Andréa Loparic

LIÇÕES DE LÓGICA

1996

I. Gramática de $\mathcal{L}$

1. Vocabulário de $\mathcal{L}$

O vocabulário de $\mathcal{L}$ é formado pelas seguintes categorias de símbolos:

a) variáveis;
b) símbolos interpretáveis
- nominais,
- verbais;

c) símbolos lógicos
- predicado de igualdade,
- conectivos,
- símbolos de quantificação;

d) símbolos auxiliares de pontuação.

a) As variáveis são letras minúsculas entre 'u' e 'z', com ou sem índices inferiores (subscritos).
Exemplos: x, y, z_1, x_2, w_7, x_{123}, etc.

b) Os símbolos interpretáveis nominais são minúsculas de 'a' a 't', com ou sem índices inferiores (subscritos) e com ou sem índices superiores (sobrescritos).
- Os símbolos interpretáveis nominais sem sobrescritos são ditos saturados e são chamadas *constantes individuais*.
Exemplos: a, b, a_1, c_2, m, n_5, etc.
- Os símbolos interpretáveis nominais com sobrescritos são ditos insaturados; o sobrescrito indica o seu grau de insaturação ou carência. Eles são chamados *símbolos funcionais de carência n* (onde *n* é o seu sobrescrito) ou *operadores de carência n.*
Exemplos: f^1, $g_2{}^1$, $h_3{}^1$, são operadores de carência 1;
f^2, g^2, $h_1{}^2$, $h_2{}^2$, operadores de carência 2;
f^3, g^3, etc., operadores de carência 3;
etc.

c) Os símbolos interpretáveis verbais são maiúsculas com ou sem subscrito e com ou sem sobrescrito.
- Os símbolos interpretáveis verbais sem sobrescritos são ditos saturados e são chamados *letras sentenciais*, ou ainda *predicados de carência* 0.
Exemplos: P, Q, R_1, A, B_3, S_{15}, etc.
- Os símbolos interpretáveis verbais com sobrescrito são ditos insaturados; o sobrescrito indica o seu grau de insaturação ou sua carência. Ele são chamados *predicados de carência n*, onde *n* é seu sobrescrito.

Exemplos: F^1, G^1, $S_1{}^1$, $H_2{}^1$ são predicados de carência 1;
F^2, B^2, $H_3{}^2$, $G_{19}{}^2$são predicados de carência 2;
A^3, $F_2{}^3$, $H_1{}^3$, $G_5{}^3$, L^3, R^3 são predicados de carência 3;
etc.

Observação:

1. Os subscritos e sobrescritos que podem afetar símbolos das categorias a), b) e c) são números inteiros positivos. Eles não são considerados símbolos, mas parte de símbolos. Assim x_1 é um único símbolo, assim como a_{12}, $f_2{}^1$, $h_1{}^3$, etc.
2. Os subscritos cumprem unicamente a função de aumentar a quantidade de símbolos disponíveis em cada categoria. Assim, há uma quantidade infinita de símbolos em cada uma delas. Além disso, eles induzem uma *ordem alfabética* em cada categoria ou subconjunto:
   ordem alfabética das variáveis: $u, v, x, \ldots, z, u_1, v_1, \ldots, z_1, u_2, \ldots$
   ordem alfabética das constantes individuais: $a, b, \ldots, t, a_1, b_1, \ldots, t_1, a_2, \ldots$
   ordem alfabética das letras sentenciais: $A, B, \ldots, Z, A_1, B_1, \ldots, Z_1, A_2, \ldots$
   ordem alfabética dos operadores de carência 1: $a^1, b^1, \ldots, t^1, a_1{}^1, b_1{}^1, \ldots, t_1{}^1, a_2{}^1, \ldots$
   ordem alfabética dos predicados de carência 1: $A^1, B^1, \ldots, Z^1, A_1{}^1, B_1{}^1, \ldots, Z_1{}^1, A_2{}^1 \ldots$
   ordem alfabética dos operadores de carência 2: $a^2, b^2, \ldots, t^2, a_1{}^2, b_1{}^2, \ldots, t_1{}^2, a_2{}^2, \ldots$
   etc.
3. As variáveis e as constantes individuais formam um grupo de símbolos chamados *símbolos individuais*.

d) Os símbolos lógicos são 8:

- o predicado de igualdade = ; também chamado de símbolo lógico da identidade;
- 5 conectivos, a saber:
  o conectivo unário da negação: $\neg$,
  e 4 conectivos binários:
  conjunção: $\wedge$,
  disjunção: $\vee$,
  condicional: $\rightarrow$,
  bicondicional: $\leftrightarrow$;
- 2 símbolos de quantifição, a saber:
  universal: $\forall$,
  e existencial: $\exists$.

Observação:

As notações usadas para alguns símbolos lógicos não são unificadas na literatura. Assim, entre outros, encontramos na literatura as notações:

conectivos	notações também usadas na literatura
negação	$-$, $\sim$
conjunção	. , &
condicional	$\supset$
bicondicional	$\equiv$

e) Os símbolos de pontuação são os dois parênteses: ( )

2. Gramática de $\mathcal{L}$

2.1) Expressões de $\mathcal{L}$.

Uma expressão do vocabulário de $\mathcal{L}$ é uma sequência finita de símbolos do vocabulário de $\mathcal{L}$. Assim, se $\sigma_1, \sigma_2, \ldots, \sigma_n$ são símbolos do vocabulário de $\mathcal{L}$, a sequência $\sigma_1, \sigma_2, \ldots, \sigma_n$ é uma *expressão* do vocabulário de $\mathcal{L}$.

Se $\sigma_1, \sigma_2, \ldots, \sigma_n$ é uma expressão e para algum i, $1 \leq i \leq n$, σ é σ_i, dizemos que σ *ocorre* em $\sigma_1, \sigma_2, \ldots, \sigma_n$. Se σ ocorre mais de uma vez numa dada expressão, as diferentes ocorrências de σ são contadas da esquerda para a direita: falamos na primeira, na segunda, na terceira ocorrência de σ na expressão, e assim por diante.

