Introdução à Lógica Simbólica

Joaquim Reis

ISCTE-IUL, DCTI, Setembro de 2013

Índice

ÍNDICE	**1**
TABELAS PARA LÓGICA SIMBÓLICA	**1**
REGRAS DE INFERÊNCIA PARA DEDUÇÃO NATURAL	**2**
1- HISTÓRIA RESUMIDA DA LÓGICA NO OCIDENTE	**7**
2- ALGUMAS DEFINIÇÕES DA LÓGICA	**11**
3- CÁLCULO PROPOSICIONAL - SINTAXE	**13**
4- CÁLCULO PROPOSICIONAL - SEMÂNTICA	**14**
5- CÁLCULO PROPOSICIONAL – DEDUÇÃO NATURAL	**21**
6- CÁLCULO PROPOSICIONAL – O PRINCÍPIO DA RESOLUÇÃO	**31**
7- CÁLCULO DE PREDICADOS – SINTAXE	**35**
8- CÁLCULO DE PREDICADOS - SEMÂNTICA	**38**
9- CÁLCULO DE PREDICADOS – DEDUÇÃO NATURAL	**42**
10- CÁLCULO DE PREDICADOS – O PRINCÍPIO DA RESOLUÇÃO	**51**
BIBLIOGRAFIA	**60**

Tabelas para Lógica Simbólica

<u>Conectivas lógicas</u> por ordem de prioridade:

1	$\neg$ negação	(*não*)				
2	$\wedge$ conjunção	(*e*)	$\vee$ disjunção	(*ou*)		
3	$\Rightarrow$ implicação	(*se ..., então ...*)	$\Leftrightarrow$ equivalência	(*... se e só se ...*)		
4	$\forall$ quantificador universal	(*para todos ...*)	$\exists$ quantificador existencial	(*existe um ...*)		

<u>Nota</u>: Conectivas em 4, só para cálculo de predicados.

<u>Tabelas de verdade</u> das conectivas lógicas (V = VERDADEIRO, F = FALSO):

G	$\neg$G
V	F
F	V

G	H	G $\wedge$ H	G $\vee$ H	G $\Rightarrow$ H	G $\Leftrightarrow$ H
V	V	V	V	V	V
V	F	F	V	F	F
F	V	F	V	V	F
F	F	F	F	V	V

<u>Equivalências lógicas</u>:

	equivalências				
1		$P \Leftrightarrow Q \equiv (P \Rightarrow Q) \wedge (Q \Rightarrow P)$			eliminação da equivalência
2		$P \Rightarrow Q \equiv \neg Q \Rightarrow \neg P$			contraposição
3		$P \Rightarrow Q \equiv \neg P \vee Q$			eliminação da implicação
4	a)	$P \vee Q \equiv Q \vee P$	b)	$P \wedge Q \equiv Q \wedge P$	comutatividade
5	a)	$(P \vee Q) \vee R \equiv P \vee (Q \vee R)$	b)	$(P \wedge Q) \wedge R \equiv P \wedge (Q \wedge R)$	associatividade
6	a)	$P \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R)$	b)	$P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$	distributividade
7	a)	$P \vee F \equiv P$	b)	$P \wedge V \equiv P$	elemento neutro
8	a)	$P \vee V \equiv V$	b)	$P \wedge F \equiv F$	elemento absorvente
9	a)	$P \vee \neg P \equiv V$	b)	$P \wedge \neg P \equiv F$	tautologia, contradição
10		$\neg\neg P \equiv P$			dupla negação
11	a)	$\neg(P \vee Q) \equiv \neg P \wedge \neg Q$	b)	$\neg(P \wedge Q) \equiv \neg P \vee \neg Q$	leis de de Morgan
12	a)	$\neg(\forall x\ P(x)) \equiv \exists x\ \neg P(x)$	b)	$\neg(\exists x\ P(x)) \equiv \forall x\ \neg P(x)$	negação da quantificação
	c)	$\forall x\ P(x) \equiv \neg(\exists x\ \neg P(x))$	d)	$\exists x\ P(x) \equiv \neg(\forall x\ \neg P(x))$	
13	a)	$(Kx\ P(x)) \vee Q \equiv Kx\ P(x) \vee Q$	b)	$(Kx\ P(x)) \wedge Q \equiv Kx\ P(x) \wedge Q$	quantificação da conjunção e da disjunção
14	a)	$(K_1x\ P(x)) \vee (K_2x\ Q(x)) \equiv K_1x\ K_2y\ P(x) \vee Q(y)$	b)	$(K_1x\ P(x)) \wedge (K_2x\ Q(x)) \equiv K_1x\ K_2y\ P(x) \wedge Q(y)$	

<u>Nota</u>: As equivalências 12 a 14 são válidas apenas para o cálculo de predicados. Naquelas, $P(x)$ e $Q(x)$ significam fórmulas contendo a variável x, Q uma fórmula não contendo a variável x, e K, K_1 e K_2 significam quantificadores ($\forall$ ou $\exists$).

Regras de Inferência para Dedução Natural

regras de introdução / regras de eliminação

I.C. Introdução da Conjunção

$$\frac{\begin{array}{l}P\\ Q\end{array}}{P \land Q}$$

E.C. Eliminação da Conjunção

$$\frac{P \land Q}{\begin{array}{l}P\\ Q\end{array}}$$

I.D. Introdução da Disjunção

$$\frac{P}{P \lor Q}$$

E.D. Eliminação da Disjunção

$$\frac{\begin{array}{l}P \lor Q\\ \neg P\end{array}}{Q}$$

I.D.N. Introdução da Dupla Negação

$$\frac{P}{\neg\neg P}$$

E.D.N. Eliminação da Dupla Negação

$$\frac{\neg\neg P}{P}$$

M.P. Modus Ponens

$$\frac{\begin{array}{l}P \Rightarrow Q\\ P\end{array}}{Q}$$

M.T. Modus Tolens

$$\frac{\begin{array}{l}P \Rightarrow Q\\ \neg Q\end{array}}{\neg P}$$

I.U. Instanciação Universal

$$\frac{\forall x\ P(x)}{P(\tau)}$$

G.E. Generalização Existencial

$$\frac{P(A)}{\exists x\ P(x)}$$

I.E. Instanciação Existencial

$$\frac{\exists x\ P(x)}{P(\tau(v_1, \ldots, v_n))}$$

<u>Notas</u>: Na regra para I.U., τ é uma constante, ou uma função (termo funcional), que denota um objecto conhecido do domínio de discurso. Na regra para I.E., $\tau(v_1, \ldots, v_n)$ é uma função (função de Skolem) das n variáveis livres ($v_1, \ldots, v_n$) que existam no âmbito do quantificador existencial (se não existir variável livre alguma nesse âmbito essa será uma função de 0 variáveis, ou constante) completamente nova e arbitrária. Estas regras são uma versão simplificada não completamente correta porque P pode ter mais variáveis, incluindo outras instâncias da mesma variável x.

1- História Resumida da Lógica no Ocidente

Séc. V A.C. – **Escola de Mégara** (ou escola erística - arte da controvérsia -, Corinto, fundada por Euclides, o Socrático) e **escola dos Estóicos** (Zenão de Cítio)

Estudaram sistematicamente a implicação e outras operações lógicas ainda hoje usadas no cálculo proposicional. Os filósofos de Mégara já usavam tabelas de verdade para explicar o significado das conectivas lógicas do cálculo proposicional.

Foram rigorosos e explícitos na análise das regras de inferência (mais do que Aristóteles). A partir de 5 regras de inferência tomadas *a priori* como válidas derivaram as restantes. Uma das regras era o Modus Ponens (que usa a implicação).

Séc. IV A.C. – **Aristóteles**

Tentou codificar o pensamento correcto, *i.e.*, os processos de raciocínio irrefutáveis, ou silogismos. Os silogismos permitem tirar a conclusão correcta dadas as premissas correctas. Exemplo:

Sócrates é um homem
Todos os homens são mortais
Portanto, Sócrates é mortal

Os gregos estudaram extensivamente a inferência lógica (raciocínio lógico). Aristóteles classificou os silogismos. Tentou provar a validade das várias formas de silogismo reduzindo-os à forma que seria a mais importante:

Todos os ? são ??
Todos os ?? são ???
Portanto, todos os ? são ???

A lógica de Aristóteles era uma lógica não simbólica (não abstracta), actualmente correspondente a uma lógica de ordem 0 (ou cálculo proposicional, em que as frases lógicas exprimem factos acerca de objectos concretos do domínio de discurso) ou, em alguns casos, uma forma limitada de lógica de 1ª ordem (ou cálculo de predicados, em que as frases lógicas exprimem propriedades e relações gerais entre objectos do domínio de discurso). A lógica de Aristóteles era limitada (pelos padrões modernos). Havia limites à complexidade interna das frases lógicas que podiam ser analisadas. Não tinha em conta os princípios lógicos de imbricação de frases lógicas (salvo raras excepções de frases lógicas isoladas) e tratava apenas com predicados de um único argumento. Augustus de Morgan indica como exemplo de frase não tratável pela lógica de Aristóteles a seguinte:

Todos os cavalos são animais
A cabeça de um cavalo é, portanto, a cabeça de um animal

Para analisar logicamente esta frase lógica é necessário empregar o predicado de 2 argumentos "y é a cabeça de x" (Cabeça(x,y)). No cálculo de predicados convencional actual seria, de:

$\forall$x Cavalo(x) $\Rightarrow$ Animal(x) (para todo o x, se x é um cavalo, então x é um animal)
Cavalo(y) $\wedge$ Cabeça(y,z) (y é um cavalo e z é a cabeça de y)

inferir:

Animal(y) $\wedge$ Cabeça(y,z) (y é um animal e z é a cabeça de y).

Séc. XVII – Gottfried Wilhelm **Leibnitz** (1646-1717)

(teoria das mónadas, análise matemática – em simultâneo com Isaac Newton -, máquina calculadora, defesa da fusão entre igrejas católica e protestante)

Ideias relativas à lógica:

- Criar uma linguagem artificial formal, baseada numa notação matemática, para clarificar as relações lógicas.
- Reduzir a inferência lógica a um processo puramente formal e mecânico e poder dispor de um procedimento geral de decisão para provar teoremas.

1847 – George **Boole**

Introduziu o 1º sistema lógico aproximadamente correcto e razoavelmente abrangente baseado numa linguagem artificial formal (superior ao sistema de Aristóteles).

1864 – Augustus **de Morgan**

Tratou pela 1ª vez (desde os tempos Aristóteles), de forma sistemática, as relações lógicas, *i.e.*, os predicados de mais de 1 argumento.

1870 – Charles Sanders **Peirce**

Estudou em profundidade a lógica das relações. Desenvolveu também a lógica de 1ª ordem (independentemente de Frege e mais tarde, em 1883).

1877 – Ernst **Schröder**

Forma normal conjuntiva e forma normal disjuntiva.

Forma normal conjuntiva - Nesta forma, uma fórmula lógica é composta apenas por conjunções de disjunções de literais:

$(L_{1,1} \vee ... \vee L_{1,k_1}) \wedge ... \wedge (L_{i,1} \vee ... \vee L_{i,k_i}) \wedge ... \wedge (L_{n,1} \vee ... \vee L_{n,k_n})$

Forma normal disjuntiva - Nesta forma, uma fórmula lógica é composta apenas por disjunções de conjunções de literais:

$(L_{1,1} \wedge ... \wedge L_{1,k_1}) \vee ... \vee (L_{n,1} \wedge ... \wedge L_{n,k_n})$

Em ambos os casos os $L_{i,j}$ significam literais.

1879 – Gottlob **Frege**

Lógica simbólica (com manipulação de símbolos abstractos). Notação difícil. O 1º a expor de forma abrangente o cálculo proposicional e o cálculo de predicados. Semântica do cálculo de predicados. Introdução de quantificadores e possibilidade de imbricar âmbitos de quantificadores nas frases lógicas.

1895 – Giuseppe **Peano** (1858-1932)

Notação actual da lógica de 1ª ordem.

1910 – A. N. **Whitehead** e B. **Russell**

(*Principia Mathematica*) Reduziram as leis básicas da aritmética a proposições lógicas elementares. Passagem à forma prenex, *i.e.* frases lógicas com quantificadores no início:

Quantificadores FraseLógica

1915 – Leopold **Löwenheim**

Teoria dos modelos.

1920 – Thoralf **Skolem**

Ampliou a teoria dos modelos. Tratamento da quantificação existencial com constantes e funções de Skolem.

1921 – Emil **Post**

1922 – Ludwig **Wittgenstein**

Uso de tabelas de verdade (já antes usadas pelos filósofos de Mégara para explicar o significado das conectivas da lógica proposicional) para testar a validade ou a não-satisfiabilidade na linguagem do cálculo proposicional.

1930 – Jacques **Herbrand**

Com o teorema de Herbrand, sugeriu um método importante para a automatização do raciocínio, difícil de aplicar, no entanto, por ser demorado, já que não havia computadores.

1930 – Kurt **Gödel**

A lógica de 1ª ordem tem procedimentos de prova completos.

1934 – Gerhard **Gentzen**. Stanislaw **Jáskowski**

Sistemas de dedução natural. Nestes sistemas não é requerido um preprocessamento extenso das fórmulas lógicas e as regras de inferência são intuitivas.

1935 – Thoralf **Skolem**

Definição de verdade e de satisfação lógica segundo o modelo da lógica de 1ª ordem empregando teoria dos conjuntos.

1936 – Alan **Turing**. Alonzo **Church**

A validade na lógica de 1ª ordem não é decidível.

1951 – Alfred **Horn**

Forma de Horn:	$\neg A_1 \vee \neg A_2 \vee ... \vee \neg A_n \vee C$
equivalente a:	$\neg(A_1 \wedge A_2 \wedge ... \wedge A_n) \vee C$
por sua vez equivalente a:	$A_1 \wedge A_2 \wedge ... \wedge A_n \Rightarrow C$

1960 – P. C. **Gilmore**. M. **Davis** e H. **Putnam**

Implementação em computador do método de Herbrand.

1965 – J. A. **Robinson**

Princípio da resolução, uma regra de inferência eficiente e facilmente realizável em computador.

1964, 1967 – S. **Maslov**

"Método inverso", equivalente ao princípio da resolução e igualmente eficiente do ponto de vista computacional.

1958 – John **McCarthy**

Introduz a lógica de 1ª ordem como ferramenta para construir sistemas de Inteligência Artificial.

1962 – Jaakko **Hintikka**

Inventou a lógica modal, um método para raciocinar sobre o conhecimento em filosofia. A lógica modal amplia a lógica de 1ª ordem com operadores de crença e conhecimento que tomam como argumentos frases lógicas.

1963 – Saul **Kripke**

Semântica da lógica modal em termos de mundos possíveis, i.e, os mundos que são consistentes com tudo aquilo que o agente conhece.

1963 – John **McCarthy**

Cálculo situacional, espécie de lógica temporal para raciocínio sobre "estados do mundo", ou "situações", associados a "pontos" no tempo.

