	$\rightarrow$E(2,3)
5.$\forall xP(x)$	$\forall$int4
6. $(Q(y)\rightarrow\forall xP(x)^2$	$\rightarrow$I(3,5)
7. $\forall x(Q(y)\rightarrow P(x))\rightarrow(Q(y)\rightarrow\forall xP(x))^1$	$\rightarrow$I(1,6)

Exemplo 3: 1. $\forall x \forall y P(x,y)$

2. $\forall x \forall y (P(x,y) \rightarrow Q(x) \wedge R(y))$ (deduz) $\exists x T(x)$?

3. $\exists x R(x) \wedge \forall x Q(x) \rightarrow \exists x (S(x) \wedge T(x))$

4. $\forall y P(a,y)$	$\forall$elim1
5. $P(a,b)$	$\forall$elim4
6. $\forall y (P(a,y) \rightarrow Q(a) \wedge R(y))$	$\forall$elim2
7. $P(a,b) \rightarrow Q(a) \wedge R(b)$	$\forall$elim6
8. $Q(a) \wedge R(b)$	$\rightarrow$E(5,7)
9. $Q(a)$	$\wedge$E(8)
10. $\forall x Q(x)$	$\forall$int9
11. $R(b)$	$\wedge$E(8)
12. $\exists x R(x)$	$\exists$int11
13. $\exists x R(x) \wedge \forall x Q(x)$	$\wedge$I(10,12)
14. $\exists x (S(x) \wedge T(x))$	$\rightarrow$E(13,3)
15. $[S(a) \wedge T(a)]^1$	
16. $T(a)$	$\wedge$E(15)
17. $\exists x T(x)$	$\exists$int16
18. $\exists x T(x)^1$	$\exists$elim(14,15,17)

III.2.2-Método Axiomático:

◆Os conectivos $\wedge, \vee, \rightarrow, \neg$ são definidos pelos mesmos axiomas esquema da Lógica Proposicional.

Axiomas Lógicos:

- ***Implicação***:
  - (1) $\alpha \rightarrow(\beta\rightarrow\alpha)$
  - (2) $(\alpha \rightarrow(\beta\rightarrow\gamma)) \rightarrow((\alpha \rightarrow \beta) \rightarrow (\alpha \rightarrow\gamma))$

- ***Conjunção:***
  - (3) $(\alpha \wedge \beta) \rightarrow \alpha$
  - (4) $(\alpha \wedge \beta) \rightarrow \beta$
  - (5) $\alpha \rightarrow (\beta \rightarrow (\alpha \wedge \beta))$

- ***Disjunção:***
  - (6) $\alpha \rightarrow \alpha \vee \beta$
  - (7) $\beta \rightarrow \alpha \vee \beta$
  - (8) $((\alpha \rightarrow \gamma) \wedge (\beta \rightarrow \gamma)) \rightarrow ((\alpha \vee \beta) \rightarrow \gamma)$

- ***Negação***:
  - (9) $\alpha \rightarrow \neg\neg\alpha$
  - (10) $\neg\neg\alpha \rightarrow \alpha$
  - (11) $(\alpha \rightarrow \beta) \rightarrow ((\alpha \rightarrow \neg\beta) \rightarrow \neg\alpha)$

◆**Quantificador Universal:**

(12) $\forall x\alpha\rightarrow\alpha(x/t)$, onde x é substituível por t em α; ($\forall$-elim)

(13) $(\alpha\rightarrow\beta)\rightarrow(\alpha\rightarrow\forall x\beta)$, onde x não ocorre livre em α; ($\forall$-introd)

(14)$\exists x\alpha(x)\leftrightarrow\neg\forall x\neg\alpha(x)$ ***por definição***

Exemplo 1: 1.$\forall xP(x)$

2.$\forall x((P(x) \wedge Q(x))\rightarrow R(x))$ $\quad\wp\quad$ $\forall yR(y)$?