Nem todas as expressões do vocabulário de $\mathcal{L}$ são *expressões gramaticais* ou *expressões bem formadas* de $\mathcal{L}$. As expressões bem formadas de $\mathcal{L}$ compõem 2 grupos: os termos e as fórmulas.

Expressões bem formadas de $\mathcal{L}$:

a) nominais – termos,
b) verbais – fórmulas.

2.2) Termos de $\mathcal{L}$.

O conjunto dos *termos* de $\mathcal{L}$ pode ser assim definido:

a) um símbolo individual é um termo de $\mathcal{L}$,
b) se φ^n é um operador de carência *n* e $\tau_1, \ldots, \tau_n$ são termos de $\mathcal{L}$, então a expressão $\varphi^n\tau_1, \ldots, \tau_n$ é um termo de $\mathcal{L}$.

Exemplos: a, x, f^1a, g^2ax, h^1f^1a, $g^2f^1ah^1b$, x_1, y, $f_1{}^1x$, $h^3f^1ah^1g^2xyg^2ab$ são termos.

Observação:

A definição de termo é um exemplo de um tipo de definição muito comum em lógica: a *definição por indução*. Um conjunto indutivamente definido contém dois tipos de elementos : os elementos *originais* ou *básicos*; e os elementos que são gerados por meio de um número finito de aplicações de um certo número de *regras de geração*. Assim, podemos associar aos elementos gerados um número que corresponde à quantidade de aplicações de regras necessário para a sua geração; e, aos elementos originais, associamos o número 0. O número assim associado a cada elemento de um conjunto indutivamente definido é, então, dito o grau de complexidade do elemento ou geração do elemento. No caso dos termos, por exemplo:

- a, x, b_1 são exemplos de termos originais ou básicos, ou seja, termos de complexidade 0 ou de geração 0;
- f^1a, g^2ab, h^3xbc são termos de complexidade 1;
- g^2f^1ab, h^3abg^2xy são termos de complexidade2;
- $f^2g^2abh^1a$, $f^1f^1f^1b$, $h^1g^2abf^3xby$ têm complexidade 3;
- etc.

Observe-se que, no caso dos termos, há uma única regra de geração de termos complexos: a condição b) da definição de termo. Como essa regra prescreve que se tome um operador de carência *n* seguido de *n* termos para formar um novo termo, e como os termos básicos [condição a)] não contêm operadores, a complexidade de um termo pode ser medida pelo número de ocorrência de operadores nesse termo.

Os termos podem ser ainda classificados em dois grupos: termos abertos ou fechados.

Os *termos abertos* são aqueles que contêm ocorrência de variáveis (pelo menos uma ocorrência de variável); os *termos fechados* não contêm ocorrências de variáveis.

2.3) Fórmulas de $\mathcal{L}$.

A outra classe de expressões bem formadas de $\mathcal{L}$ é constituída pelas fórmulas. A definição completa de fórmula é também uma definição por indução. Primeiramente, definimos um conjunto básico de fórmulas, ditas *fórmulas atômicas*. Essas são as fórmulas de complexidade 0. Em seguida, são dadas regras de obtenção de fórmulas mais complexas a partir de fórmulas menos complexas. Nessa primeira fase de estudo da gramática, vamos nos limitar a apresentar a definição de fórmula atômica. A definição indutiva completa de fórmula será introduzida posteriormente.

Há 3 tipos de fórmula atômica, a saber:

i. uma letra sentencial é uma fórmula atômica;

ii. um predicado de carência *n* seguido de *n* termos é uma fórmula atômica;

iii. uma expressão formada pelo predicado lógico da igualdade entre dois termos é uma fórmula atômica.

Assim, se Π representa uma letra sentencial, Π^n, um predicado de carência n e $\tau_1, \tau_2, \ldots, \tau_n$, termos de $\mathcal{L}$, uma fórmula atômica tem uma das seguintes formas:

i. Π;

ii. $\Pi^n\tau_1, \ldots, \tau_n$;

iii. $\tau_1 = \tau_2$.

Como os termos, as fórmulas atômicas também se subdividem em abertas e fechadas. Fórmulas atômicas *abertas* são as que contêm ocorrência de variáveis, as *fechadas* são as que não contêm tais ocorrências. As fórmulas atômicas fechadas são chamadas *sentenças atômicas*.

Observe-se que se uma variável ocorre numa fórmula atômica, essa ocorrência se dá dentro de um termo que figura como parte dessa fórmula. Assim:

- fórmulas atômicas do tipo Π são sentenças atômicas;
- fórmulas do tipo $\Pi^n\tau_1, \ldots, \tau_n$ são sentenças se e só se $\tau_1, \ldots, \tau_n$ forem termos fechados;
- fórmulas atômicas do tipo $\tau_1 = \tau_2$ são sentenças se e só se τ_1 e τ_2 forem termos fechados.

Observe-se, ainda, que o símbolo = é um predicado lógico; assim, é um símbolo do tipo verbal e, nesse sentido, integra a classe dos símbolos verbais ou predicativos, juntamente com as letras sentenciais e predicados de carência *n*. A seguinte característica distingue, então, os termos das fórmulas atômicas: enquanto os termos são formados exclusivamente por símbolos nominais, uma fórmula atômica conterá necessariamente uma ocorrência de um símbolo verbal. Do ponto de vista gráfico, enquanto um termo é uma sequência de letras minúsculas, a fórmula atômica conterá um símbolo que não é uma letra minúscula: poderá ser uma maiúscula ou o sinal de igualdade.