A lógica temporal, em que se estuda o tempo e os acontecimentos, é uma forma de lógica modal em que os operadores modais se referem aos instantes de tempo em que os factos descritos são verdadeiros (por exemplo, determinado facto será verdadeiro em todos os instantes futuros, ou será verdadeiro num certo instante futuro). Este estudo remonta à Antiguidade com Aristóteles e as escolas de Mégara e a dos Estóicos da Grécia antiga.

1968 – John **McCarthy**

(*Programs with Common Sense*) Agentes inteligentes que usam raciocínio lógico para ir da percepção à acção. Assume que é aceitável atribuir qualidades, ou atributos, "mentais" (por exemplo, crenças, objectivos, intenções) a artefactos "inteligentes" se isso é adequado para uma melhor compreensão deles do ponto de vista do utilizador.

1976 – Vaughan **Pratt**

Lógica dinâmica, uma forma de lógica temporal em que há operadores modais para designar os efeitos de acções e de programas de computador.

1979 – Robert **Kowalski**

Forma normal implicativa (ou forma de Kowalski). Semelhante à forma de Horn mas pode ter disjunções no consequente.

Exemplos: $A_1 \wedge A_2 \wedge ... \wedge A_n \Rightarrow C_1 \vee C_2 \vee ... \vee C_m$
$\text{VERDADEIRO} \Rightarrow C_1 \vee C_2 \vee ... \vee C_m$

Exemplos de equivalência entre as duas formas:

Forma de Horn	Forma normal implicativa
$\neg P \vee Q$	$P \Rightarrow Q$
$R \vee S$	$\text{VERDADEIRO} \Rightarrow R \vee S$

1984 – James **Allen**

Raciocínio com relações entre intervalos (e não pontos, como no cálculo situacional) de tempo.

1984 – Peter **Ladkin**

Raciocínio com relações entre intervalos de tempo côncavos também e não apenas convexos (convencionais).

1987 – J. S. **Rosenschein** e M. R. **Genesereth**

Abordagem do problema da cooperatividade das acções de agentes inteligentes que internamente empregam lógica para representar o mundo exterior.

2- Algumas Definições da Lógica

Lógica (definições variadas):

- "Arte" de raciocinar correctamente.
- Forma explícita de representar o raciocínio.
- Sistema conceptual simbólico para prova de teoremas.
- Linguagem formal para exprimir conhecimento e métodos para raciocinar usando essa linguagem.

Cálculo proposicional (ou lógica proposicional, ou lógica de ordem 0)

As fórmulas (ou frases) lógicas representam factos e combinações de factos através de operações lógicas.

Cálculo de predicados (ou lógica de predicados, ou lógica de 1ª ordem)

As fórmulas (ou frases) lógicas representam factos, objectos e predicados e combinações deles através de operações lógicas e de quantificadores (predicados exprimem propriedades ou relações entre objectos).

Comparação de vários formalismos:

Linguagem formal (lógica)	Suposições Ontológicas (natureza da realidade)	Suposições Epistemológicas (tipo de crença)
cálculo proposicional	factos	verdadeiro, falso, desconhecido
cálculo de predicados	factos, objectos, relações	verdadeiro, falso, desconhecido
lógica temporal	factos, objectos, relações, tempo	verdadeiro, falso, desconhecido
teoria das probabilidades	factos	grau de crença (0..1)
lógica vaga	grau de verdade	grau de crença (0..1)

Um sistema de lógica é composto por:

1- Sub-sistema sintático (sintaxe) – Descreve como se formam fórmulas lógicas bem formadas.
2- Sub-sistema semântico (semântica) – Descreve como as fórmulas lógicas se relacionam com os factos do mundo real e que fórmulas são verdadeiras ou falsas.
3- Sub-sistema de prova (regras de inferência) – Conjunto de regras para deduzir consequência lógica de fórmulas lógicas a partir de outras fórmulas lógicas.

Lógica sólida (ou correcta, ou sã, ou que preserva a verdade) – Sistema de lógica que permite gerar apenas as fórmulas que são consequências lógicas dos axiomas.

Lógica completa – Sistema de lógica que permite gerar todas as fórmulas que são consequências lógicas dos axiomas.

Consequência lógica (1) – Uma fórmula G é consequência lógica das fórmulas $P_1, P_2,.., P_n$ se e só se, sempre que P_1 e P_2 e ... e P_n são verdadeiras, G é também verdadeira. $P_1, P_2,.., P_n$ são os axiomas (ou postulados, ou premissas) de G, e G é o teorema.

Axioma (ou postulado, ou premissa) – Fórmula lógica que exprime um facto tomado como facto fundamental ou ponto de partida e que é uma verdade universal.

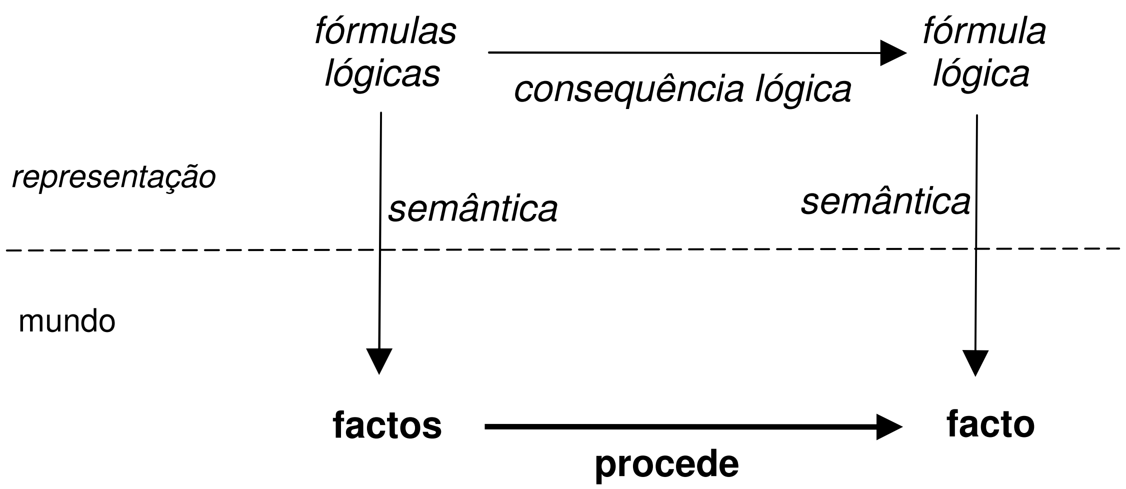
Teorema - Fórmula lógica que exprime um facto que é uma verdade universal considerada relevante e que se pode provar ser uma consequência lógica de axiomas.

Prova – Demonstração de que uma fórmula lógica (o teorema) é consequência lógica de um conjunto de outras fórmulas lógicas (as premissas, ou axiomas).

Inferência (ou raciocínio, ou dedução) – Processo de geração de novas fórmulas lógicas a partir de um conjunto de outras fórmulas lógicas, tomadas como premissas, usando regras de inferência. Se a lógica usada for sólida, as novas fórmulas lógicas serão garantidamente verdadeiras se as premissas forem verdadeiras.

Regra de inferência (ou regra de dedução, ou silogismo) (1) – Forma particular de geração de uma nova fórmula lógica a partir de outras fórmulas lógicas tomadas como premissas.

A ideia base da lógica como forma de representação dos, e raciocínio com os, factos do mundo real (ligação entre fórmulas e factos) é ilustrada pela figura seguinte:



3- Cálculo Proposicional - Sintaxe

Proposição – Fórmula lógica cujo valor lógico pode ser VERDADEIRO (V) ou FALSO (F). Pode ser:

Proposição simples (ou fórmula atómica, ou átomo) – Fórmula lógica unitária representando uma frase declarativa que pode ser ou verdadeira ou falsa.

Proposição composta – Combinação de proposições simples através de conectivas lógicas.

Conectiva lógica (de cálculo proposicional) – Símbolo que denota uma operação lógica. Pode ser (por ordem de prioridade):

1º	¬	negação	(*não*)			
2º	∧	conjunção	(*e*)	∨	disjunção	(*ou*)
3º	⇒	implicação	(*se ..., então ...*)	⇔	equivalência	(*... se e só se ...*)

Literal – Uma proposição simples (literal positivo) ou uma proposição simples negada (literal negativo). Por exemplo, se P for uma proposição simples, P e ¬P são ambos literais (o primeiro é um literal positivo e o segundo é um literal negativo).

Sintaxe do cálculo proposicional

Fórmulas lógicas bem formadas:

1- Uma proposição simples (ou átomo, ou fórmula atómica) é uma fórmula.
2- Se G é uma fórmula, então ¬G é uma fórmula.
3- Se G e H são fórmulas, então G ∧ H, G ∨ H, G ⇒ H e G ⇔ H são fórmulas.
4- Todas as fórmulas são geradas aplicando estas regras.

Nota: Numa fórmula lógica podem usar-se os parêntesis, os símbolos (e), devidamente emparelhados, para alterar a ordem de prioridade das conectivas.

Mais formalmente, a gramática do cálculo proposicional é a seguinte:

```
Fórmula               = ProposiçãoSimples | ProposiçãoComposta
ProposiçãoSimples     = V | F | P | Q | Xpto  | Xp3Zv ...
ProposiçãoComposta    = ( Fórmula ) | Fórmula Conectiva Fórmula
                        | ¬ Fórmula
Conectiva             = ∧ | ∨ | ⇔ | ⇒
```

Exemplos de fórmulas correctas (bem formadas) e incorrectas:

Fórmulas correctas	Fórmulas incorrectas
P ∨ Q	P ⇒
P ∨ ¬Q	∧)
P ⇒ Q ∨ R	∧ P
P ⇒ (Q ∨ R)	P Q
P ∧ U ⇔ Q ∧ ¬R ∨ S	(P ∨ Q(
P ⇔ (Q ∧ ((¬R) ∨ S)	( )

4- Cálculo Proposicional - Semântica

Como se determina o valor lógico de uma proposição?

1- É associado um facto a cada símbolo de proposição simples.
2- Interpretação: atribuem-se valores lógicos a cada símbolo de proposição simples (o valor lógico de uma proposição simples é VERDADEIRO se o facto associado a ela for verdadeiro e FALSO no caso contrário).
3- O valor lógico de uma proposição composta é dado pela tabela de verdade das conectivas lógicas envolvidas na proposição.

<u>Tabelas de verdade das conectivas lógicas</u> (V = VERDADEIRO, F = FALSO):

G	¬G
V	F
F	V

G	H	G ∧ H	G ∨ H	G ⇒ H	G ⇔ H
V	V	V	V	V	V
V	F	F	V	F	F
F	V	F	V	V	F
F	F	F	F	V	V

<u>Exemplos</u>:

fórmula	factos	interpretação	valor lógico
P	P: Chove em Lisboa	V	V
¬P	P: Chove em Lisboa	V	F
P ∧ Q	P: Chove em Lisboa Q: Faz sol em Coimbra	V V	V
P ∧ Q	P: Chove em Lisboa Q: Faz sol em Coimbra	F V	F
P ∨ Q	P: Chove em Lisboa Q: Faz sol em Coimbra	F V	V
P ∨ Q	P: Chove em Lisboa Q: Faz sol em Coimbra	F F	F
P ⇒ Q	P: Choveu Q: O piso está molhado	V V	V
P ⇒ Q	P: Choveu Q: O piso está molhado	V F	F
P ⇒ Q	P: Choveu Q: O piso está molhado	F V	V
P ⇒ Q	P: Choveu Q: O piso está molhado	F F	V
P ⇔ Q	P: João é irmão de Pedro Q: Pedro é irmão de João	V V	V
P ⇔ Q	P: João é irmão de Pedro Q: Pedro é irmão de João	V F	F
P ⇔ Q	P: João é irmão de Pedro Q: Pedro é irmão de João	F F	V

<u>Interpretação</u> – Atribuição de valores lógicos a cada símbolo de proposição simples de uma fórmula lógica.

Uma fórmula com N símbolos de proposição diferentes tem 2^N interpretações distintas. Para cada interpretação a fórmula lógica tem um valor lógico.

<u>Fórmula verdadeira sob uma interpretação</u> – Fórmula cujo valor lógico é V para essa interpretação.

<u>Fórmula falsa sob uma interpretação</u> – Fórmula cujo valor lógico é F para essa interpretação.

Para deteminar o valor lógico de uma fórmula para cada interpretação possível faz-se uma tabela de verdade.

<u>Exemplos</u>:

1- Para a fórmula $(P \wedge Q) \Rightarrow (R \Leftrightarrow (\neg S))$: ($2^4 = 16$ interpretações)

P	Q	R	S	¬S	P∧Q	R⇔(¬S)	(P∧Q)⇒(R⇔(¬S))
V	V	V	V	F	V	F	F
V	V	V	F	V	V	V	V
V	V	F	V	F	V	V	V
V	V	F	F	V	V	F	F
V	F	V	V	F	F	F	V
V	F	V	F	V	F	V	V
V	F	F	V	F	F	V	V
V	F	F	F	V	F	F	V
F	V	V	V	F	F	F	V
F	V	V	F	V	F	V	V
F	V	F	V	F	F	V	V
F	V	F	F	V	F	F	V
F	F	V	V	F	F	F	V
F	F	V	F	V	F	V	V
F	F	F	V	F	F	V	V
F	F	F	F	V	F	F	V

(consistente e inválida)

2- Para a fórmula $((P \Rightarrow Q) \wedge P) \Rightarrow Q$: ($2^2 = 4$ interpretações)

P	Q	P⇒Q	(P⇒Q)∧P	((P⇒Q)∧P)⇒Q
V	V	V	V	V
V	F	F	F	V
F	V	V	F	V
F	F	V	F	V

(válida e, portanto, consistente)

3- Para a fórmula $(P \Rightarrow Q) \wedge (P \wedge \neg Q)$: ($2^2 = 4$ interpretações)

P	Q	¬Q	(P⇒Q)	(P∧¬Q)	(P⇒Q)∧(P∧¬Q)
V	V	F	V	F	F
V	F	V	F	V	F
F	V	F	V	F	F
F	F	V	V	F	F

(inconsistente e, portanto, inválida)

Fórmula válida (ou tautologia) – Fórmula que é verdadeira para todas as suas interpretações.

Fórmula inválida (ou não válida) – Fórmula que não é verdadeira para todas as suas interpretações (*i.e.*, é falsa pelo menos para uma).

Fórmula inconsistente (ou insatisfiável, ou não satisfiável, ou contradição) – Fórmula que é falsa para todas as suas interpretações.

Fórmula consistente (ou satisfiável) – Fórmula que não é falsa para todas as suas interpretações (*i.e.*, é verdadeira pelo menos para uma).

Assim, deve ser evidente que:

1- Uma fórmula é válida se e só se a sua negação é inconsistente.
2- Uma fórmula é inconsistente se e só se a sua negação é válida.
3- Uma fórmula é inválida se e só se existe pelo menos uma intrepretação sob a qual é falsa.
4- Uma fórmula é consistente se e só se existe pelo menos uma intrepretação sob a qual é verdadeira.
5- Se uma fórmula é válida, então é consistente, mas não vice-versa.
6- Se uma fórmula é inconsistente, então é inválida, mas não vice-versa.