3.$\forall x\, Q(x)$

4.$\forall xP(x)\rightarrow P(y)$ axioma 12 x/y
5.$P(y)$ MP(1,4)
6.$\forall xQ(x)\rightarrow Q(y)$ axioma 12 x/y
7.$Q(y)$ MP(3,6)
8.$P(y)\rightarrow(Q(y)\rightarrow(P(y)\wedge Q(y)))$ axioma 5
9.$Q(y)\rightarrow(P(y) \wedge Q(y))$ MP(5,8)
10.$P(y) \wedge Q(y)$ MP(7,9)

11. $\forall x((P(x) \wedge Q(x)) \rightarrow R(x)) \rightarrow ((P(y) \wedge Q(y)) \rightarrow R(y))$ axioma12 x/y
12. $(P(y) \wedge Q(y)) \rightarrow R(y)$ MP(2,11)
13. $R(y)$ MP(10,12)
14. $R(y) \rightarrow (\forall x P(x) \rightarrow R(y))$ axioma1
15. $\forall x P(x) \rightarrow R(y)$ MP(13,14)
16 $(\forall x P(x) \rightarrow R(y)) \rightarrow (\forall x P(x) \rightarrow \forall y R(y))$ axioma13
17. $\forall x P(x) \rightarrow \forall y R(y)$ MP(15,16)
18. $\forall y R(y)$ MP(1,17)

OBS: As noções de prova, teorema e a relação de derivabilidade $\Gamma \wp \alpha$ são análogas às da Lógica Proposicional.

III.3. Semântica:

↳Revisão da Semântica da Lógica Proposicional:

Maria foi ao cinema. Se ela foi ao cinema então ela comprou pipoca e assistiu ao filme. Se ela comprou pipoca então ela tem dinheiro ou ela pegou emprestado com João. Se ela pegou emprestado com João então João tem dinheiro.

C: Maria foi ao cinema.
P: Maria comprou pipoca.
F: Maria assistiu ao filme.
D: Maria tem dinheiro.
E: Maria pegou dinheiro emprestado com João.
J: João tem dinheiro.

C
$C \rightarrow P \wedge F$
$P \rightarrow D \vee E$
$E \rightarrow J$

v(C) = F v(P) = V v(D) = v(E) = F v(J) = V	Não é um modelo para o conjunto de fórmulas, pois não satisfaz todas as fórmulas.
v(C) = V v(P) = V v(F) = V v(D) = F v(E) = V v(J) = V	É um modelo.
v(C) = V v(P) = V v(F) = V v(D) = V v(E) = F v(J) = F	É um modelo.

- A semântica da lógica de primeira ordem tem como objetivo atribuir significados às fórmulas da linguagem.
- Uma fórmula só tem significado quando uma interpretação é dada a seus símbolos não lógicos.

↳ $\forall x(Q(x)\rightarrow P(x))$ é verdadeira ou falsa?

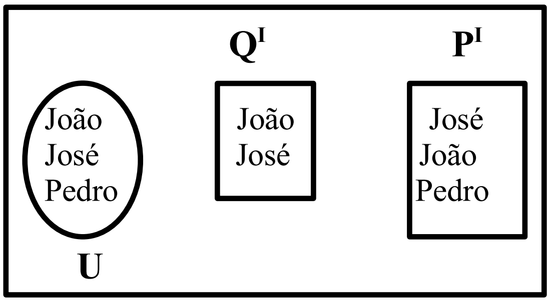
Nós só podemos dizer se esta fórmula é V ou F se interpretarmos seus símbolos não-lógicos.

Primeiro, precisamos saber qual o universo em que as variáveis estão quantificando. Por exemplo: números inteiros, números reais, pessoas...

Depois, precisamos interpretar os predicados, funções e constantes.

Exemplo: $\forall x(Q(x)\rightarrow P(x))$

Interpretação: •**universo:** pessoas
•**predicados:** Q: é funcionário da UFRJ.
P: é funcionário público.



Estrutura

$\forall x(Q(x)\rightarrow P(x))$ é verdadeira na interpretação acima.

Exemplo 2:

U={João, José, Pedro}
Q^I = { <João>,<José>}
P^I = {<José>,<Pedro>}

$\forall x(Q(x)\rightarrow P(x))$ é falsa nesta interpretação.

Exemplo 3: $\forall x(Q(x)\rightarrow P(x))$

U = Z (inteiros)
Q^I = { <0>,<1>,...} (naturais)
P^I = {...<-2>,<-1>,<0>,<1>,<2>,...} (inteiros)

$\forall x(Q(x)\rightarrow P(x))$ é verdadeira nesta interpretação.