II. Semântica de $\mathcal{L}$

Vamos introduzir, nesse momento, alguns conceitos que nos possibilitarão atribuir significados a algumas expressões de $\mathcal{L}$ já introduzidas sintaticamente.

1. Interpetações para $\mathcal{L}$

Uma *interpretação* I para $\mathcal{L}$ é dada por:

- um conjunto não-vazio de objetos quaisquer, U_I, dito *universo de discurso* de I;
- uma *função* $\Im$ que associa aos símbolos interpretáveis de $\mathcal{L}$ significados, de acordo com a seguinte estipulação:
  - i) a cada constante individual κ, $\Im$ associa um objeto de U_I — ou seja: $\Im(\kappa) \in U_I$;
  - ii) a cada operador de carência *n*, φ^n, $\Im$ associa uma operação n-ária em U_I — ou seja: $\Im(\varphi^n)$: $U_I^n \to U_I$;
  - iii)a cada letra sentencial, Π, $\Im$ associa um dos dois valores de verdade, o verdadeiro (V) ou o falso (F) — ou seja: $\Im(\Pi) \in \{V, F\}$;
  - iv)a cada predicado de carência *n*, Π^n, $\Im$ associa uma relação n-ária entre objetos de U_I — ou seja: $\Im(\Pi^n) \subset U_I^n$.

Como se vê, infinitas interpretações podem ser dadas para a linguagem $\mathcal{L}$ e a escolha de uma interpretação particular é arbitrária e poderá atender a interesses de conveniências nas aplicações e usos de $\mathcal{L}$.

Exemplo de uma interpretação: I

U_I: conjunto dos números naturais $\{0, 1, 2, ...\}$

$\Im(a) = 0$; $\Im(b) = 5$; $\Im(c) = 7$; as demais constantes individuais denotam o 1.

$\Im(f^1)$ = função "sucessor"; $\Im(g^1)$ = função "quadrado"; $\Im(h^1)$ = função "dobro";

$\Im(s^2)$ = função "soma"; $\Im(p^2)$ = função "produto";

$\Im(d^2)$ = função que associa a cada par $\langle m, n\rangle$: m-n, se $m \geq n$; 0, se $m < n$.

As demais funções n-árias são interpretadas pelas funções "projeção do primeiro argumento", i.e. $\Im(\varphi^n (m_1, ..., m_2)) = m_1$, para todo símbolo funcional n-ário φ^n antes que os acima relacionados.

$\Im(A) = V$; $\Im(B_1) = V$; as demais letras sentenciais são associadas ao F.

$\Im(F^1) = \{0, 2, 4, 6, \ldots\}$ (conjunto dos números naturais pares);

$\Im(G^1)$ = conjunto dos números naturais primos;

$\Im(H^1) = \{2, 71, 245, 1024, 3258, 10999\}$;

$\Im(R^2)$ = relação "menor que";

$\Im(D^2)$ = relação "divisível por";

Os demais símbolos de predicado são associados ao conjunto vazio; i.e., para cada predicado Π^n não mencionado acima, $\mathfrak{I}(\Pi^n) = \emptyset$.

Outro exemplo de interpretação: I'

$U_{I'}$ = conjunto das pessoas humanas

$\mathfrak{I}'(a)$ = Aristóteles; $\mathfrak{I}'(b)$ = Kant; $\mathfrak{I}'(n)$ = Napoleão Bonaparte; $\mathfrak{I}'(p)$ = Dom Pedro II; as demais constantes são associadas à Princesa Isabel.

$\mathfrak{I}'(f^1)$ = função que associa a cada primeiro ancestral humano, ele próprio; e a cada um dos demais, o seu pai; vamos nos permitir dizer: função "pai"; $\mathfrak{I}'(m^1)$ = função "mãe", definida analogamente à função "pai", acima.

$\mathfrak{I}'(F^1)$ = conjunto dos filósofos; $\mathfrak{I}'(B^1)$ = conjunto dos brasileiros; $\mathfrak{I}'(M^1)$ = conjunto das mulheres; $\mathfrak{I}'(R^2)$ = relação "mais velho que"; $\mathfrak{I}'(S^2)$ = relação "casado com"; $\mathfrak{I}'(C^2)$ = relação "conterrâneo de"; $\mathfrak{I}'(P^2)$ = relação "professor de".
Todas as letras sentenciais são associadas ao V; todos os símbolos funcionais não mencionados são associados à função "projeção do primeiro argumento"; todos os símbolos de predicado não mencionados são associados ao conjunto vazio.

Um terceiro exemplo de interpretação: I"

$U_{I''}$ = conjunto dos países e cidades do mundo

$\mathfrak{I}''(a)$ = Argentina; $\mathfrak{I}''(b)$ = Brasil; $\mathfrak{I}''(c)$ = Cuba; $\mathfrak{I}''(d)$ = Dinamarca; $\mathfrak{I}''(e)$ = Espanha; $\mathfrak{I}''(f)$ = França; $\mathfrak{I}''(b_1)$ = Brasília; $\mathfrak{I}''(b_2)$ = São Paulo; $\mathfrak{I}''(b_3)$ = Porto Alegre; $\mathfrak{I}''(b_4)$ = Rio de Janeiro; $\mathfrak{I}''(b_5)$ = Campinas; $\mathfrak{I}''(b_6)$ = Recife; $\mathfrak{I}''(f_1)$ = Paris; $\mathfrak{I}''(o_1)$ = Buenos Aires. As demais constantes individuais são associadas à Inglaterra.

$\mathfrak{I}''(f^1)$ = função que associa: $\begin{cases} \text{a cada cidade, ela própria, e} \\ \text{a cada país, sua capital;} \end{cases}$

$\mathfrak{I}''(g^1)$ = função que associa: $\begin{cases} \text{a cada cidade, o país a que pertence, e} \\ \text{a cada país, ele próprio;} \end{cases}$

Todos os demais símbolos funcionais são associados à função projeção do primeiro argumento.