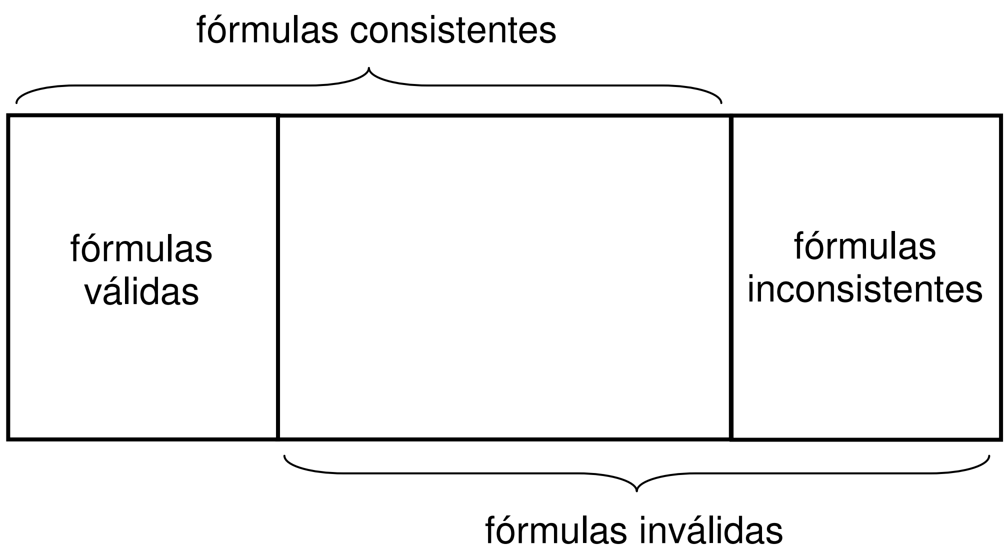
Exemplos:

$P \wedge \neg P$ é inconsistente e, portanto, também inválida.

$P \vee \neg P$ é válida e, portanto, também consistente.

$P \Rightarrow \neg P$ é inválida mas é, no entanto, consistente.

Explicação gráfica (rectângulos representam conjuntos de fórmulas):



Diz-se que uma interpretação I satisfaz a fórmula P, ou que P é satisfeita por I, se P é verdadeira sob I.

Diz-se que uma interpretação I não satisfaz (ou falsifica) a fórmula P, ou que P não é satisfeita (ou é falsificada) por I, se P é falsa sob I.

Diz-se que uma interpretação I é um modelo da fórmula P, se I satisfaz P.

Exercícios:

Das seguintes fórmulas, dizer quais são válidas, inconsistentes ou nem uma coisa nem outra.

a) Fumo $\Rightarrow$ Fumo

b) Fumo $\Rightarrow$ Fogo

c) (Fumo $\Rightarrow$ Fogo) $\Rightarrow$ ($\neg$Fumo $\Rightarrow$ $\neg$Fogo)

d) Fumo $\vee$ Fogo $\vee$ $\neg$Fogo

e) ((Fumo $\wedge$ Calor) $\Rightarrow$ Fogo) $\Leftrightarrow$ ((Fumo $\Rightarrow$ Fogo) $\vee$ (Calor $\Rightarrow$ Fogo))

f) (Fumo $\Rightarrow$ Fogo) $\Rightarrow$ ((Fumo $\wedge$ Calor) $\Rightarrow$ Fogo)

g) Grande $\vee$ Estúpido $\vee$ (Grande $\Rightarrow$ Estúpido)

h) (Grande $\wedge$ Estúpido) $\vee$ $\neg$Estúpido

Equivalência lógica – Duas fórmulas P e Q são equivalentes (ou P é equivalente a Q, o que se denota por P $\equiv$ Q) se e só se, os valores lógicos de P e Q são iguais sob todas as interpretações de P e Q.

Exemplo:

Verificar que $P \Rightarrow Q \equiv \neg P \vee Q$

P	Q	$P \Rightarrow Q$	$\neg P \vee Q$
V	V	V	V
V	F	F	F
F	V	V	V
F	F	V	V

Para operar sobre fórmulas lógicas de cálculo proposicional e transformá-las em fórmulas equivalentes existe um conjunto de equivalências básicas, exposto na seguinte tabela de equivalências (P, Q e R são fórmulas e V e F os valores VERDADEIRO e FALSO, respectivamente):

equivalências

1		$P \Leftrightarrow Q \equiv (P \Rightarrow Q) \wedge (Q \Rightarrow P)$			eliminação.da equivalência
2		$P \Rightarrow Q \equiv \neg Q \Rightarrow \neg P$			contraposição
3		$P \Rightarrow Q \equiv \neg P \vee Q$			eliminação da implicação
4	a)	$P \vee Q \equiv Q \vee P$	b)	$P \wedge Q \equiv Q \wedge P$	comutatividade
5	a)	$(P \vee Q) \vee R \equiv P \vee (Q \vee R)$	b)	$(P \wedge Q) \wedge R \equiv P \wedge (Q \wedge R)$	associatividade
6	a)	$P \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R)$	b)	$P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$	distributividade
7	a)	$P \vee F \equiv P$	b)	$P \wedge V \equiv P$	elemento neutro
8	a)	$P \vee V \equiv V$	b)	$P \wedge F \equiv F$	elemento absorvente
9	a)	$P \vee \neg P \equiv V$	b)	$P \wedge \neg P \equiv F$	tautologia, contradição
10		$\neg\neg P \equiv P$			dupla negação
11	a)	$\neg(P \vee Q) \equiv \neg P \wedge \neg Q$	b)	$\neg(P \wedge Q) \equiv \neg P \vee \neg Q$	leis de de Morgan

Consequência lógica (2) – Uma fórmula G é consequência lógica das fórmulas P_1, P_2,.., P_n se e só se, para qualquer interpretação I sob a qual $P_1 \wedge P_2 \wedge ... \wedge P_n$ é verdadeira, G é também verdadeira. P_1, P_2,.., P_n são os axiomas (ou postulados, ou premissas) de G, e G é o teorema.

Teorema 1 – Uma fórmula G é consequência lógica das fórmulas P_1, P_2,.., P_n se e só se, a fórmula $P_1 \wedge P_2 \wedge ... \wedge P_n \Rightarrow G$ é válida.

Teorema 2 - Uma fórmula G é consequência lógica das fórmulas P_1, P_2,.., P_n se e só se, a fórmula $P_1 \wedge P_2 \wedge ... \wedge P_n \wedge \neg G$ é inconsistente.

Maneiras de verificar se a fórmula G é consequência lógica das fórmulas P_1, P_2,.., P_n,:

Pela definição - Constrói-se uma tabela de verdade para a fórmula $P_1 \wedge P_2 \wedge ... \wedge P_n$ e para a fórmula G para verificar se para cada interpretação em que $P_1 \wedge P_2 \wedge ... \wedge P_n$ é verdadeira G é também verdadeira (e apenas isso pois, note-se que, o contrário não é necessário verificar-se).

Pelo teorema 1 – Constrói-se a tabela de verdade para a fórmula $P_1 \wedge P_2 \wedge ... \wedge P_n \Rightarrow G$ para verificar se esta fórmula é válida.

Pelo teorema 2 – Constrói-se a tabela de verdade para a fórmula $P_1 \wedge P_2 \wedge ... \wedge P_n \wedge \neg G$ para verificar se esta fórmula é inconsistente.

Exemplo:

Considerem-se as seguintes fórmulas de cálculo proposicional:

1	P ⇒ Q
2	¬Q
3	¬P

Demonstrar, por cada um dos três métodos antes descritos, que a fórmula 3 é uma consequência lógica das fórmulas 1 e 2.

Método 1 (pela definição) – Constrói-se uma tabela de verdade para (P⇒Q)∧¬Q e ¬P, para mostrar que ¬P é verdadeira para todos os modelos de (P⇒Q)∧¬Q:

P	Q	(P⇒Q)	¬Q	(P⇒Q)∧¬Q	¬P
V	V	V	F	F	F
V	F	F	V	F	F
F	V	V	F	F	V
F	F	V	V	V	V

Em cada linha em que (P⇒Q)∧¬Q é V, ¬P é também V.

Método 2 (pelo teorema 1) – Constrói-se uma tabela de verdade para ((P⇒Q)∧¬Q)⇒¬P, para mostrar que esta fórmula é válida:

P	Q	(P⇒Q)	¬Q	(P⇒Q)∧¬Q	¬P	((P⇒Q)∧¬Q)⇒¬P
V	V	V	F	F	F	V
V	F	F	V	F	F	V
F	V	V	F	F	V	V
F	F	V	V	V	V	V

A fórmula ((P⇒Q)∧¬Q)⇒¬P é válida.

<u>Método 3</u> (pelo teorema 2) – Constrói-se uma tabela de verdade para $(P \Rightarrow Q) \wedge \neg Q \wedge P$, para mostrar que esta fórmula é inconsistente:

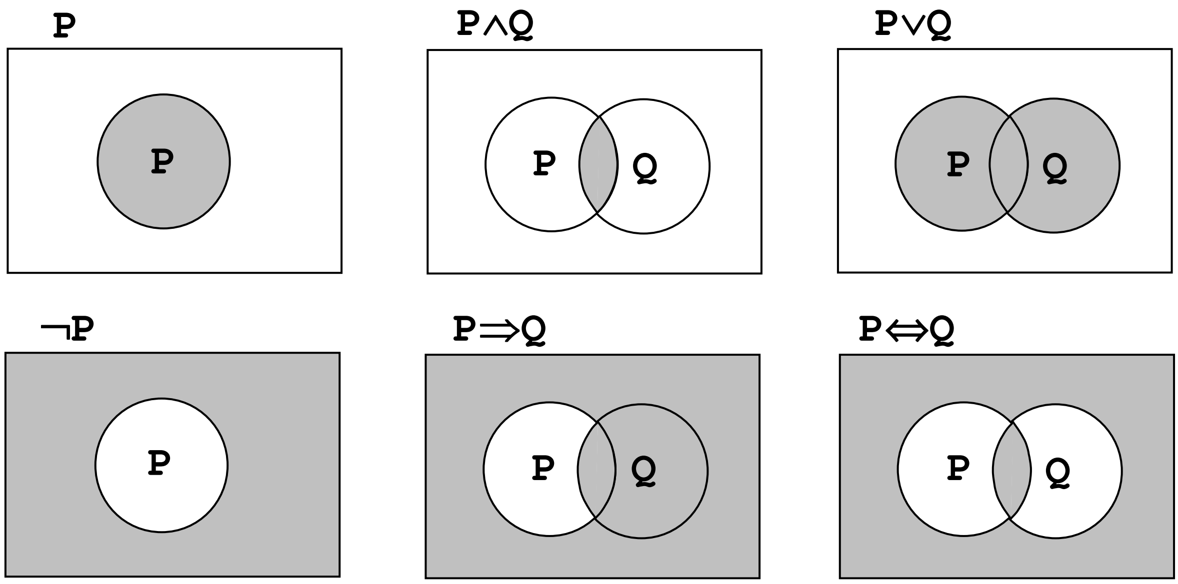
P	Q	$(P \Rightarrow Q)$	$\neg Q$	$(P \Rightarrow Q) \wedge \neg Q$	$(P \Rightarrow Q) \wedge \neg Q \wedge P$
V	V	V	F	F	F
V	F	F	V	F	F
F	V	V	F	F	F
F	F	V	V	V	F

A fórmula $(P \Rightarrow Q) \wedge \neg Q \wedge P$ é inconsistente.

<u>Semântica e teoria dos modelos</u> – Como foi dito, um modelo de uma fórmula lógica é uma interpretação que satisfaz essa fórmula. Por outras palavras, um modelo representa um mundo onde essa fórmula é verdadeira.

Os modelos permitem definir o significado de uma fórmula lógica de uma forma alternativa à de defini-lo através de uma atribuição de valores lógicos a cada um dos símbolos de proposição simples da fórmula lógica, como se fez anteriormente ao construir tabelas de verdade (cada modelo de uma fórmula lógica corresponde a uma linha na tabela de verdade para essa fórmula cujo valor lógico para a fórmula é V).

Os modelos permitem definir o significado de uma fórmula lógica através de operações de conjuntos sobre conjuntos de modelos. Modelos de fórmulas lógicas complexas (proposições compostas) podem ser definidos em termos de modelos de fórmulas mais simples suas componentes, como se mostra na figura seguinte. Nesta figura, P e Q representam as fórmulas lógicas componentes e as partes sombreadas correspondem aos modelos da fórmula lógica complexa.



<u>Formas normais</u> – São formas padronizadas das fórmulas lógicas de cálculo proposicional (o número de formas em que uma mesma declaração pode ser expressa em lógica é mais restrito quando a formula lógica está numa forma normal). Certos sistemas de dedução necessitam que as fórmulas lógicas sejam convertidas para uma forma normal. No âmbito do cálculo proposicional as formas normais são a forma normal conjuntiva e a forma normal disjuntiva. Em ambos os casos, as fórmulas lógicas contêm apenas as conectivas lógicas $\wedge$, $\vee$ e $\neg$.

De forma resumida, para conversão de uma fórmula lógica de cálculo proposicional para uma forma normal empregam-se as regras de equivalência básicas, primeiro para eliminar as conectivas $\Leftrightarrow$ e $\Rightarrow$ que existam na fórmula, depois para que conectiva $\neg$ fique a afectar directamente apenas proposições simples e, por fim, usar as leis da distributividade de modo a concluir obtendo a respectiva forma normal.

<u>Forma normal conjuntiva</u> - Nesta forma, uma fórmula lógica é composta apenas por conjunções de disjunções de literais, esquematicamente:

$(L_{1,1} \vee ... \vee L_{1,k_1}) \wedge ... \wedge (L_{i,1} \vee ... \vee L_{i,k_i}) \wedge ... \wedge (L_{n,1} \vee ... \vee L_{n,k_n})$

em que os $L_{i,j}$ são literais.

<u>Forma clausal</u> – Forma normal conjuntiva representada através de um conjunto de <u>cláusulas</u> em conjunção implícita, em que as cláusulas são conjuntos de literais em disjunção implícita.

Os sistemas de dedução baseados no princípio da resolução (ver mais adiante) necessitam que o conjunto de fórmulas lógicas a usar seja convertido para esta forma. Nestes sistemas, a fórmula lógica acima apresentada de forma esquemática em forma normal conjuntiva é representada na forma de um conjunto de cláusulas C_i:

$\{ C_1, ..., C_i, ..., C_n \}$

entre as quais as conjunções estão implícitas e em que cada cláusula C_i é o conjunto:

$\{ L_{i,1} , ..., L_{i,k_i} \}$

que representa o componente $(L_{i,1} \vee ... \vee L_{i,k_i})$ da fórmula lógica (com disjunções implícitas entre os literais da cláusula).

<u>Forma normal disjuntiva</u> - Nesta forma, uma fórmula lógica é composta apenas por disjunções de conjunções de literais, esquematicamente:

$(L_{1,1} \wedge ... \wedge L_{1,k_1}) \vee ... \vee (L_{n,1} \wedge ... \wedge L_{n,k_n})$

5- Cálculo Proposicional – Dedução Natural

Sistemas de dedução natural - Sistemas de lógica em que não é requerido um preprocessamento extenso (por exemplo, passagem a uma forma normal) das fórmulas lógicas e em que as regras de inferência são intuitivas.