Exemplo 4: $\exists x(P(x) \wedge Q(x,c))$

U = R (reais)
$Q^I = x > c$
P^I = x é racional
$c^I = 0$
"Existe algum número real que também é racional e maior do que zero.

Exemplo 5: $\forall x(P(x) \wedge Q(x) \rightarrow R(x,f(c)))$

$U = Z$ (inteiros)
$c^I = 0$
$f^I = x + 1$
$Q^I = \{<2>, <4>, <6>, ...\}$
$P^I = \{<1>, <2>, <3>, ...\}$
$R^I = x > y$

"Todo número inteiro positivo e par é maior do que 1." (verdadeiro)

Exemplo 6: $\forall x(P(x) \land Q(x) \rightarrow R(x,f(c)))$
$U = Z$ (inteiros)
$c^I = 4$
$f^I = x + 1$
$Q^I = \{<2>, <4>, <6>, ...\}$
$P^I = \{<1>, <2>, <3>, ...\}$
$R^I = x > y$

"Todo número inteiro positivo e par é maior do que 4." (falso)

Exemplo 7: $(\forall y C(x,y)) \rightarrow G(x)$
U = {José,João,Pedro,Paulo}
C : x é chefe de y
G : x é gerente

C^I

João	José
João	Paulo
João	Pedro
João	João
Paulo	João
Paulo	Paulo
Paulo	Pedro
Paulo	José
Pedro	José

G^I

João
Pedro

x = João V
x = José V
x = Pedro V
x = Paulo F

◆ Em geral, para sabermos se uma fórmula é verdadeira ou falsa, nós precisamos saber o universo e interpretar cada símbolo não-lógico neste universo.

(1) Interpretar variáveis livres e constantes em elementos do domínio.
(2) Interpretar predicados em relações entre elementos do domínio.
(3) Interpretar funções em funções sobre o domínio.

Def inição: Definimos uma **interpretação** como sendo um par ordenado <D,I> onde D é um conjunto não-vazio de indivíduos chamado **domínio**. E V é uma função chamada de **função de interpretação**, definida como:

1.I associa a cada variável livre x um elemento do domínio $d^I \in D$.

$$I(x) = d^I$$

2.I associa a cada constante c, um elemento do domínio $c^I \in D$.

$$I(c) = c^I$$

3.I associa a cada símbolo funcional n-ário f uma função n-ária $f^I : D^n \rightarrow D$ tal que $I(f(t_1,...,t_n)) = f^I(I(t_1),...,I(t_n))$, onde $t_1,...,t_n$ são termos.

4.I associa a cada símbolo predicativo n-ário P uma relação n-ária sobre D.

$I(P) = P^I$ $\qquad P^I \subseteq D^I$

Definição:

Seja L uma linguagem de primeira ordem e α e β, fórmulas de L, $t_1,...,t_n$ termos, P um símbolo predicativo n-ário e <D,I> uma interpretação. Definimos a função de avaliação de fórmulas de L como:

$V_I : W \rightarrow \{V,F\}$, onde W é o conjunto de fórmulas,
tal que:

(1) $V_I(P(t_1,...,t_n)) =$ V se somente se $<I(t_1),...,I(t_n)> \in P^I$
F caso contrário.

(2) $V_I(\neg\alpha) =$ V se $V_I(\alpha) = F$
F caso contrário.

(3) $V_I(\alpha \wedge \beta) =$ V se $V_I(\alpha) = V$ e $V_I(\beta) = V$
F caso contrário.

(4) $V_I(\alpha \vee \beta) =$ F se $V_I(\alpha) = F$ e $V_I(\beta) = F$
V caso contrário.

(5) $V_I(\alpha \rightarrow \beta) =$ F se $V_I(\alpha) = V$ e $V_I(\beta) = F$
V caso contrário.

(6) $V_I(\forall x\alpha) = V$ se somente se para todo $d \in D$, $V_I(\alpha) = V$
F caso contrário.

(7) $V_I(\exists x\alpha) = V$ se para algum $d \in D$, $V_I(\alpha) = V$
F caso contrário.