Todas as letras sentenciais sem índices inferiores são associadas ao V; todas as letras sentenciais com índices inferiores são associadas ao F.

$\mathfrak{I}''(F^1)$ = conjunto dos países; $\mathfrak{I}''(G^1)$ = conjunto das capitais; $\mathfrak{I}''(H^1)$ = conjunto das cidades com mais de 1.000.000 de habitantes; $\mathfrak{I}''(B^1)$ = conjunto das cidades brasileiras; $\mathfrak{I}''(R^2)$ = ser "mais populoso que"; $\mathfrak{I}''(C^2)$ = ser "uma cidade de". Todos os demais símbolos de predicado são associados a $\emptyset$.

2. Denotação de um termo fechado

O conjunto dos termos fechados já foi definido anteriormente. No entanto, ele pode ser indutivamente obtido pelas condições que se seguem:

1) uma constante individual é um termo fechado;
2) se $\tau_1, \ldots, \tau_n$ são termos fechados e φ^n é um símbolo funcional de carência *n*, então $\varphi^n\tau_1\ldots\tau_n$ é um termo fechado.

Para definir a denotação de um termo fechado τ, de acordo com uma interpretação qualquer I, usaremos a estratégia sugerida pela definição acima: definiremos a denotação de um termo fechado simples, obtdio pela cláusula 1) e, em seguida, diremos como calcular a denotação de um termo $\varphi^n\tau_1\ldots\tau_n$, obtida pela cláusula 2), a partir das denotações de $\tau_1, \ldots, \tau_n$. Para abreviar a expressão "a denotação de τ segundo I", usaremos a notação: $\delta_I(\tau)$.

Definição da denotação de τ

1) se τ é uma constante individual, $\delta_I(\tau)=\Im(\tau)$;
2) se τ é $\varphi^n\tau_1\ldots\tau_n$, $\delta_I(\tau)=\Im(\varphi^n)[\delta_I(\tau_1), \ldots, \delta_I(\tau_n)]$.

Exemplos:

Seja I a primeira interpretação dada como exemplo. Então:

$\delta_I(a)=0 \quad \delta_I(b)=5 \quad \delta_I(c)=7 \quad \delta_I(d)=1$

$\delta_I(f^1a)=\Im(f^1)[\delta_I(a)]=$ o sucessor de $0 = 1$

$\delta_I(s^2bc)=\Im(s^2)[\delta_I(b), \delta_I(c)]=5+7=12$

$\delta_I(g^1s^2bc)=\Im(g^1)[\delta_I(s^2bc)]=$ o dobro de $12 = 24$

$\delta_I(p^2cg^1b)=\Im(p^2)[\delta_I(c), \delta_I(g^1b)]=7\times\delta_I(g^1b)=7\times\Im(g^1)[\delta_I(b)]=7\times$ o dobro de $5=$

$=7\times$ o dobro de $5 = 7 \times 10 = 70$

Seja, agora, I a interpretação anteriormente denotada por I". Então:

$\delta_I(a)=$ a Argentina $\quad \delta_I(b)=$ o Brasil

$\delta_I(f^1b)=\Im(f^1)[\delta_I(b)]=\Im(f^1)[\text{Brasil}]=$Brasília

Analogamente, $\delta_I(f^1a)=$Buenos Aires $\quad \delta_I(b_5)=$Campinas $\quad \delta_I(g^1b_5)=$Brasil

$\delta_I(f^1g^1b_5)=$Brasília

3. Valor de verdade das sentenças atômicas

Já vimos o que é uma sentença atômica: uma fórmula atômica que não contém variáveis. Mas as sentenças atômicas também admitem outras definições, a saber:

α é uma sentença atômica se e só se:

1) α é uma letra sentencial;
2) Π^n é um predicado de carência *n*, $\tau_1,\ldots,\tau_n$ são termos fechados e α é $\Pi^n\tau_1\ldots\tau_n$;
3) τ_1 e τ_2 são termos fechados e α é $\tau_1=\tau_2$.

Veremos, agora, como associar valores de verdade a sentenças atômicas de acordo com uma interpretação I.[1]

1) Se α é a letra sentencial Π, $\nu_I(\alpha)=\Im(\Pi)$;
2) se α é $\Pi^n\ \tau_1\ldots\tau_n$, $\nu_I(\alpha)=V$ se e só se a n-upla formada pelas denotações de $\tau_1,\ldots,\tau_n$, tomadas nessa ordem, pertence à relação que I associa a Π^n; ou seja: $\nu_I(\alpha)=V$ se e só se $\langle\delta_I(\tau_1),\ \ldots,\ \delta_I(\tau_n)\rangle \in \Im(\Pi^n)$;
3) se α é $\tau_1=\tau_2$, $\nu_I(\alpha)=V$ se e só se $\delta_I(\tau_1)=\delta_I(\tau_2)$; ou seja, se as denotações segundo I dos termos τ_1 e τ_2 são o mesmo objeto.

Exemplos:

Seja I a interpretação introduzida como I' anteriormente. Então:

$\nu_I(P)=V \quad \nu_I(B_1)=V$

$\nu_I(F^1a)=V$, pois $\delta_I(a)$=Aristóteles, $\Im(F^1)$=conjunto dos filósofos, e Aristóteles pertence ao conjunto dos filósofos.

$\nu_I(B^1c)=V$, pois $\Im(c)$=Princesa Isabel, $\Im(B^1)$=conjunto dos brasileiros, e a Princesa Isabel era brasileira.

$\nu_I(B^1n)=F$, pois Napoleão não pertence ao conjunto dos brasileiros.

$\nu_I(C^2an)=F$, pois a dupla ⟨Aristóteles, Napoleão⟩ não pertence à relação "conterrâneo de".

$\nu_I(R^2pq)=V$, pois o par ⟨Pedro II, Princesa Isabel⟩ pertence à relação "mais velho que".