Regra de inferência (ou regra de dedução, ou silogismo) (2) – Padrão particular de geração de uma nova fórmula lógica (a conclusão) a partir de outras fórmulas lógicas (as premissas). O emprego de regras de inferência é um método para gerar consequências lógicas mais geral do que o método da tabela de verdade.

Descrição de uma regra de inferência:

(P_1, ..., P_n são premissas e G_1, ..., G_n, são conclusões):

$$\frac{\begin{array}{c}P_1\\ :\\ P_n\end{array}}{\begin{array}{c}G_1\\ :\\ G_m\end{array}}$$

Regras de inferência de dedução natural em cálculo proposicional:

Regras de Introdução

I.C. Introdução da Conjunção

$$\frac{\begin{array}{l}P\\ Q\end{array}}{P \land Q}$$

I.D. Introdução da Disjunção

$$\frac{P}{P \lor Q}$$

I.D.N. Introdução da Dupla Negação

$$\frac{P}{\neg\neg P}$$

Regras de Eliminação

E.C. Eliminação da Conjunção

$$\frac{P \land Q}{\begin{array}{l}P\\ Q\end{array}}$$

E.D. Eliminação da Disjunção

$$\frac{\begin{array}{l}P \lor Q\\ \neg P\end{array}}{Q}$$

E.D.N. Eliminação da Dupla Negação

$$\frac{\neg\neg P}{P}$$

M.P. Modus Ponens

$$\frac{\begin{array}{l}P \Rightarrow Q\\ P\end{array}}{Q}$$

M.T. Modus Tolens

$$\frac{\begin{array}{l}P \Rightarrow Q\\ \neg Q\end{array}}{\neg P}$$

Exemplos elementares de dedução natural em cálculo proposicional:

Exemplo elementar 1:

Poupo. Tenho dinheiro. Poupo e tenho dinheiro?

Frase em português	Proposição
Poupo	P
Tenho dinheiro	D

Dadas as premissas:

	Frase em português	Fórmula lógica
1	Poupo	P
2	Tenho dinheiro	D

Pretende-se provar que a seguinte conclusão é uma consequência lógica das premissas:

Frase em português	Fórmula lógica
Poupo e tenho dinheiro	P ∧ D

Como se prova P ∧ D (poupo e tenho dinheiro)?

	Frase em português	Fórmula lógica	
1	Poupo	P	
2	Tenho dinheiro	D	
3	**Poupo e tenho dinheiro**	**P ∧ D**	(1,2,I.C.)

Regra de inferência usada: *Introdução da Conjunção* (em 3)

Exemplo elementar 2:

Poupo e tenho dinheiro. Será que poupo?

Dada a premissa:

	Frase em português	Fórmula lógica
1	Poupo e tenho dinheiro	P ∧ D

Pretende-se provar que a seguinte conclusão é uma consequência lógica da premissa:

Frase em português	Fórmula lógica
Poupo	P

Como se prova P (poupo)?

	Frase em português	Fórmula lógica
1	Poupo e tenho dinheiro	P ∧ D
2	**Poupo**	**P** (1,E.C.)

Regra de inferência usada: *Eliminação da Conjunção* (em 2)

Exemplo elementar 3:

Poupo. Poupo ou durmo?

Frase em português	Proposição
Poupo	P
Durmo	S

Dadas a premissa:

	Frase em português	Fórmula lógica
1	Poupo	P

Pretende-se provar que a seguinte conclusão é uma consequência lógica da premissa:

Frase em português	Fórmula lógica
Poupo ou durmo	P ∨ S

Como se prova P ∨ S (poupo ou durmo)?

	Frase em português	Fórmula lógica
1	Poupo	P
2	**Poupo ou durmo**	**P ∨ S** (1,I.D.)

Regra de inferência usada: *Introdução da Disjunção* (em 2)

Exemplo elementar 4:

Poupo ou durmo. Não poupo. Será que durmo?

Dadas as premissas:

	Frase em português	Fórmula lógica
1	Poupo ou durmo	P ∨ S
2	Não poupo	¬P

Pretende-se provar que a seguinte conclusão é uma consequência lógica das premissas:

Frase em português	Fórmula lógica
Durmo	S

Como se prova S (durmo)?

	Frase em português	Fórmula lógica	
1	Poupo ou durmo	P ∨ S	
2	Não poupo	¬P	
3	**Durmo**	**S**	(1,2,E.D.)

Regra de inferência usada: *Eliminação da Disjunção* (em 3)

<u>Exemplo elementar 5</u>:

Poupo. Será que não é verdade que não poupo?

Frase em português	Proposição
Poupo	P

Dadas a premissa:

	Frase em português	Fórmula lógica
1	Poupo	P

Pretende-se provar que a seguinte conclusão é uma consequência lógica da premissa:

Frase em português	Fórmula lógica
Não é verdade que não poupo	¬¬P

Como se prova ¬¬P (não é verdade que não poupo)?

	Frase em português	Fórmula lógica	
1	Poupo	P	
2	**Não é verdade que não poupo**	**¬¬P**	(1,I.D.N.)

Regra de inferência usada: *Introdução da Dupla Negação* (em 2)

<u>Exemplo elementar 6</u>:

Não é verdade que não gasto. Será que gasto?

Frase em português	Proposição
Gasto	G

Dadas a premissa:

	Frase em português	Fórmula lógica
1	Não é verdade que não gasto	$\neg\neg G$

Pretende-se provar que a seguinte conclusão é uma consequência lógica da premissa:

Frase em português	Fórmula lógica
Gasto	G

Como se prova G (gasto)?

	Frase em português	Fórmula lógica	
1	Não é verdade que não gasto	$\neg\neg G$	
2	**Gasto**	$\mathbf{G}$	(1,E.D.N.)

Regra de inferência usada: *Eliminação da Dupla Negação* (em 2)

<u>Exemplo elementar 7</u>:

Se poupo, então tenho dinheiro. Eu poupo. Tenho dinheiro?

Frase em português	Proposição
Poupo	P
Tenho dinheiro	D

Dadas as premissas:

	Frase em português	Fórmula lógica
1	Se poupo, então tenho dinheiro	$P \Rightarrow D$
2	Poupo	P

Pretende-se provar que a seguinte conclusão é uma consequência lógica das premissas:

Frase em português	Fórmula lógica
Tenho dinheiro	D

Como se prova D (tenho dinheiro)?

	Frase em português	Fórmula lógica	
1	Se poupo, então tenho dinheiro	$P \Rightarrow D$	
2	Poupo	P	
3	**Tenho dinheiro**	$\mathbf{D}$	(1,2,M.P.)

Regra de inferência usada: *Modus Ponens* (em 3)

Exemplo elementar 8:

Se poupo, então tenho dinheiro. Não tenho dinheiro. Será que não poupo?

Dadas as premissas:

	Frase em português	Fórmula lógica
1	Se poupo, então tenho dinheiro	$P \Rightarrow D$
2	Não tenho dinheiro	$\neg D$

Pretende-se provar que a seguinte conclusão é uma consequência lógica das premissas:

Frase em português	Fórmula lógica
Não poupo	$\neg P$

Como se prova ¬P (não poupo)?

	Frase em português	Fórmula lógica
1	Se poupo, então tenho dinheiro	$P \Rightarrow D$
2	Não tenho dinheiro	$\neg D$
3	**Não poupo**	$\neg \mathbf{P}$ (1,2,M.T.)

Regra de inferência usada: *Modus Tolens* (em 3)

Outros exemplos de dedução natural em cálculo proposicional:

Exemplo 1:

Se é humano ou símio, então é mamífero. Sabe-se que é símio. Será mamífero?

Frase em português	Proposição
É humano	H
É símio	S
É mamífero	M

Dadas as premissas:

	Frase em português	Fórmula lógica
1	Se é humano ou símio, então é mamífero	$H \vee S \Rightarrow M$
2	É símio	S

Pretende-se provar que a seguinte conclusão é uma consequência lógica das premissas:

Frase em português	Fórmula lógica
É mamífero	M

Como se prova M (é mamífero)?

	Frase em português	Fórmula lógica	
3	É humano ou símio	$H \lor S$	(2,I.D.)
4	**É mamífero**	$\mathbf{M}$	(1,3,M.P.)

Regras de inferência usadas: *Introdução da Disjunção* (em 3)
Modus Ponens (em 4)

<u>Exemplo 2</u>:

Se é humano ou símio, então é mamífero. Não é mamífero. Será que não é símio?

Dadas as premissas:

	Frase em português	Fórmula lógica
1	Se é humano ou símio, então é mamífero	$H \lor S \Rightarrow M$
2	Não é mamífero	$\neg M$

Pretende-se provar que a seguinte conclusão é uma consequência lógica das premissas:

Frase em português	Fórmula lógica
Não é símio	$\neg S$

Como se prova ¬S (não é símio)?

	Frase em português	Fórmula lógica	
3	Não é humano e não é símio	$\neg H \land \neg S$	(1,2,M.T.)
4	**Não é símio**	$\mathbf{\neg S}$	(3,E.C.)

Regras de inferência usadas: *Modus Tolens* (em 3)
Eliminação da Conjunção (em 4)

<u>Exemplo 3</u>:

Sabe-se que se está quente e está húmido, então choverá e que se está húmido, então está quente. Sabe-se ainda que está húmido. Pretende-se concluir que: a) Está quente; b) Choverá.

Frase em português	Proposição
Está quente	Q
Está húmido	H
Choverá	C

Dadas as premissas:

	Frase em português	Fórmula lógica
1	Se está quente e está húmido, então choverá	$Q \land H \Rightarrow C$
2	Se está húmido, então está quente	$H \Rightarrow Q$
3	Está húmido	H

Pretende-se provar que as seguintes conclusões são consequências lógicas das premissas:

Frase em português	Fórmula lógica
Está quente	Q
Choverá	C

a) Como se prova Q (está quente)?

	Frase em português	Fórmula lógica	
4	**Está quente**	**Q**	(2,3,M.P.)

Regra de inferência usada: *Modus Ponens* (em 4)

b) Como se prova C (choverá)?

	Frase em português	Fórmula lógica	
4	Está quente	Q	(2,3,M.P.)
5	Está quente e está húmido	$Q \wedge H$	(4,3,I.C.)
6	**Choverá**	**C**	(1,5,M.P.)

Regras de inferência usadas: *Modus Ponens* (em 4 e 6)
Introdução da Conjunção (em 5)

Exemplo 4:

Suponha-se que o preço das acções diminui se a taxa de juro aumenta. Suponha-se também que há pessoas descontentes se o preço das acções diminui. Admitindo que a taxa de juro aumenta, mostrar que há pessoas descontentes.

Frase em português	Proposição
A taxa de juro aumenta	J
O preço das acções diminui	A
Há pessoas descontentes	D

Dadas as premissas:

	Frase em português	Fórmula lógica
1	Se a taxa de juro aumenta, então o preço das acções diminui	$J \Rightarrow A$
2	Se o preço das acções diminui, então há pessoas descontentes	$A \Rightarrow D$
3	A taxa de juro aumenta	J

Pretende-se provar que as seguinte conclusão é uma consequência lógica das premissas:

Frase em português	Fórmula lógica
Há pessoas descontentes	D

Como se prova D (há pessoas descontentes)?

	Frase em português	Fórmula lógica
4	O preço das acções diminui	A (1,3,M.P.)
5	**Há pessoas descontentes**	**D** (2,4,M.P.)

Regra de inferência usada: *Modus Ponens* (em 4 e 5)

Exemplo 5:

Suponha-se que se conhecem as seguintes reacções químicas:

$MgO + H_2 \rightarrow Mg + H_2O$ $C + O_2 \rightarrow CO_2$ $CO_2 + H_2O \rightarrow H_2CO_3$

Suponha-se ainda que dispomos de quantidades de MgO, de H_2, de O_2 e de C. Será que poderemos produzir H_2CO_3?

Frase	Proposição
MgO	MgO
H_2	H2
Mg	Mg
H_2O	H2O
C	C
O_2	O2
CO_2	CO2
H_2CO_3	H2CO3

Dadas as premissas:

	Frase	Fórmula lógica
1	$MgO + H_2 \rightarrow Mg + H_2O$	MgO ∧ H2 ⇒ Mg ∧ H2O
2	$C + O_2 \rightarrow CO_2$	C ∧ O2 ⇒ CO2
3	$CO_2 + H_2O \rightarrow H_2CO_3$	CO2 ∧ H2O ⇒ H2CO3
4	MgO	MgO
5	H_2	H2
6	O_2	O2
7	C	C

Pretende-se provar que as seguinte conclusão é uma consequência lógica das premissas:

Frase	Fórmula lógica
H_2CO_3	H2CO3

Como se prova H2CO3 (H_2CO_3)?

	Frase em português	Fórmula lógica	
8	$C + O_2$	C ∧ O2	(7,6,I.C.)
9	CO_2	CO2	(2,8,M.P.)
10	$MgO + H_2$	MgO ∧ H2	(4,5,I.C.)
11	$Mg + H_2O$	Mg ∧ H2O	(1,10,M.P.)
12	H_2O	H2O	(11,E.C.)
13	$CO_2 + H_2O$	CO2 ∧ H2O	(9,12,I.C.)
14	$\mathbf{H_2CO_3}$	**H2CO3**	(3,13,M.P.)

Regras de inferência usadas: *Introdução da Conjunção* (em 8, 10 e 13)
Modus Ponens (em 9, 11 e 14)
Eliminação da Conjunção (em 12)

<u>Exemplo 6</u>:

Dadas as premissas:

1	P ⇒ (¬Q ∨ (R ∧ S))
2	P
3	¬S

Pretende-se concluir ¬Q.

Como se prova ¬Q?

4	(¬Q ∨ R) ∧ (¬Q ∨ S)	(1,2,M.P.)
5	¬Q ∨ S	(4,E.C.)
6	**¬Q**	(3,5,E.D.)

Regras de inferência usadas: *Modus Ponens* (em 4)
Eliminação da Conjunção (em 5)
Eliminação da Disjunção (em 6)

6- Cálculo Proposicional – O Princípio da Resolução

Ao contrário dos sistemas de dedução natural, os sistemas de dedução baseados em resolução usam uma única regra de inferência, denominada regra da resolução, ou princípio da resolução. No entanto (e também ao contrário de no caso dos sistemas de dedução natural) as fórmulas lógicas tomadas como premissas têm de passar por um pré-processamento de modo a serem convertidas para a forma clausal.