Definição:

Seja L uma linguagem de 1ª ordem. I uma interpretação para L, Γ um conjunto de fórmulas de L e α uma fórmula.

1. I **satisfaz** α ($\varnothing_I \alpha$) se somente se $V_I(\alpha) = V$;
2. I **satisfaz** Γ se somente se satisfaz cada membro de Γ;
3. Γ é **satisfatível** se somente se existe uma interpretação I que satisfaça Γ;
4. α é **válida** ($\varnothing\ \alpha$) se somente se para toda interpretação I , $\varnothing_I \alpha$, i.e., $V_I(\alpha) = V$ para todo I; (*válida é equivalente a tautologia*)
5. Γ **implica logicamente** em α ($\Gamma \varnothing\ \alpha$) se somente se para toda interpretação I, se I satisfaz Γ, então I satisfaz α;
6. Γ é **insatisfatível** se somente se Γ não é satisfatível, i.e., não existe uma interpretação I que satisfaz Γ;

(7) Uma interpretação I que satisfaz Γ é dita **modelo** para Γ.

Exemplo 1: $\forall x(P(x) \rightarrow E(x,s(x))$

Esta fórmula é satisfatível?

Interpretação: D = { João, José, Pedro, 0,100, 200}
P : pessoas
E : empregado
s : salário

pessoas

João
José

empregado

João	100
José	200
Pedro	0

salário

José	100
João	200
...	...

OBS: As funções têm que ser totais,i.e., devem retornar algum valor pertencente ao domínio a cada elemento do domínio.

$V_I(\forall x(P(x) \rightarrow E(x,s(x))) = ?$

- Para todo $d \in D$:

d = João

$V_I(P(João)) = V \quad V_I(E(João,200)) = F \Rightarrow V_I(\forall x(P(x) \rightarrow E(x,s(x))) = F$

$*$ Trocando os valores na tabela de salário:

salário

José	200
João	100
...	...

- Para todo $d \in D$:

d = João

$V_I(P(\text{João})) = V \quad V_I(E(\text{João},100)) = V \;\Rightarrow V_I(\forall x(P(x) \to E(x,s(x))) = V$

d = José

$V_I(P(\text{José})) = V \quad V_I(E(\text{José},200)) = V \;\Rightarrow V_I(\forall x(P(x) \to E(x,s(x))) = V$

d = Pedro

$V_I(P(\text{Pedro})) = F \quad V_I(E(\text{Pedro},...)) = ? \;\Rightarrow V_I(\forall x(P(x) \to E(x,s(x))) = V$

$d = 0$

$V_I(P(0)) = F \quad V_I(E(0,...)) = ? \;\Rightarrow V_I(\forall x(P(x) \to E(x,s(x))) = V$

$d = 100$

$V_I(P(100)) = F \quad V_I(E(100,...)) = F \;\Rightarrow V_I(\forall x(P(x) \to E(x,s(x))) = V$

$d = 200$

$V_I(P(200)) = F \quad V_I(E(200,...)) = F \;\Rightarrow V_I(\forall x(P(x) \to E(x,s(x))) = V$

Exemplo 2: $\forall x(P(x) \wedge \exists y Q(x,y))$

não satisfaz	**satisfaz**
$D = \{0,1\}$	$D = \{0,1\}$
$P^I = \{<0>\}$	$P^I = \{<0>,<1>\}$
$Q^I = \{<0,1>\}$	$Q^I = \{<0,1>,<1,0>\}$

Exemplo 3: $\forall x(P(x,y) \to Q(x) \vee R(c))$

não satisfaz	**satisfaz**
$D = \{0,1\}$	$D = \{0,1\}$
$y^I = 0$	$y^I = 0$
$c^I = 0$	$c^I = 0$
$P^I = \{<1,1>\}$	$P^I = \{<0,0>,<1,0>\}$
$Q^I = \{<1>\}$	$Q^I = \varnothing$
$R^I = \{<0>\}$	$R^I = \{<1>\}$

III.4 - Relação entre Sintaxe e Semântica:

TEOREMA DA CORRETUDE:

Se $\Gamma \wp \alpha$ então $\Gamma \varnothing \alpha$..

TEOREMA DA COMPLETUDE:

Se $\Gamma \varnothing \alpha$ então $\Gamma \wp \alpha$..