$\nu_I(F^1p^1p^1q)=F$, pois $\delta_I(p^1p^1q) \notin \Im(F^1)$; ou seja, o avô paterno da Princesa Isabel não era filósofo.

$\nu_I(p^1q=p)=V$, pois os termos p^1q e p têm em I a mesma denotação; ou seja, $\delta_I(p^1q)=\delta_I(q)$=Pedro II.

$\nu_I(m^1p^1a=m^1b)=F$, pois a avó paterna de Aristóteles não é a mesma pessoa que a mãe de Kant.

Como vemos, o valor de verdade de uma sentença atômica segundo uma interpretação fica perfeitamente determinada pela interpretação, uma vez que é estabelecido por meio de regras precisas formuladas em função das denotações dos termos envolvidos e dos conjuntos e relações atribuídos aos predicados pela interpretação em causa.

[1] A expressão "o valor de verdade de α segundo I" é abreviada por $\nu_I(\alpha)$.

III. Gramática de $\mathcal{L}$

1. Gramática de $\mathcal{L}$

Retornamos à gramática para apresentar a definição de *fórmula* de $\mathcal{L}$. Veremos que essa será também uma definição indutiva e, assim, será dada em dois passos:

1° passo: especificação de um conjunto básico ou original;

2° passo: formulação das regras que permitem obter elementos mais complexos a partir de elementos menos complexos.

Definição de fórmula

1) As fórmulas atômicas são fórmulas;

2) Se α e β são fórmulas e § é uma variável, então,

a) $\neg\alpha$ é uma fórmula;

b) $(\alpha \wedge \beta)$, $(\alpha \vee \beta)$, $(\alpha \to \beta)$, $(\alpha \leftrightarrow \beta)$ são fórmulas;

c) $\forall§\alpha$ e $\exists§\alpha$ são fórmulas.

Dizemos que:

$\neg\alpha$ é a *negação* de α;

$(\alpha \wedge \beta)$ é a *conjunção* de α e β; α e β são ditos os *conjuntivos* de $(\alpha \wedge \beta)$;

$(\alpha \vee \beta)$ é a *disjunção* de α e β; α e β são ditos os *disjuntivos* de $(\alpha \vee \beta)$;

$(\alpha \to \beta)$ é a *implicação* de β por α; α é dito o *antecedente*, β o *consequente* de $(\alpha \to \beta)$;

$(\alpha \leftrightarrow \beta)$ é a equivalência entre α e β.

Observação:

A implicação também é chamada *condicional*; a equivalência também é chamada *bicondicional* ou *bi-implicação*.

Como sabemos, as expressões abaixo são fórmulas atômicas:

$F^2ab \quad a = x \quad G^2f^1ay_1 \quad H^3x_2g^2aby \quad y = f^1z$

Portanto, são fórmulas. Aplicando reiteradamente as regras de obtenção de fórmulas mais complexas, construimos as fórmulas abaixo:

$\neg F^2ab \quad \neg a = x \quad \neg\neg a = x \quad \neg\neg\neg a = x$

$(F^2ab \wedge a = x) \quad (P \wedge H^3x_2g^2aby) \quad (\neg a = x \wedge y = f^1z)$

$(P \vee \neg\neg a = x) \quad (((F^2ab \wedge a = x) \vee \neg P) \quad (\neg\neg\neg a = x \vee (F^2ab \wedge a = x))$

$(G^2f^1ay_1 \to y = f^1z) \quad ((P \vee \neg\neg a = x) \to \neg F^2ab)$

$(P \leftrightarrow \neg y = f^1z) \quad ((F^2ab \wedge a = x) \leftrightarrow (G^2f^1ay_1 \to y = f^1z))$

$\forall x\, a = x \quad \forall y\, (P \wedge H^3xg^2aby) \quad \forall z\, (\neg a = x \wedge y = f^1z)$

$\exists y\, P \quad \exists y\, (G^2f^1ay_1 \to y = f^1z) \quad \exists x\, \forall y\, (P \wedge H^3xg^2aby)$

$\neg (P \vee \neg\neg a = x) \quad \neg \forall x\, a = x \quad \neg \exists y\, (G^2f^1ay_1 \to y = f^1z)$

etc...

As expressões da forma Q§, onde $Q \in \{\forall, \exists\}$ e § é uma variável são ditas *quantificadores*. $\forall$§ é dito um *quantificador* universal com respeito a § e $\exists$§ um *quantificador existencial* com respeito a §.

Exemplos: $\forall x \quad \exists y \quad \forall x_1$

<u>Cuidado</u>: Quantificadores não são fórmulas!!!

Observação:

Os símbolos auxiliares de pontuação são usados, nas fórmulas, apenas quando formamos uma fórmula mais complexa a partir de duas mais simples por mais de um dos conectivos binários [$\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$]. Assim, cada conectivo binário é usado com um par próprio de parênteses.

Na leitura de uma fórmula complexa, pode ser de especial importância a identificação de cada par de parênteses e de cada conectivo binário correspondente ao par. Para tanto, podemos usar o algoritmo abaixo descrito:

Algorítmo de ordenação e emparelhamento dos parênteses numa fórmula.

1) Numere, da esquerda para a direita em ordem crescente cada um dos parênteses que abrem que ocorrem em α. Se *n* é o número mais alto obtido, então repita *n* vezes o procedimento 2, abaixo:
2) Da esquerda para a direita, procure o primeiro dos parênteses que fecham ainda não numerados; no segmento que antecede este símbolo, procure o número mais alto ainda não usado; numere o símbolo com este número; no interior do par de parênteses com este número deve haver um único conectivo binário ainda não-numerado; numere-o também com este número.

Por meio deste algoritmo, procedemos a um emparelhamento dos parêntese, assim como a uma associação de cada par deles ao conectivo binário correspondente.