Na descrição que se segue do princípio da resolução, as fórmulas lógicas tomadas como premissas são expressas na forma clausal, *i.e.*, cada fórmula lógica é expressa na forma de uma cláusula. Lembra-se que uma cláusula equivale a uma fórmula lógica representada na forma de um conjunto de literais em disjunção implícita, quer dizer, na forma:

$\{ L_1 , ..., L_j , ..., L_n \}$

em que os L_j são literais. Lembra-se também que esta forma representa a fórmula lógica (de forma normal conjuntiva):

$(L_1 \vee ... \vee L_j \vee ... \vee L_n)$

De uma forma geral, sendo A_h e B_k literais complementares (*i.e.*, um deles é igual à negação do outro), a regra da resolução pode descrever-se da seguinte forma:

$$\frac{\{ A_1, ..., A_{h-1}, A_h, A_{h+1}, ..., A_n \} \quad \{ B_1, ..., B_{k-1}, B_k, B_{k+1}, ..., B_m \}}{\{ A_1, ..., A_{h-1}, A_{h+1}, ..., A_n, B_1, ..., B_{k-1}, B_{k+1}, ..., B_m \}}$$

A cláusula de conclusão $\{ A_1, ..., A_{h-1}, A_{h+1}, ..., A_n, B_1, ..., B_{k-1}, B_{k+1}, ..., B_m \}$ é, então, apelidada resolvente das cláusulas $\{ A_1, ..., A_{h-1}, A_h, A_{h+1}, ..., A_n \}$ e $\{ B_1, ..., B_{k-1}, B_k, B_{k+1}, ..., B_m \}$.

Exemplos elementares de aplicação da regra da resolução:

Exemplo elementar 1:

1	{ ¬P, Q }	
2	{ P }	
3	**{ Q }**	de 1 e 2

Nota: Este exemplo elementar corresponde, no caso dos sistemas de dedução natural, ao emprego do Modus Ponens, já que a cláusula `{ ¬P, Q }` representa a fórmula lógica ¬P∨Q, que equivale logicamente à fórmula P⇒Q (pela equivalência lógica apelidada de eliminação da implicação), permitindo, então, de P⇒Q e de P deduzir Q.

Exemplo elementar 2:

1	{ ¬P, Q }	
2	{ ¬Q }	
3	**{ ¬P }**	de 1 e 2

Nota: Este exemplo elementar corresponde, no caso dos sistemas de dedução natural, ao emprego do Modus Tolens, já que a cláusula `{ ¬P, Q }` representa a fórmula lógica

¬P∨Q, que equivale logicamente à fórmula P⇒Q (pela equivalência lógica apelidada de eliminação da implicação), permitindo, então, de P⇒Q e de ¬Q deduzir ¬P.

Exemplo elementar 3:

1	{ ¬P, Q }	
2	{ ¬Q, R }	
3	**{ ¬P, R }**	de 1 e 2

Nota: Este exemplo elementar corresponde, no caso de alguns sistemas de dedução, ao emprego de uma regra denominada regra do encadeamento, já que as cláusulas { ¬P, Q } e { ¬Q, R } representam, respectivamente, as fórmulas lógicas ¬P∨Q e ¬Q∨R, que equivalem logicamente às fórmulas P⇒Q e Q⇒R, respectivamente (pela equivalência lógica apelidada de eliminação da implicação), permitindo, então, de P⇒Q e de Q⇒R deduzir P⇒R.

Exemplo elementar 4:

1	{ ¬P, Q }	
2	{ P, Q }	
3	**{ Q }**	de 1 e 2

Exemplo elementar 5:

1	{ ¬P, Q, R }	
2	{ P, S }	
3	**{ Q, R, S }**	de 1 e 2

Para provar que uma dada fórmula lógica P é uma consequência lógica de um conjunto de fórmulas lógicas tomadas como premissas dado opera-se de maneira diferente do que para os sistemas de dedução natural, de acordo com o seguinte:

1- Todas as fórmulas lógicas têm de ser convertidas para a forma normal conjuntiva e representadas na forma clausal.
2- A prova é uma prova por refutação: em vez de derivar P directamente do conjunto de fórmulas lógicas tomadas como premissas, prova-se que ¬P é inconsistente com esse conjunto de fórmulas (*i.e.*, que ¬P em conjunção com esse conjunto de fórmulas é insatisfiável).

O processo geral de prova manual de uma fórmula P quando se emprega a regra da resolução é, então:

1- Converter o conjunto de fórmulas lógicas tomadas como premissas para a forma clausal.
2- Negar P.
3- Converter ¬P para a forma clausal.
4- Juntar as cláusulas de ¬P ao conjunto de fórmulas tomadas como premissas.
5- Empregando a regra da resolução, tentar derivar a cláusula vazia (*i.e.*, a cláusula { }).

Se o passo 5- tem sucesso, então P é uma consequência lógica do conjunto de fórmulas tomadas como premissas, caso contrário, não o é.

<u>Conversão para a forma clausal</u> – A conversão de uma fórmula lógica para a correspondente forma clausal pode ser realizada por um processo sistemático em seis fases:

1- Remover equivalências e implicações.
2- Mover negações para dentro, de modo a afectem apenas as proposições simples.
3- Passagem à forma normal conjuntiva, de modo a obter conjunções de disjunções de literais.
4- Colocar em forma de cláusulas (cada cláusula é um conjunto de literais em disjunção implícita que corresponde a um dos disjuntos da fase anterior).

<u>Exemplos</u> de conversão para a forma clausal (explicita-se cada uma das fases de conversão):

<u>Exemplo 1</u>: Converter para forma clausal a fórmula lógica $P \land S \Leftrightarrow Q$.

Resposta:

1	$(\neg(P \land S) \lor Q) \land (\neg Q \lor (P \land S))$
2	$(\neg P \lor \neg S \lor Q) \land (\neg Q \lor (P \land S))$
3	$(\neg P \lor \neg S \lor Q) \land (\neg Q \lor P) \land (\neg Q \lor S)$
4	**{ ¬P, ¬S, Q }** **{ ¬Q, P }** **{ ¬Q, S }**

<u>Exemplo 2</u>: Converter para forma clausal a fórmula lógica $P \land Q \Rightarrow \neg(R \Rightarrow S)$.

Resposta:

1	$\neg(P \land Q) \lor \neg(\neg R \lor S)$
2	$\neg P \lor \neg Q \lor (R \land \neg S)$
3	$(\neg P \lor \neg Q \lor R) \land (\neg P \lor \neg Q \lor \neg S)$
4	**{ ¬P, ¬Q, R }** **{ ¬P, ¬Q, ¬S }**

<u>Exemplo 3</u>: Converter para forma clausal a fórmula lógica $P \Rightarrow \neg(Q \Rightarrow R)$.

Resposta:

1	$\neg P \lor \neg(\neg Q \lor R)$
2	$\neg P \lor (Q \land \neg R)$
3	$(\neg P \lor Q) \land (\neg P \lor \neg R)$
4	**{ ¬P, Q }** **{ ¬P, ¬R }**

Exemplos de dedução empregando resolução:

Exemplo 1: Considere o seguinte conjunto de cláusulas composto pelas cláusulas 1 a 3 (todas em forma clausal):

1	{ ¬F, G }
2	{ ¬G, H }
3	{ F }

Empregando a regra da resolução e justificando cada passo de prova com os números das cláusulas usadas, prove por refutação H.

Resposta:

4	{ ¬H }

5	{ G }	de 1 e 3
6	{ H }	de 2 e 5
7	**{ }**	de 4 e 6

Exemplo 2: Considere o seguinte conjunto de cláusulas composto pelas cláusulas 1 a 4 (todas em forma clausal):

1	{ ¬A, ¬B, C }
2	{ ¬A, ¬B, D }
3	{ A }
4	{ ¬D }

Empregando a regra da resolução e justificando cada passo de prova com os números das cláusulas usadas, prove por refutação ¬B.

Resposta:

5	{ B }

6	{ ¬B, D }	de 2 e 3
7	{ D }	de 5 e 6
8	**{ }**	de 4 e 7

7- Cálculo de Predicados – Sintaxe

No cálculo proposicional, os elementos mais simples, as proposições simples, ou átomos, são tratados como uma unidade indivisível, sem estrutura ou composição própria. Isto é limitativo na expressão de frases lógicas. Por exemplo, considere-se o seguinte raciocínio, intuitivamente correcto:

Todo o humano é mortal.
Dado que Confúcio é humano, Confúcio é mortal.

No entanto, ao tentar representar estas frases usando cálculo proposicional:

Frase em português	Proposição
Todo o humano é mortal	P
Confúcio é humano	Q
Confúcio é mortal	R

Percebe-se que R não é consequência lógica de P e Q no âmbito do cálculo proposicional. Isto acontece porque a estrutura interna de P, de Q e de R não é representável no cálculo proposicional.

O cálculo de predicados tem mais as noções lógicas de termo, de predicado e de quantificador, além das do cálculo proposicional.

Termos – Significam objectos de um domínio de discurso. São definidos do seguinte modo:

1- Uma constante é um termo.
2- Uma variável é um termo.
3- $F(t_1, ..., t_n)$ é um termo, se F é um símbolo de função de n argumentos e $t_1, ..., t_n$ são termos.
4- Todos os termos são gerados aplicando estas regras.

Termo básico – Termo que não contém variáveis.

Funções – Estabelecem uma correspondência entre uma lista de constantes (os argumentos, cada um valor de determinado domínio) e uma constante (valor do contradomínio).

Fórmula atómica (ou átomo) – $P(t_1, ..., t_n)$ é uma fórmula atómica, se P é um símbolo de predicado de n argumentos e $t_1, ..., t_n$ são termos. Nada mais é uma fórmula atómica.

Literal – Uma fórmula atómica (literal positivo) ou uma fórmula atómica negada (literal negativo).

Predicados - Significam declarações elementares que exprimem propriedades de, ou relações entre, objectos de domínios de discurso. Estabelecem uma correspondência entre uma lista de constantes (os argumentos, cada um valor de um determinado domínio) e os valores lógicos V ou F.

Quantificadores – Conectivas lógicas de cálculo de predicados que servem para caracterizar variáveis em fórmulas lógicas. Podem ser:

$\forall$ – Quantificador universal (*para todos*, ou *para cada*). Exemplo: $\forall x$.
$\exists$ – Quantificador existencial (*existe um*, ou *existe pelo menos um*). Exemplo: $\exists x$.

Ocorrência de variável ligada – Diz-se da ocorrência de uma variável numa fórmula lógica quando essa ocorrência está dentro do âmbito de um quantificador que emprega a variável.

Ocorrência de variável livre (ou não ligada) – Diz-se da ocorrência de uma variável numa fórmula lógica quando essa ocorrência não é ligada.

Variável livre (ou não ligada, ou não quantificada) numa fórmula lógica – Diz-se quando pelo menos uma ocorrência da variável é livre na fórmula lógica.

Variável ligada (ou quantificada) numa fórmula lógica – Diz-se quando pelo menos uma ocorrência da variável é ligada na fórmula lógica.

Conectiva lógica (de cálculo de predicados) – Símbolo que denota uma operação lógica. Pode ser (por ordem de prioridade):

1º	$\neg$	negação	(*não*)			
2º	$\wedge$	conjunção	(*e*)	$\vee$	disjunção	(*ou*)
3º	$\Rightarrow$	implicação	(*se ..., então ...*)	$\Leftrightarrow$	equivalência	(*... se e só se ...*)
4º	$\forall$	quant. universal	(*para todos ...*)	$\exists$	quant. existencial	(*existe um ...*)

Sintaxe do cálculo de predicados

Fórmulas lógicas bem formadas:

1- Uma fórmula atómica (ou átomo) é uma fórmula.
2- Se G e H são fórmulas, então $\neg$G, G $\wedge$ H, G $\vee$ H, G $\Rightarrow$ H e G $\Leftrightarrow$ H são fórmulas.
3- Se G é uma fórmula e x é uma variável livre em G, então $\forall$x G e $\exists$x G são fórmulas.
4- Todas as fórmulas são geradas aplicando as regras 1-, 2- e 3- um número finito de vezes.

Nota: Numa fórmula lógica podem usar-se os parêntesis, os símbolos (e), devidamente emparelhados, para alterar a ordem de prioridade das conectivas.

Mais formalmente, a gramática do cálculo de predicados é a seguinte:

```
Fórmula          = FórmulaAtómica
                   | Fórmula Conectiva Fórmula
                   | Quantificador Variável ...  Fórmula
                   | ¬ Fórmula
                   | ( Fórmula )
FórmulaAtómica   = Predicado(Termo, ...)
Termo            = Função(Termo, ...)  | Constante | Variável
Conectiva        =   ⇒   | ∧   |  ∨  |   ⇔
Quantificador    =   ∀  |  ∃
Constante        =   A  |  X | Xpto  | Z31p  | ...
Variável         =   a  |  x | spt   | v3    | ...
Predicado        =   P  |  Q | Rzt | Xp7 | ...
Função           =   F  |  G | Lbt | F3    | ...
```

Exemplos de funções:

Significado em português	Função (termo funcional)
O valor que é o dobro de 34	`Dobro(34)`
A área de um rectângulo com lados de comprimento b e h	`ÁreaRectângulo(b,h)`
A mãe de Ana	`Mãe(Ana)`
A mãe de x	`Mãe(x)`

Exemplos de predicados:

Significado em português	Predicado (fórmula atómica)
x é gordo	`Gordo(x)`
y é humano	`Humano(y)`
Confúcio é humano	`Humano(Confúcio)`
z é mortal	`Mortal(z)`
x é maior que y	`Maior(x,y)`
João ama Maria	`Ama(João,Maria)`
A mãe de Maria ama Maria	`Ama(Mãe(Maria),Maria)`
A distância entre Lisboa e Porto é 300)	`Distância(Lisboa,Porto,300)`

Exemplos de fórmulas lógicas:

Significado em português	Fórmula lógica
Todo o humano é mortal	$\forall x$ Humano(x) $\Rightarrow$ Mortal(x)
Alguns indivíduos são altos e gordos	$\exists y$ Alto(y) $\wedge$ Gordo(y)
Há sempre valor maior que qualquer outro	$\forall x\ \exists y$ Maior(y,x)
Existe um valor maior que quaisquer outros	$\exists y\ \forall x$ Maior(y,x)

8- Cálculo de Predicados - Semântica

Como se determina o valor lógico de uma fórmula?

Interpretação – Consiste em definir o domínio de discurso e:

1- A cada constante, associar um elemento do domínio de discurso.
2- A cada símbolo de função de n argumentos, associar uma correspondência entre tuplos de n valores de argumentos do domínio de discurso e o domínio de discurso.
3- A cada símbolo de predicado de n argumentos, associar uma correspondência entre tuplos de n valores de argumentos do domínio de discurso e o domínio { V, F }.

Numa interpretação de uma fórmula o valor lógico, V ou F, obtém-se usando as regras:

1- Com o valor lógico das fórmulas atómicas determinado, o valor lógico das fórmulas não atómicas é determinado por meio da tabela de verdade das conectivas lógicas.
2- Uma fórmula $\forall$x G tem o valor V se o valor lógico de G é V para toda constante do domínio de discurso. Caso contrário tem o valor F.
3- Uma fórmula $\exists$x G tem o valor V se o valor lógico de G é V pelo menos para uma constante do domínio de discurso. Caso contrário tem o valor F.

Nota: Formalmente, o valor lógico de fórmulas contendo variáveis livres não se pode determinar. Normalmente, ou se assume que não existem variáveis livres, ou então elas são tratadas como sendo quantificadas universalmente.

Tabelas de verdade das conectivas lógicas (V = VERDADEIRO, F = FALSO):

G	$\neg$G
V	F
F	V

G	H	G $\wedge$ H	G $\vee$ H	G $\Rightarrow$ H	G $\Leftrightarrow$ H
V	V	V	V	V	V
V	F	F	V	F	F
F	V	F	V	V	F
F	F	F	F	V	V

Nota: Idêntico às tabelas de verdade para o cálculo proposicional.