Observemos que se esse algoritmo não puder ser corretamente levado a termo, a expressão em questão não é uma fórmula. Assim, a boa aplicação desse algoritmo constitui uma condição necessária (mas não suficiente) da correção gramatical de uma fórmula que contém conectivos binários.

Como vimos, as fórmulas podem ser atômicas ou geradas a partir de atômicas por meio da aplicação sucessiva de regras de introdução de quantificadores e/ou conectivos.

Dizemos então que a *complexidade* de uma fórmula α é o número de ocorrências de conectivos e/ou quantificadores em α; se *n* é esse número, escrevemos: Cpl [α] = n.

Assim:

a) as fórmulas atômicas têm complexidade 0 (zero); e

b) $\text{Cpl}\,[\neg\alpha] = \text{Cpl}\,[\alpha] + 1$

c) $\text{Cpl}\,[\alpha * \beta] = \text{Cpl}\,[\alpha] + \text{Cpl}\,[\beta] + 1$, para $* \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$

d) $\text{Cpl}\,[Q\S\alpha] = \text{Cpl}\,[\alpha] + 1$, para $Q \in \{\forall, \exists\}$

Se α tem complexidade maior que 0 (zero) então α é de uma das seguintes formas: $\neg\beta$, $(\beta_1 * \beta_2)$, $Q\S\beta$; ou seja, o símbolo mais à esquerda de α é um dos seguintes quatro: $\neg$, (, $\forall$, $\exists$. Nos primeiros dois casos, dizemos que α é *molecular*; nos últimos dois, que α é *geral*.

Exemplos:

a) As fórmulas abaixo são moleculares:

$\neg F^1x$, $(P \rightarrow (\forall x F^1x \vee Q))$, $\neg\forall x\exists y R^2xy$

b) As fórmulas abaixo são gerais:

$\forall x \neg F^1x$, $\exists y(H^2xy \vee R^2ay)$, $\exists x\forall y R^2xy$

Introduzimos, agora, a noção de subfórmula de uma fórmula.

Dizemos que β é *subfórmula* de uma fórmula α se:

1) β é a própria α;
2) existe uma subfórmula de α de uma das seguintes formas:
   a) $\neg\beta$
   b) $(\beta * \beta')$
   c) $(\beta' * \beta)$
   d) $Q\S\beta$;

onde $* \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$ e $Q \in \{\forall, \exists\}$.

Exemplo:

Seja α a fórmula: $\forall x\,(\neg\exists y\,(H^2xy \rightarrow R^2ax) \leftrightarrow F^1x))$. Então, são as seguintes as subfórmulas de α:

(1) $\forall x\,(\neg\exists y\,(H^2xy \rightarrow R^2ax) \leftrightarrow F^1x))$

(2) $(\neg\exists y\,(H^2xy \rightarrow R^2ax) \leftrightarrow F^1x))$

(3) $\forall x\,(\neg\exists y\,(H^2xy \rightarrow R^2ax)$

(4) F^1x

(5) $\exists y\,(H^2xy \rightarrow R^2ax) \leftrightarrow F^1x)$

(6) $(H^2xy \rightarrow R^2ax)$

(7) H^2xy

(8) R^2ax

Explicação: (1) é subfórmula de α porque é a própria α; como (1) é da forma $\forall\S(2)$, (2) é subfórmula de α; como (2) é da forma ((3) $\leftrightarrow$ (4)), (3) e (4) são subfórmulas de α; como (3) é da forma $\exists\S(6)$, (6) é subfórmula de α; como (6) é da forma ((7) $\rightarrow$ (8)), (7) e (8) são subfórmulas de α.

Veremos, em seguida, como construir o que chamamos *árvore de geração* ou *árvore de subfórmulas* de uma fórmula α.

2. Árvore de geração ou árvore de subfórmulas de α

Seja α uma fórmula. Para construir a árvore de subfórmulas (árvore de geração) de α, procedemos da seguinte maneira:

1) Emparelhamos todos os pares de parênteses que porventura ocorrem em α, associando-os a seus respectivos conectivos binários.
2) O nó inicial da árvore é formado pela própria α;
3) Se β é a fórmula que ocorre em um nó da árvore então: se β é atômica, o nó é dito *terminal*; se β não é atômica, então, se o primeiro símbolo que ocorre em β é:
   - 3.1) $\neg$: acresentamos um novo nó com a expressão que se segue;
   - 3.2) (: acrescentamos dois novos nós com as expressões que se encontram entre (e o conectivo * e entre * e) — (, *,) são associados ao mesmo número.
   - 3.3) Q : acrescentamos um novo nó com o que se segue à variável § que está imediatamente depois de Q.

A árvore estará terminada após *n* passos, onde n = Cpl (α).

Exemplo:

α é $\forall x\,(F^1x \rightarrow \neg(G^1x \vee P))$

Árvore de α: $\forall x\,(F^1x \rightarrow \neg(G^1x \vee P))$

$(F^1x \rightarrow \neg(G^1x \vee P))$

F^1x $\qquad\qquad$ $\neg(G^1x \vee P)$

$(G^1x \vee P)$

G^1x $\qquad\qquad$ P

Nós terminais: F^1x, G^1x, P.

Outro exemplo:

α é $\neg((\exists x\; a = x \vee R^2ax) \wedge \forall y\,\neg(a = y \rightarrow R^2xy))$

Árvore de α:

$$(1)\quad \neg((\exists x\ a = x \vee R^2ax) \wedge \forall y\ \neg(a = y \rightarrow R^2xy))$$

$$(2)\quad ((\exists x\ a = x \vee R^2ax) \wedge \forall y\ \neg(a = y \rightarrow R^2xy))$$

$$(3)\quad (\exists x\ a = x \vee R^2ax) \qquad (4)\quad \forall y\ \neg(a = y \rightarrow R^2xy)$$

$$(5)\quad \exists x\ a = x \qquad (6)\quad R^2ax \qquad (7)\quad \neg(a = y \rightarrow R^2xy)$$

$$(8)\quad a = x \qquad (9)\quad (a = y \rightarrow R^2xy))$$

$$(10)\quad a = y \qquad (11)\quad R^2xy$$

Note-se que uma expressão é uma fórmula se e só se tiver uma árvore de subfórmulas assim construída contendo fórmulas atômicas nos seus terminais.