Exemplo de interpretação para a fórmula: $\forall x\ P(x) \Rightarrow Q(F(x),A)$

Existem nesta fórmula: uma constante A, um símbolo de função de um argumento F, um símbolo de predicado de um argumento P e um símbolo de predicado de dois argumentos, Q. Uma interpretação I possível é:

Domínio: D = { 1, 2 }

Associações:

A
1

F(1)	F(2)
2	1

P(1)	P(2)
F	V

Q(1,1)	Q(1,2)	Q(2,1)	Q(2,2)
V	V	F	F

Se x = 1, então: $P(x) \Rightarrow Q(F(x),A) \equiv P(1) \Rightarrow Q(F(1),A)$
$\equiv P(1) \Rightarrow Q(2,1)$
$\equiv F \Rightarrow F \quad \equiv V$

Se x = 2, então: $P(x) \Rightarrow Q(F(x),A) \equiv P(2) \Rightarrow Q(F(2),A)$
$\equiv P(2) \Rightarrow Q(1,1)$
$\equiv V \Rightarrow V \quad \equiv V$

A fórmula $\forall x\ P(x) \Rightarrow Q(F(x),A)$ é verdadeira na interpretação I, visto ter o valor lógico V para todos os elementos x do domínio D.

<u>Fórmula consistente</u> (ou satisfiável) – Fórmula para a qual existe uma interpretação sob a qual a fórmula é verdadeira.

Diz-se que uma interpretação I <u>satisfaz</u> a fórmula P, ou que P <u>é satisfeita</u> por I, se P é verdadeira sob I.

Diz-se que uma interpretação I é um <u>modelo</u> da fórmula P, se I satisfaz P.

<u>Fórmula válida</u> – Fórmula que é satisfeita para todas as suas interpretações.

<u>Fórmula inconsistente</u> (ou insatisfiável, ou não satisfiável) – Fórmula para a qual não existe interpretação alguma que a satisfaz.

<u>Consequência lógica</u> (3) – Uma fórmula G é consequência lógica das fórmulas P_1, P_2,.., P_n se e só se, para qualquer interpretação I, se $P_1 \wedge P_2 \wedge ... \wedge P_n$ é verdadeira sob I, G é também verdadeira sob I.

<u>Equivalência Lógica</u> – Para operar sobre fórmulas lógicas de cálculo de predicados e transformá-las em fórmulas equivalentes existe um conjunto de equivalências básicas, exposto na seguinte tabela de equivalências (P, Q, R, P_x e Q_x são fórmulas e V e F os valores VERDADEIRO e FALSO, respectivamente):

equivalências

1		$P \Leftrightarrow Q \equiv (P \Rightarrow Q) \wedge (Q \Rightarrow P)$			eliminação da equivalência
2		$P \Rightarrow Q \equiv \neg Q \Rightarrow \neg P$			contraposição
3		$P \Rightarrow Q \equiv \neg P \vee Q$			eliminação da implicação
4	a)	$P \vee Q \equiv Q \vee P$	b)	$P \wedge Q \equiv Q \wedge P$	comutatividade
5	a)	$(P \vee Q) \vee R \equiv P \vee (Q \vee R)$	b)	$(P \wedge Q) \wedge R \equiv P \wedge (Q \wedge R)$	associatividade
6	a)	$P \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R)$	b)	$P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$	distributividade
7	a)	$P \vee F \equiv P$	b)	$P \wedge V \equiv P$	elemento neutro
8	a)	$P \vee V \equiv V$	b)	$P \wedge F \equiv F$	elemento absorvente
9	a)	$P \vee \neg P \equiv V$	b)	$P \wedge \neg P \equiv F$	tautologia, contradição
10		$\neg\neg P \equiv P$			dupla negação
11	a)	$\neg(P \vee Q) \equiv \neg P \wedge \neg Q$	b)	$\neg(P \wedge Q) \equiv \neg P \vee \neg Q$	leis de de Morgan
12	a)	$\neg(\forall x\ P(x)) \equiv \exists x\ \neg P(x)$	b)	$\neg(\exists x\ P(x)) \equiv \forall x\ \neg P(x)$	negação da
	c)	$\forall x\ P(x) \equiv \neg(\exists x\ \neg P(x))$	d)	$\exists x\ P(x) \equiv \neg(\forall x\ \neg P(x))$	quantificação
13	a)	$(Kx\ P(x)) \vee Q \equiv Kx\ P(x) \vee Q$	b)	$(Kx\ P(x)) \wedge Q \equiv Kx\ P(x) \wedge Q$	quantificação da
14	a)	$(K_1x\ P(x)) \vee (K_2x\ Q(x)) \equiv K_1x\ K_2y\ P(x) \vee Q(y)$	b)	$(K_1x\ P(x)) \wedge (K_2x\ Q(x)) \equiv K_1x\ K_2y\ P(x) \wedge Q(y)$	conjunção e da disjunção

<u>Nota</u>: À excepção das últimas três equivalências, 12 a 14, cada uma das restantes corresponde a uma equivalência aplicável no cálculo proposicional. Naquelas, `P(x)` e `Q(x)` significam fórmulas contendo a variável `x`, `Q` uma fórmula não contendo a variável `x`, e `K`, `K1` e `K2` significam quantificadores (∀ ou ∃).

<u>Semântica e teoria dos modelos</u> – Tal como se disse para o caso do cálculo proposicional, também no caso do cálculo de predicados os modelos permitem definir o significado de uma fórmula lógica através de operações de conjuntos sobre conjuntos de modelos. Modelos de fórmulas lógicas complexas (fórmula não atómicas) podem ser definidos em termos de modelos de fórmulas mais simples suas componentes, de forma similar à que foi exposta em

figura para o caso do cálculo proposicional (para o cálculo de predicados, o caso de as fórmulas lógicas conterem predicados de um único argumento é o mais intuitivo).

Formas normais – São formas padronizadas das fórmulas lógicas de cálculo de predicados (o número de formas em que uma mesma declaração pode ser expressa em lógica é mais restrito quando a formula lógica está numa forma normal). Tal como se disse para o caso do cálculo proposicional, certos sistemas de dedução necessitam que as fórmulas lógicas sejam convertidas para uma forma normal. No âmbito do cálculo de predicados as formas normais são a forma normal conjuntiva, a forma normal disjuntiva e a forma normal prenex. Nos dois primeiros casos, as fórmulas lógicas contêm apenas as conectivas lógicas $\wedge$, $\vee$ e $\neg$ (não contêm conectivas $\Leftrightarrow$, $\Rightarrow$, $\forall$ nem $\exists$). No terceiro caso (forma normal prenex) as fórmulas caracterizam-se por terem os quantificadores todos (em caso de existirem quantificadores na fórmula original) no início da fórmula.

Forma normal conjuntiva - Nesta forma, uma fórmula lógica é composta apenas por conjunções de disjunções de literais, esquematicamente:

$(L_{1,1} \vee ... \vee L_{1,k_1}) \wedge ... \wedge (L_{i,1} \vee ... \vee L_{i,k_i}) \wedge ... \wedge (L_{n,1} \vee ... \vee L_{n,k_n})$

em que os $L_{i,j}$ são literais (eventualmente contendo variáveis).

Forma clausal – Forma normal conjuntiva representada através de um conjunto de cláusulas em conjunção implícita, em que as cláusulas são conjuntos de literais em disjunção implícita.

Os sistemas de dedução baseados no princípio da resolução (ver mais adiante) necessitam que o conjunto de fórmulas lógicas a usar seja convertido para esta forma. Nestes sistemas, a fórmula lógica acima apresentada de forma esquemática em forma normal conjuntiva é representada na forma de um conjunto de cláusulas C_i:

$\{ C_1, ..., C_i, ..., C_n \}$

entre as quais as conjunções estão implícitas e em que cada cláusula C_i é o conjunto:

$\{ L_{i,1} , ..., L_{i,k_i} \}$

que representa o componente $(L_{i,1} \vee ... \vee L_{i,k_i})$ da fórmula lógica (com disjunções implícitas entre os literais da cláusula).

Forma normal disjuntiva - Nesta forma, uma fórmula lógica é composta apenas por disjunções de conjunções de literais, esquematicamente:

$(L_{1,1} \wedge ... \wedge L_{1,k_1}) \vee ... \vee (L_{n,1} \wedge ... \wedge L_{n,k_n})$

De forma resumida, para conversão de uma fórmula lógica de cálculo de predicados para uma forma normal conjuntiva ou disjuntiva empregam-se as regras de equivalência básicas, primeiro para eliminar as conectivas $\Leftrightarrow$ e $\Rightarrow$ que existam na fórmula, depois para que conectiva $\neg$ fique a afectar directamente apenas proposições simples; após isso são eliminados os quantificadores que existam na fórmula; por fim, usam-se as leis da distributividade de modo a concluir obtendo a respectiva forma normal.

Forma Normal Prenex – Forma em que a fórmula lógica tem os quantificadores todos no início. Uma fómula de cálculo de predicados $P_{x_1,...x_n}$ diz-se estar em forma normal prenex se e só se tem a forma:

$K_1x_1 ... K_nx_n P_{x_1,...x_n}$

em que cada K_ix_i ou é $\forall x_i$ ou é $\exists x_i$. A forma normal prenex é usada para converter as fórmulas, de modo a facilitar as provas. A conversão de uma fórmula para a forma normal prenex pode ser realizada empregando as equivalências 12 a 14 da tabela de equivalências anterior.

9- Cálculo de Predicados – Dedução Natural

A dedução natural no cálculo de predicados é semelhante à do cálculo proposicional à excepção de um processo adicional chamado proposicionalização, em que as variáveis nas fórmulas lógicas são substituídas por constantes apropriadas com o objectivo de poder aplicar regras de inferência. Estas substituições têm a finalidade de tornar certos termos e fórmulas atómicas literalmente idênticos para aplicação de uma regra de inferência.

Por detrás destas substituições para tornar fórmulas atómicas e termos idênticos está um processo mais geral, que se emprega, sempre que apropriado, ao aplicar regras de inferência. O processo referido é usado não apenas nos sistemas de dedução natural, onde é menos sofisticado, mas também nos sistemas baseados em resolução, que dele mais dependem e onde ele é aplicado de forma mais explícita. Descreve-se a seguir este, que se chama unificação.

Unificação – Processo que tenta realizar emparelhamento de duas fórmulas atómicas, ou dois termos, procurando torná-los idênticos, com ou sem sucesso. Em caso de sucesso no emparelhamento, a unificação produz um conjunto (eventualmente vazio) de associações de variáveis de cada um dos termos a termos contidos no outro termo, denominado substituição. Em caso de existir mais de uma substituição possível, esta substituição deve ser a que é a substituição mais geral.

Emparelhamento – Diz-se que dois termos, ou duas fórmulas atómicas emparelham se:

1- São idênticos, ou
2- As variáveis contidas em cada um deles podem ser associadas a termos contidos no outro termo de modo que, fazendo as substituições das variáveis pelos termos associados, os dois ficam idênticos.

O emparelhamento de duas fórmulas atómicas ou de dois termos que idênticos que não contenham variáveis terá sucesso sem associações.

Substituição – Conjunto finito de associações de variáveis a termos em que:

a) Cada variável é associada, no máximo, a um termo, e

b) Nenhuma variável da substituição ocorre dentro dos termos associados a qualquer variável.

Exemplos:

{ x/A, y/F(B), z/w } é uma substituição

{ x/G(y), y/F(x) } não é uma substituição

Exemplos de unificação e substituições respectivas:

fórmulas ou termos a unificar		emparelhamento	substituição mais geral
A	A	sim	{ }
A	B	não	
P(A)	P(A)	sim	{ }
P(A)	Q(A)	não	
P(A)	P(B)	não	
P(A)	P(F(B))	não	
P(F(B))	P(F(B))	sim	{ }
P(F(A))	P(F(B))	não	
x	A	sim	{ x/A }
B	y	sim	{ y/B }
x	P(A)	sim	{ x/P(A) }
x	P(y)	sim	{ x/P(y) }
P(F(B))	z	sim	{ z/P(F(B)) }
x	y	sim	{ x/y }
P(x)	P(y)	sim	{ x/y }
P(F(G(z)))	P(F(y))	sim	{ y/G(z) }
P(F(G(H(J(z)))))	P(F(G(H(A))))	não	
P(x,x,y,v)	P(A,A,F(B),v)	sim	{ x/A, y/F(B) }
P(A,y,C)	P(x,B,z)	sim	{ x/A, y/B, z/C }
P(x,y,C)	P(y,B,z)	sim	{ x/B, y/B, z/C }
Data(d1,m,1983)	Data(d2,Maio,y)	sim	{ d1/d2, m/Maio, y/1983 }
Ponto(x,y)	Ponto(1,3)	sim	{ x/1, y/3 }
Ponto(x,y)	Ponto(x,y,z)	não	

Note-se que as substituições resultantes da unificação com sucesso são as substituições mais gerais possível (*i.e.*, as menos restritivas). Por exemplo:

no exemplo de unificação de P(x,x,y,v) com P(A,A,F(B),v), { x/A, y/F(B) } é a substituição mais geral, pois haveria outras (menos gerais) como, por exemplo, { x/A, y/F(B), z/w }.

no exemplo de unificação de Data(d1,m,1983) com Data(d2,Maio,y), { d1/d2, m/Maio, y/1983 } é a substituição mais geral, pois haveria outras (menos gerais) como, por exemplo, { d1/3, d2/3, m/Maio, y/1983 }, ou { d1/terceiro, d2/terceiro, m/Maio, y/1983 } (*etc.*).

<u>Outros exemplos</u> de unificação e substituições respectivas:

fórmulas ou termos a unificar		emparelhamento	substituição mais geral
Cor(Tweety,Amarelo)	Cor(x,y)	sim	{ x/Tweety, y/Amarelo }
Cor(Tweety,Amarelo)	Cor(x,x)	não	
Cor(Chapeu(Carteiro),Azul)	Cor(Chapeu(y),x)	sim	{ y/Carteiro, x/Azul }
R(F(x),B)	R(y,z)	sim	{ y/F(x), z/B }
R(F(y),x)	R(x,F(B))	sim	{ x/F(B), y/B }
R(F(y),y,x)	R(x,F(A),F(v))	sim	{ x/F(F(A)), y/F(A), v/F(A) }
Ama(x,y)	Ama(y,x)	sim	{ x/y }

Regras de inferência de dedução natural em cálculo de predicados:

	Regras de Introdução	Regras de Eliminação	
I.C. Introdução da Conjunção	$\dfrac{\begin{array}{l}P\\Q\end{array}}{P \land Q}$	$\dfrac{P \land Q}{\begin{array}{l}P\\Q\end{array}}$	E.C. Eliminação da Conjunção
I.D. Introdução da Disjunção	$\dfrac{P}{P \lor Q}$	$\dfrac{\begin{array}{l}P \lor Q\\\neg P\end{array}}{Q}$	E.D. Eliminação da Disjunção
I.D.N. Introdução da Dupla Negação	$\dfrac{P}{\neg\neg P}$	$\dfrac{\neg\neg P}{P}$	E.D.N. Eliminação da Dupla Negação
		$\dfrac{\begin{array}{l}P \Rightarrow Q\\P\end{array}}{Q}$	M.P. Modus Ponens
		$\dfrac{\begin{array}{l}P \Rightarrow Q\\\neg Q\end{array}}{\neg P}$	M.T. Modus Tolens
		$\dfrac{\forall x\ P(x)}{P(\tau)}$	I.U. Instanciação Universal
G.E. Generalização Existencial	$\dfrac{P(A)}{\exists x\ P(x)}$	$\dfrac{\exists x\ P(x)}{P(\tau(v_1, ..., v_n))}$	I.E. Instanciação Existencial

Notas: Na regra para I.U., τ é uma constante, ou uma função (termo funcional), que denota um objecto conhecido do domínio de discurso. Na regra para I.E., $\tau(v_1, ..., v_n)$ é uma função (função de Skolem) das n variáveis livres ($v_1, ..., v_n$) que existam no âmbito do quantificador existencial (se não existir variável livre alguma nesse âmbito essa será uma função de 0 variáveis, ou constante) completamente nova e arbitrária. Estas regras são uma versão simplificada não completamente correta porque P pode ter mais variáveis, incluindo outras instâncias da mesma variável x.