Note-se, ainda, que a árvore de uma dada fórmula α tem uma certa configuração gráfica que corresponde à estrutura gramatical de α. Assim a árvore do exemplo acima tem a seguinte configuração:

1

2

3 4

5 6 7

8 9

10 11

Nessa árvore, os nós (8), (6), (10) e (11) são nós terminais. Como em qualquer árvore, (1) é o nó inicial.

Um *ramo* da árvore é uma seqüência de nós que vai do nó inicial a um nó terminal. Dessa forma, haverá numa árvore tantos ramos quantos sejam os seus terminais. No exemplo, os quatro ramos são:

a) (1), (2), (3), (5), (8)
b) (1), (2), (3), (6)
c) (1), (2), (4), (7), (9), (10)
d) (1), (2), (4), (7), (9), (11)

Observemos, ainda, que se uma ocorrência de β e uma ocorrência de β' estiverem num mesmo ramo, β' é subfórmula de β se e só se a ocorrência de β preceder a de β' no ramo.

Assim, no exemplo anterior, observa-se claramente que: (8) é subfórmula de (5), mas não de (6); (5) e (6) são subfórmulas de (3), mas (7) não é subfórmula de (3); e todas as onze fórmulas são subfórmula de (1).

Um outro tipo de árvore que pode ser definida é a *árvore molecular* de uma fórmula. A árvore molecular difere da árvore de subfórmula por não admitir decomposição por quantificadores; assim, seus terminais serão atômicos ou gerais.

A definição da construção de uma árvore molecular pode ser obtida a partir da definição da construção de subfórmulas, eliminando-se a cláusula 3.3. Portanto, é a seguinte a árvore molecular de $\neg((\exists x\ a = x \vee R^2ax) \wedge \forall y(a = y \rightarrow R^2yx))$:

$$(1)\quad \neg((\exists x\ a = x \vee R^2ax) \wedge \forall y(a = y \rightarrow R^2yx))$$

$$(2)\quad ((\exists x\ a = x \vee R^2ax) \wedge \forall y(a = y \rightarrow R^2yx))$$

$$(3)\quad (\exists x\ a = x \vee R^2ax) \qquad (4)\quad \forall y(a = y \rightarrow R^2yx)$$

$$(5)\quad \exists x\ a = x \qquad (6)\quad R^2ax$$

E a árvore molecular de $\forall x\ (F^1x \rightarrow \neg(G^1x \vee P))$ tem apenas um nó, constando, assim, apenas dessa mesma fórmula.

Vamos agora analisar os dois distintoss modos segundo os quais uma variável pode ocorrer numa fórmula.

3. Ocorrência ligada de uma variável § numa fórmula α

Dizemos que uma ocorrência de uma variável § numa fórmula α é *ligada* se ela se dá numa subfórmula de α da forma $\forall§\beta$ ou $\exists§\beta$. Se a ocorrência de § em α não é ligada, ela se diz *livre*.

Exemplos:

Na fórmula '$(\forall x(F^2xy \rightarrow G^1x) \vee \exists yR^2yx)$', a variável 'x' tem quatro ocorrências. As três primeiras são ligadas porque se dão dentro da subfórmula '$\forall x(F^2xy \rightarrow G^1x)$' e a última é livre; a variável 'y' tem três ocorrências; a primeira é livre, as demais são ligadas, pois se dão dentro da subfórmula '$\exists yR^2yx$'.

Observemos que uma ocorrência de uma variável § numa fórmula α é ligada (ou livre) se e somente se ela for ligada (ou livre) no respectivo terminal da árvore molecular de α. Analisemos mais um exemplo:

$$\neg(\forall x(R^2xa \wedge F^1y) \leftrightarrow \exists z(R^2xz \vee \forall yG^1y))$$

$$(\forall x(R^2xa \wedge F^1y) \leftrightarrow \exists z(R^2xz \vee \forall yG^1y))$$

$$\forall x(R^2xa \wedge F^1y) \qquad\qquad \exists z(R^2xz \vee \forall yG^1y)$$

Assim, as duas primeiras ocorrências de 'x' se dão no terminal '$\forall x(R^2xa \wedge F^1y)$; portanto, são ligadas; a última se dá no terminal $\exists z(R^2xz \vee \forall yG^1y)$; portanto, são livres. A primeira ocorrência de 'y' se dá em '$\forall x(R^2xa \wedge F^1y)$; assim, é livre; as demais se dão em $\exists z(R^2xz \vee \forall yG^1y)$; ora essa fórmula tem a subfórmula '$\forall yG^1y$', onde se dão as ocorrências; então elas são ligadas. Claramente, as ocorrências de 'z' são todas ligadas.

Podemos, agora, definir o conceito de sentença de $\mathcal{L}$. Uma *sentença* de $\mathcal{L}$ é uma fórmula de $\mathcal{L}$ onde nenhuma ocorrência de variável é livre.

Observação

Nas fórmulas atômicas todas as ocorrências de variáveis são livres. Assim, as sentenças atômicas são fórmulas atômicas que não contêm variáveis. As ocórrências ligadas de variáveis só aparecem em fórmulas complexas que contêm quantificadores.