Exemplos elementares de dedução natural em cálculo de predicados (apenas regras de inferência G.E., I.U. e I.E.):

Exemplo elementar 1:

O Pedro (do universo de discurso dos humanos) é alto.

1	`Alto(Pedro)`	

Concluir que existe alguém (no universo de discurso dos humanos) que é alto.

Prova:

2	**∃x Alto(x)**	1,G.E.

Regra de inferência usada: *Generalização Existencial* (em 2)

Nota: Um quantificador existencial é introduzido.

Exemplo elementar 2:

Todos (do universo de discurso dos humanos) são altos.

1	`∀x Alto(x)`	

Concluir que o Pedro (do universo de discurso dos humanos) é alto.

Prova:

2	**Alto(Pedro)**	1,I.U.

Regra de inferência usada: *Instanciação Universal* (em 2)

Nota: Um quantificador universal é eliminado.

Exemplo elementar 3:

Para qualquer ser humano existe um outro que o deu à luz.

1	`∀x Humano(x)⇒(∃y DeuÀluz(y,x))`	

Concluir que para cada humano existe um outro humano específico (a mãe dele) que o deu à luz.

Prova:

2	**∀x Humano(x)⇒DeuÀluz(Mãe(x),x)**	1,I.E.

Regra de inferência usada: *Instanciação Existencial* (em 2)

Nota: Um quantificador existencial é eliminado. Empregou-se uma função de Skolem, `Mãe`, tal que, dado um humano `a`, `Mãe(a)` denota a mãe dele, isto é, um outro humano `b` que satisfaz a relação `DeuÀluz(b,a)`.

Exemplos de dedução natural em cálculo de predicados:

Exemplo 1: Dado o seguinte conjunto de premissas:

```
1  ∀x Humano(x)⇒Mortal(x)
2  Humano(Confúcio)
```

Deduzir `Mortal(Confúcio)`, empregando dedução natural e justificando cada passo de prova com a regra e os números das cláusulas usadas.

Resposta:

3	`Humano(Confúcio)⇒Mortal(Confúcio)`	1,I.U.
4	**`Mortal(Confúcio)`**	2,3,M.P.

Exemplo 2: Dado o seguinte conjunto de premissas:

```
1  ∀x Produto(x)∧(BomDesign(x)∨BemPublicitado(x))⇒Vende(x)
2  ∀x Produto(x)∧Util(x)∧(BomDesign(x)∨BemPublicitado(x))⇒
   VendeMuito(x)
-----------------------------------------------------------------
3  Produto(MataMoscas)∧BomDesign(MataMoscas)∧¬VendeMuito(MataMoscas)
```

Deduzir, empregando dedução natural e justificando cada passo de prova com a regra e os números das cláusulas usadas: a) `Vende(MataMoscas)`

b) `¬Util(MataMoscas)`

Resposta:

4	`Produto(MataMoscas)∧(BomDesign(MataMoscas)∨ BemPublicitado(MataMoscas))⇒Vende(MataMoscas)`	1,I.U.
5	`Produto(MataMoscas)`	3,E.C.(2)
6	`BomDesign(MataMoscas)`	3,E.C.(2)
7	`BomDesign(MataMoscas)∨BemPublicitado(MataMoscas)`	6,I.D.
8	`Produto(MataMoscas)∧(BomDesign(MataMoscas)∨ BemPublicitado(MataMoscas))`	5,7,I.C.
9	**`Vende(MataMoscas)`**	4,8,M.P.
10	`Produto(MataMoscas)∧Util(MataMoscas)∧ (BomDesign(MataMoscas)∨BemPublicitado(MataMoscas))⇒ VendeMuito(MataMoscas)`	2,I.U.
11	`¬VendeMuito(MataMoscas)`	3,E.C.(2)
12	`¬Produto(MataMoscas)∨¬Util(MataMoscas)∨ ¬(BomDesign(MataMoscas)∨BemPublicitado(MataMoscas))`	10,11,M.T.
13	`¬Util(MataMoscas)∨¬(BomDesign(MataMoscas)∨ BemPublicitado(MataMoscas))`	5,12,E.D.
14	**`¬Util(MataMoscas)`**	7,13,E.D.

Exemplo 3: Dado o seguinte conjunto de premissas:

1	∀x Sol(x)∨(¬Chuva(x)∧¬Vento(x))⇒BomTempo(x)
2	∀x ∀y ∀z ∀u Viagem(x,y,z,u)∧BoaMoto(z)∧BomTempo(x)∧ BomTempo(y)⇒BoaViagem(x,y,z,u)
3	∀x ∀y ∀z ∀u ¬BoaViagem(x,y,z,u)∨¬Cuidadoso(u)⇒EmPerigo(u)
4	Viagem(Lisboa,Coimbra,V1000,Pedro)∧BoaMoto(V1000)
5	Viagem(Lisboa,Coimbra,T650,João)∧¬Cuidadoso(João)
6	¬Sol(Lisboa)∧¬Chuva(Lisboa)∧¬Vento(Lisboa)∧Sol(Coimbra)

Deduzir, empregando dedução natural e justificando cada passo de prova com a regra e os números das cláusulas usadas: a) BomTempo(Lisboa)

b) BoaViagem(Lisboa,Coimbra,V1000,Pedro) c) EmPerigo(João)

Resposta:

7	Sol(Lisboa)∨(¬Chuva(Lisboa)∧¬Vento(Lisboa))⇒ BomTempo(Lisboa)	1,I.U.
8	¬Chuva(Lisboa)∧¬Vento(Lisboa)	6,E.C.(2)
9	Sol(Lisboa)∨(¬Chuva(Lisboa)∧¬Vento(Lisboa))	8,I.D.
10	**BomTempo(Lisboa)**	7,9,M.P.
11	Viagem(Lisboa,Coimbra,V1000,Pedro)∧BoaMoto(V1000)∧ BomTempo(Lisboa)∧BomTempo(Coimbra)⇒ BoaViagem(Lisboa,Coimbra,V1000,Pedro)	2,I.U.
12	Viagem(Lisboa,Coimbra,V1000,Pedro)∧BoaMoto(V1000)∧ BomTempo(Lisboa)	4,10,I.C.
13	Sol(Coimbra)∨(¬Chuva(Coimbra)∧¬Vento(Coimbra))⇒ BomTempo(Coimbra)	1,I.U.
14	Sol(Coimbra)	6,E.C.(3)
15	Sol(Coimbra)∨(¬Chuva(Coimbra)∧¬Vento(Coimbra))	14,I.D.
16	BomTempo(Coimbra)	13,15,M.P.
17	Viagem(Lisboa,Coimbra,V1000,Pedro)∧BoaMoto(V1000)∧ BomTempo(Lisboa)∧BomTempo(Coimbra)	12,16,I.C.
18	**BoaViagem(Lisboa,Coimbra,V1000,Pedro)**	11,17,M.P.
19	¬BoaViagem(Lisboa,Coimbra,T650,João)∨ ¬Cuidadoso(João)⇒EmPerigo(João)	3,I.U.
20	¬Cuidadoso(João)	5,E.C.
21	¬BoaViagem(Lisboa,Coimbra,T650,João)∨ ¬Cuidadoso(João)	20,I.D.
22	**EmPerigo(João)**	19,21,M.P.

<u>Exemplo 4</u>: Dado o seguinte conjunto de premissas:

1	∀x Marciano(x)∨Terrestre(x)⇒SerVivo(x)
2	∀x Humano(x)∨Cão(x)∨Macaco(x)⇒Terrestre(x)
3	∀x Humano(x)∨Cão(x)∨Macaco(x)⇒Mamífero(x)
4	∀x Homem(x)∨Mulher(x)⇒Humano(x)
5	∀x Quadrúpede(x)∧Castanho(x)⇒Cão(x)
6	∀x Bípede(x)∧¬Verde(x)∧¬Castanho(x)⇒Humano(x)
7	∀x Bípede(x)∧Verde(x)⇒Marciano(x)
8	∀x Bípede(x)∧Castanho(x)⇒Macaco(x)
9	¬Terrestre(Zlot)∧Mulher(Ana)∧Bípede(Aardvark)∧Verde(Aardvark)

Deduza, empregando dedução natural e justificando cada passo de prova com a regra e os números das cláusulas usadas: a) ¬Mulher(Zlot)

b) SerVivo(Ana)∧Mamífero(Ana) c) SerVivo(Aardvark)

Resposta:

a) Para ¬Mulher(Zlot):

10	Humano(Zlot)∨Cão(Zlot)∨Macaco(Zlot)⇒Terrestre(Zlot)	2,I.U.
11	¬Terrestre(Zlot)	9,E.C.(3)
12	¬Humano(Zlot)∧¬Cão(Zlot)∧¬Macaco(Zlot)	10,11,M.T.
13	Homem(Zlot)∨Mulher(Zlot)⇒Humano(Zlot)	4,I.U.
14	¬Humano(Zlot)	12,E.C.(2)
15	¬Homem(Zlot)∧¬Mulher(Zlot)	13,14,M.T.
16	**¬Mulher(Zlot)**	15,E.C.

b) Para SerVivo(Ana)∧Mamífero(Ana):

10	Homem(Ana)∨Mulher(Ana)⇒Humano(Ana)	4,I.U.
11	Mulher(Ana)	9,E.C.(3)
12	Homem(Ana)∨Mulher(Ana)	11,I.D.
13	Humano(Ana)	10,12,M.P.
14	Humano(Ana)∨Cão(Ana)∨Macaco(Ana)⇒Terrestre(Ana)	2,I.U.
15	Humano(Ana)∨Cão(Ana)∨Macaco(Ana)	13,I.D.(2)
16	Terrestre(Ana)	14,15,M.P.
17	Marciano(Ana)∨Terrestre(Ana)⇒SerVivo(Ana)	1,I.U.
18	Marciano(Ana)∨Terrestre(Ana)	16,I.D.
19	SerVivo(Ana)	17,18,M.P.
20	Humano(Ana)∨Cão(Ana)∨Macaco(Ana)⇒Mamífero(Ana)	3,I.U.
21	Mamífero(Ana)	15,20,M.P.
22	**SerVivo(Ana)∧Mamífero(Ana)**	19,21,I.C.

b) Para `SerVivo(Aardvark)`:

10	Bípede(Aardvark)∧Verde(Aardvark)⇒Marciano(Aardvark)	7,I.U.
11	Bípede(Aardvark)∧Verde(Aardvark)	9,E.C.(2)
12	Marciano(Aardvark)	10,11,M.P.
13	Marciano(Aardvark)∨Terrestre(Aardvark)⇒SerVivo(Aardvark)	1,I.U.
14	Marciano(Aardvark)∨Terrestre(Aardvark)	12,I.D.
15	**SerVivo(Aardvark)**	13,14,M.P.

Exemplo 5: Dado o seguinte conjunto de premissas:

1	∀x Comprimento(x,Medio)∧Largura(x,Medio)⇒Automovel(x)
2	∀x Comprimento(x,Pequeno)∧Largura(x,MuitoPequeno)⇒ Motociclo(x)∨Bicicleta(x)
3	∀x ¬Bicicleta(x)
4	Comprimento(Obj1,Grande)
5	Largura(Obj1,Medio)
6	Comprimento(Obj2,Pequeno)
7	Largura(Obj2,MuitoPequeno)

Deduza, empregando dedução natural e justificando cada passo de prova com a regra e os números das cláusulas usadas que o objecto `Obj2` é um motociclo, *i.e.*, que `Motociclo(Obj2)`.

Resposta:

8	∀x Comprimento(Obj2,Pequeno)∧Largura(Obj2,MuitoPequeno)⇒ Motociclo(Obj2)∨Bicicleta(Obj2)	2,I.U.
9	Comprimento(Obj2,Pequeno)∧Largura(Obj2,MuitoPequeno)	6,7,I.C.
10	Motociclo(Obj2)∨Bicicleta(Obj2)	8,9,M.P.
11	¬Bicicleta(Obj2)	3,I.U.
12	**Motociclo(Obj2)**	10,11,E.D.

10- Cálculo de Predicados – O Princípio da Resolução

Os sistemas de dedução baseados em resolução usam uma única regra de inferência, denominada regra da resolução, ou princípio da resolução. Esta regra tem a vantagem de ser mecanizável na forma de um algoritmo eficiente, tendo sido aplicada em linguagens de programação baseadas em lógica como o Prolog. Como desvantagem tem, no entanto, a exigência de as fórmulas lógicas terem de estar na forma clausal.

Na descrição que se segue do princípio da resolução, as fórmulas lógicas tomadas como premissas são expressas na forma clausal, *i.e.*, cada fórmula lógica (contendo, muito possivelmente, variáveis, pois é agora de cálculo de predicados que se trata) é expressa na forma de uma cláusula. A regra da resolução pode ser descrita da seguinte forma:

Se existe uma substituição ϕ que unifica os literais A_h e B_k de modo que $\phi A_h = \neg\phi B_k$, é:

$$\frac{\begin{array}{l}\{ A_1, ..., A_{h-1}, A_h, A_{h+1}, ..., A_n \} \\ \{ B_1, ..., B_{k-1}, B_k, B_{k+1}, ..., B_m \}\end{array}}{\{ \phi A_1, ..., \phi A_{h-1}, \phi A_{h+1}, ..., \phi A_n, \phi B_1, ..., \phi B_{k-1}, \phi B_{k+1}, ..., \phi B_m \}}$$

A cláusula de conclusão $\{ \phi A_1, ..., \phi A_{h-1}, \phi A_{h+1}, ..., \phi A_n, \phi B_1, ..., \phi B_{k-1}, \phi B_{k+1}, ..., \phi B_m \}$ é, então, apelidada resolvente das cláusulas $\{ A_1, ..., A_{h-1}, A_h, A_{h+1}, ..., A_n \}$ e $\{ B_1, ..., B_{k-1}, B_k, B_{k+1}, ..., B_m \}$.

Exemplos elementares de aplicação da regra da resolução:

Exemplo elementar 1:

1	`{ ¬P(x,B,z), Q(x,y,z) }`	
2	`{ P(A,y,z) }`	
3	**`{ Q(A,B,z) }`**	de 1 e 2

Nota: $\phi = \{ x/A, y/B \}$

Exemplo elementar 2:

1	`{ ¬P(x,B,z), Q(x) }`	
2	`{ P(A,y,z), Q(B) }`	
3	**`{ Q(A), Q(B) }`**	de 1 e 2

Nota: $\phi = \{ x/A, y/B \}$

Para provar que uma dada fórmula lógica P é uma consequência lógica de um conjunto de fórmulas lógicas tomadas como premissas dado opera-se de maneira diferente do que para os sistemas de dedução natural, de acordo com o seguinte:

1- Todas as fórmulas lógicas têm de ser convertidas para a forma normal conjuntiva e representadas na forma clausal.
2- A prova é uma prova por refutação: em vez de derivar P directamente do conjunto de fórmulas lógicas tomadas como premissas, prova-se que ¬P é inconsistente com esse conjunto de fórmulas (*i.e.*, que ¬P em conjunção com esse conjunto de fórmulas é insatisfiável).

O processo geral de prova manual de uma fórmula P quando se emprega a regra da resolução é, então:

1- Converter o conjunto de fórmulas lógicas tomadas como premissas para a forma clausal.
2- Negar P.
3- Converter ¬P para a forma clausal.
4- Juntar as cláusulas de ¬P ao conjunto de fórmulas tomadas como premissas.
5- Empregando a regra da resolução, tentar derivar a cláusula vazia (*i.e.*, a cláusula { }).

Se o passo 5- tem sucesso, então P é uma consequência lógica do conjunto de fórmulas tomadas como premissas, caso contrário, não o é.

Conversão para a forma clausal – A conversão de uma fórmula lógica para a correspondente forma clausal pode ser realizada por um processo sistemático em seis fases:

1- Remover equivalências e implicações.
2- Mover negações para dentro, de modo a que afectem apenas as fórmulas atómicas.
3- Renomear cada variável quantificada, de modo a que cada quantificador quantifique uma única variável. Remover quantificadores existenciais ou skolemização: cada variável quantificada existencialmente é substituída por uma função de Skolem única, de tantas variáveis quantas as que, sendo quantificadas universalmente, têm um âmbito que abrange o do quantificador existencial respectivo; cada variável quantificada existencialmente fora do âmbito de qualquer outra variável quantificada universalmente é substituída por uma constante de Skolem única, *i.e.*, uma função de Skolem de zero variáveis.
4- Remover quantificadores universais (todas as variáveis existentes passam a estar quantificadas universalmente de forma implícita).
5- Passagem à forma normal conjuntiva, de modo a obter conjunções de disjunções de literais.
6- Colocar em forma de cláusulas (cada cláusula é um conjunto de literais em disjunção implícita que corresponde a um dos disjunctos da fase anterior). As variáveis que existam em cada cláusula deverão ter nomes diferentes das das outras cláusulas.

Exemplos de conversão para a forma clausal (explicita-se cada uma das fases de conversão):

Exemplo 1: Converter para forma clausal a fórmula lógica `(¬ ∀x P(x))⇒(∃x P(x))`.

Resposta:

1	`¬(¬ ∀x P(x))∨(∃x P(x))`
2	`(∀x P(x))∨(∃x P(x))`
3	`(∀x P(x))∨P(Sk)`
4	`P(x)∨P(Sk)`
5	`P(x)∨P(Sk)`
6	**`{ P(x), P(Sk) }`**

Nota: Sk é uma função de Skolem de 0 argumentos (*i.e.*, uma constante de Skolem).

Exemplo 2: Converter para forma clausal a seguinte fórmula lógica:

$\forall x\ P(x) \wedge Q(x) \Rightarrow \neg(\forall y\ R(x,y) \Rightarrow S(x))$

Resposta:

1	$\forall x\ \neg(P(x) \wedge Q(x)) \vee \neg(\forall y\ \neg R(x,y) \vee S(x))$
2	$\forall x\ \neg P(x) \vee \neg Q(x) \vee (\exists y\ R(x,y) \wedge \neg S(x))$
3	$\forall x\ \neg P(x) \vee \neg Q(x) \vee (R(x,Sk(x)) \wedge \neg S(x))$
4	$\neg P(x) \vee \neg Q(x) \vee (R(x,Sk(x)) \wedge \neg S(x))$
5	$(\neg P(x) \vee \neg Q(x) \vee R(x,Sk(x))) \wedge (\neg P(x) \vee \neg Q(x) \vee \neg S(x))$
6	**{ ¬P(x1), ¬Q(x1), R(x1,Sk(x1)) }** **{ ¬P(x2), ¬Q(x2), ¬S(x2) }**

Nota: Sk é uma função de Skolem de 1 argumento.

Exemplo 3: Converter para forma clausal a seguinte fórmula lógica:

$\forall x\ (\forall y\ P(x,y)) \Rightarrow \neg(\forall y\ Q(x,y) \Rightarrow R(x,y))$

Resposta:

1	$\forall x\ \neg(\forall y\ P(x,y)) \vee \neg(\forall y\ \neg Q(x,y) \vee R(x,y))$
2	$\forall x\ (\exists y\ \neg P(x,y)) \vee (\exists y\ Q(x,y) \wedge \neg R(x,y))$
3	$\forall x\ (\exists y\ \neg P(x,y)) \vee (\exists z\ Q(x,z) \wedge \neg R(x,z))$ $\forall x\ \neg P(x,Sk1(x)) \vee (Q(x,Sk2(x)) \wedge \neg R(x,Sk2(x)))$
4	$\neg P(x,Sk1(x)) \vee (Q(x,Sk2(x)) \wedge \neg R(x,Sk2(x)))$
5	$(\neg P(x,Sk1(x)) \vee Q(x,Sk2(x))) \wedge (\neg P(x,Sk1(x)) \vee \neg R(x,Sk2(x)))$
6	**{ ¬P(x1,Sk1(x1)), Q(x1,Sk2(x1))) }** **{ ¬P(x2,Sk1(x2)), ¬R(x2,Sk2(x2)) }**

Nota: Sk1 e Sk2 são ambas funções de Skolem de 1 argumento.

Exemplos de dedução empregando resolução:

Exemplo 1: Considere o seguinte conjunto de cláusulas composto pelas cláusulas 1 a 8 juntamente com as cláusulas adicionais 9 e 10 (todas em forma clausal):

1	{ ¬F(x1), ¬C(x1), G(x1), H(x1) }
2	{ ¬D(x2), A(x2) }
3	{ ¬E(x3), A(x3) }
4	{ ¬A(x4), B(x4), C(x4) }
5	{ ¬I(x5), ¬G(x5) }
6	{ ¬J(x6), ¬G(x6) }
7	{ ¬F(x7), C(x7), K(x7) }
8	{ ¬F(x8), G(x8), K(x8) }
9	{ ¬B(x9) }
10	{ ¬C(Var33) }

Empregando a regra da resolução e justificando cada passo de prova com os números das cláusulas usadas, prove por refutação que:

a) `¬D(Var33)` b) `¬E(Var33)`

Resposta:

a) Para `¬D(Var33)`:

11	{ D(Var33) }	

12	{ ¬A(Var33), B(Var33) }	de 4 e 10
13	{ ¬A(Var33) }	de 9 e 12
14	{ ¬D(Var33) }	de 2 e 13
15	**{ }**	de 11 e 14

b) Para `¬E(Var33)`:

11	{ E(Var33) }	

12	{ ¬A(Var33), B(Var33) }	de 4 e 10
13	{ ¬A(Var33) }	de 9 e 12
14	{ ¬E(Var33) }	de 3 e 13
15	**{ }**	de 11 e 14

Exemplo 2: Considere o seguinte conjunto de cláusulas composto pelas cláusulas 1 a 8 juntamente com as cláusulas adicionais 9 a 12 (todas em forma clausal):

1	{ ¬F(x1), ¬C(x1), G(x1), H(x1) }
2	{ ¬D(x2), A(x2) }
3	{ ¬E(x3), A(x3) }
4	{ ¬A(x4), B(x4), C(x4) }
5	{ ¬I(x5), ¬G(x5) }
6	{ ¬J(x6), ¬G(x6) }
7	{ ¬F(x7), C(x7), K(x7) }
8	{ ¬F(x8), G(x8), K(x8) }
9	{ ¬B(x9) }
10	{ E(R1) }
11	{ I(R1) }
12	{ F(R1) }

Empregando a regra da resolução e justificando cada passo de prova com os números das cláusulas usadas, prove por refutação que:

a) `C(R1)` b) `K(R1)` c) `H(R1)`

Resposta:

a) Para `C(R1)`:

13	{ ¬C(R1) }	

14	{ A(R1) }	de 3 e 10
15	{ B(R1), C(R1) }	de 4 e 14
16	{ C(R1) }	de 9 e 15
17	**{ }**	de 13 e 16

b) Para `K(R1)`:

13	{ ¬K(R1) }	

14	{ ¬G(R1) }	de 5 e 11
15	{ ¬F(R1), K(R1) }	de 8 e 14
16	{ K(R1) }	de 12 e 15
17	**{ }**	de 13 e 16

c) Para `H(R1)`:

13	{ ¬H(R1) }	

14	{ ¬C(R1), G(R1), H(R1) }	de 1 e 12
15	{ A(R1) }	de 3 e 10
16	{ B(R1), C(R1) }	de 4 e 15
17	{ C(R1) }	de 9 e 16
18	{ G(R1), H(R1) }	de 14 e 17
19	{ ¬G(R1) }	de 5 e 11
20	{ H(R1) }	de 18 e 19
21	**{ }**	de 13 e 20

Exemplo 3: Considere o seguinte conjunto de cláusulas composto pelas cláusulas 1 a 12 (todas em forma clausal):

```
1  { ¬I(x1), F(x1) }
2  { H(x2), G(x2), F(x2) }
3  { ¬E(x3,y3,z3,u3), ¬D(z3), ¬F(x3), ¬F(y3), C(x3,y3,z3,u3) }
4  { B(x4), A(y4) }
--------------------------------------------------------------
5  { E(L1,L2,M1,P1) }
6  { D(M1) }
7  { B(P1) }
8  { ¬I(L1) }
9  { ¬H(L1) }
10 { ¬G(L1) }
11 { I(L2) }
12 { ¬B(P2) }
```

Empregando a regra da resolução e justificando cada passo de prova com os números das cláusulas usadas, prove por refutação que:

a) `F(L1)` b) `C(L1,L2,M1,P1)`

Resposta:

a) Para `F(L1)`:

```
13 { ¬F(L1) }
```

14	{ G(L1), F(L1) }	de 2 e 9
15	{ F(L1) }	de 10 e 14
16	**{ }**	de 13 e 15

b) Para `C(L1,L2,M1,P1)`:

```
13 { ¬C(L1,L2,M1,P1) }
```

14	{ ¬D(M1), ¬F(L1), ¬F(L2), C(L1,L2,M1,P1) }	de 3 e 5
15	{ G(L1), F(L1) }	de 2 e 9
16	{ F(L1) }	de 10 e 15
17	{ ¬D(M1), ¬F(L2), C(L1,L2,M1,P1) }	de 14 e 16
18	{ F(L2) }	de 1 e 11
19	{ ¬D(M1), C(L1,L2,M1,P1) }	de 17 e 18
20	{ C(L1,L2,M1,P1) }	de 6 e 19
21	**{ }**	de 13 e 20

Exemplo 4: Considere o seguinte conjunto de cláusulas composto pelas cláusulas 1 a 18 (todas em forma clausal):

1	`{ ¬M(x1), L(x1) }`
2	`{ ¬K(x2), L(x2) }`
3	`{ ¬J(x3), K(x3) }`
4	`{ ¬I(x4), K(x4) }`
5	`{ ¬H(x5), K(x5) }`
6	`{ ¬J(x6), G(x6) }`
7	`{ ¬I(x7), G(x7) }`
8	`{ ¬H(x8), G(x8) }`
9	`{ ¬F(x9), J(x9) }`
10	`{ ¬E(x10), J(x10) }`
11	`{ ¬D(x11), ¬C(x11), I(x11) }`
12	`{ ¬B(x12), A(x12), C(x12), J(x12) }`
13	`{ ¬B(x13), ¬A(x13), M(x13) }`
14	`{ ¬B(x14), ¬C(x14), H(x14) }`
15	`{ ¬K(P1) }`
16	`{ E(P2) }`
17	`{ B(P3) }`
18	`{ A(P3) }`

Empregando a regra da resolução e justificando cada passo de prova com os números das cláusulas usadas, prove por refutação que:

a) `¬E(P1)` b) `L(P2)∧G(P2)` c) `L(P3)`

Resposta:

a) Para `¬E(P1)`:

19	`{ E(P1) }`	

20	`{ ¬J(P1) }`	de 3 e 15
21	`{ ¬E(P1) }`	de 10 e 20
22	**`{ }`**	de 19 e 21

b) Para `L(P2)∧G(P2)`:

19	`{ ¬L(P2), ¬G(P2) }`	

20	`{ J(P2) }`	de 10 e 16
21	`{ G(P2) }`	de 6 e 20
22	`{ ¬L(P2) }`	de 19 e 21
23	`{ K(P2) }`	de 3 e 20
24	`{ L(P2) }`	de 2 e 23
25	**`{ }`**	de 22 e 24

c) Para `L(P3)`:

19	{ ¬L(P3) }	

20	{ ¬A(P3), M(P3) }	de 13 e 17
21	{ M(P3) }	de 18 e 20
22	{ L(P3) }	de 1 e 21
23	**{ }**	de 19 e 22

Exemplo 5: Dado o seguinte conjunto de premissas:

1	{ ¬Comprimento(x,Medio), ¬Largura(x,Medio), Automóvel(x) }
2	{ ¬Comprimento(x,Pequeno), ¬Largura(x,MuitoPequeno), Motociclo(x), Bicicleta(x) }
3	{ ¬Bicicleta(x) }
4	{ Comprimento(Obj1,Grande) }
5	{ Largura(Obj1,Medio) }
6	{ Comprimento(Obj2,Pequeno) }
7	{ Largura(Obj2,MuitoPequeno) }

Empregando a regra da resolução e justificando cada passo de prova com os números das cláusulas usadas, prove por refutação que `Motociclo(Obj2)`.

Resposta:

8	{ ¬Motociclo(Obj2) }	

9	{ ¬Largura(Obj2,MuitoPequeno), Motociclo(Obj2), Bicicleta(Obj2) }	de 2 e6
10	{ Motociclo(Obj2), Bicicleta(Obj2) }	de 7 e 9
11	{ Motociclo(Obj2) }	de 3 e 10
12	**{ }**	de 8 e 11

Bibliografia

Symbolic Logic and Mechanical Theorem Proving

Chin-Liang **Chang**, Richard Char-Tung **Lee**

Academic Press, 1973.

Artificial Intelligence, A Modern Approach

Stuart **Russell**, Peter **Norvig**

Prentice Hall, 1995.

Logical Foundations of Artificial Intelligence

Michael **Genesereth**, Nils **Nilsson**

Morgan Kaufman Publishers, Inc., 1987.