Exemplo de sentenças:

$P \qquad F^1a \qquad R^2ab \qquad \neg R^2cb$

$\forall xF^1x \qquad \exists x(F^1x \vee G^1x) \qquad \forall x\exists yR^2xy \qquad \forall x(P \wedge Q)$

$(\exists xFx \rightarrow \forall y(\exists zR^2yz \rightarrow \exists wR^2wy))$

Uma relação importante entre as sentenças de $\mathcal{L}$ e suas árvores moleculares é a seguinte: uma fórmula α é uma sentença se e somente se todos os terminais de sua árvore molecular são sentenças. Essa propriedade decorre claramente do fato de que na decomposição molecular nenhum quantificador é eliminado.

Seja α uma sentença. Dizemos que β é um *componente básico* de α se β ocorre pelo menos uma vez como terminal da árvore molecular de α.

Exemplo:

Seja α a fórmula $(\exists xF^1x \rightarrow (\neg\forall yR^2ya \vee (\neg\exists xF^1x \vee F^1a)))$

$\exists xF^1x$ $\quad$ $(\neg\forall yR^2ya \vee (\neg\exists xF^1x \vee F^1a))$

$\neg\forall yR^2ya$ $\quad$ $(\neg\exists xF^1x \vee F^1a)$

$\forall yR^2ya$ $\quad$ $\neg\exists xF^1x$ $\quad$ F^1a

$\exists xF^1x$

Os componentes básicos são: $\exists xF^1x$, $\forall yR^2ya$, F^1a

Tomemos um outro exemplo.

Seja α a fórmula $((\forall x(G^1a \wedge F^1x) \rightarrow \exists y(\neg F^1a \vee G^1y)) \leftrightarrow (\exists yG^1y \rightarrow (F^1a \vee \forall xG^1x)))$.

$(\forall x(G^1a \wedge F^1x) \rightarrow \exists y(\neg F^1a \vee G^1y))$ $\quad$ $(\exists yG^1y \rightarrow (F^1a \vee \forall xG^1x))$

$\forall x(G^1a \wedge F^1x)$ $\quad$ $\exists y(\neg F^1a \vee G^1y)$ $\quad$ $\exists yG^1y$ $\quad$ $(F^1a \vee \forall xG^1x)$

F^1a $\quad$ $\forall xG^1x$

Os componentes básicos de α são: $\forall x(G^1a \wedge F^1x)$, $\exists y(\neg F^1a \vee G^1y)$, $\exists yG^1y$, F^1a, $\forall xG^1x$.

Observe-se que, embora a sentença 'G^1a' seja uma subfórmula de α que é uma sentença, 'G^1a' não é um componente básico de α, pois não ocorre nem uma vez como terminal da árvore molecular de α.

4. Sentenças CS

Seja α uma sentença formada apenas com letras sentenciais e/ou conectivos. Então α é dito uma *fórmula CS* ou uma *sentença CS*. Nesse caso, a noção de fórmula e de sentença coincidem, uma vez que não pode haver variáveis em α. A expressão 'CS' abrevia 'cálculo sentencial'. O cálculo sentencial é um subcálculo do Cálculo de Predicados e será estudado proximamente.

As sentenças CS foram aqui definidas a partir das fórmulas do cálculo dos predicados. No entanto, podemos dar diretamente uma definição indutiva para elas, como se segue:

1) As letras sentenciais são sentenças CS;

2) Se α e β são sentenças CS, então

a) $\neg\alpha$ é uma sentença CS;

b) $(\alpha * \beta)$ é uma sentença CS, para $* \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$.

Seja, agora, α uma sentença do Cálculo de Predicados. Dizemos que β é uma CS-associada de α se β pode ser obtida a partir de α pela substituição de cada componente básico de α por uma (distinta) letra sentencial.

Exemplos:

1) Seja α a sentença $((\neg\forall xF^1x \rightarrow \exists y\forall xR^2xy) \vee (\forall xF^1x \wedge R^2ab))$.

Seus componentes básicos são: $\forall xF^1x$, $\exists y\forall xR^2xy$, R^2ab.

Substituindo-os pelas letras sentenciais: P, Q, R, obtemos a CS-associada: $((\neg P \rightarrow Q) \vee (P \wedge R))$.

Obviamente, se houvéssemos escolhido as letras A, B, C, obteríamos uma outra CS-associada: $((\neg A \rightarrow B) \vee (A \wedge C))$.

2) Seja α: $\forall x((F^1x \wedge G^1x) \rightarrow \exists yR^2xy)$.

Como α é seu único componente básico, qualquer letra sentencial é uma CS-associada de α.

3) Seja α: $\neg(\forall x(F^1x \rightarrow (G^1a \wedge H^1a)) \leftrightarrow (\neg G^1a \rightarrow (G^1a \vee \exists xF^1x)))$.

Os componentes básicos de α são: $\forall x(F^1x \rightarrow (G^1a \wedge H^1a))$, G^1a, $\exists xF^1x$.

Então, um exemplo de CS-associada de α é a sentença: $\neg(P \leftrightarrow (\neg Q \rightarrow (Q \vee R)))$.

Observemos que se β é CS-associada de α, as árvore moleculares de α e β têm a mesma estrutura. Assim, β exibe a estrutura molecular de α.

II. Semântica Proposicional

Vamos, por um momento, deixar de lado a linguagem $\mathcal{L}$ completa para estudarmos o seu fragmento proposicional, i.e., o conjunto das sentenças CS.

Seja, então, $\mathbb{S}$ o conjunto das sentenças CS. Uma valoração booleana ν é uma função do conjunto $\mathbb{S}$ no conjunto $\{V, F\}$ tal que, para quaisquer $\alpha, \beta \in \mathbb{S}$, temos que:

a) $\nu[\neg\alpha] = V$ se e só se $\nu[\alpha] = F$
b) $\nu[(\alpha \wedge \beta)] = v$ se e só se $\nu[\alpha] = \nu[\beta] = V$
c) $\nu[(\alpha \vee \beta)] = V$ se e só se $\nu[\alpha] = V$ ou $\nu[\beta] = V$
d) $\nu[(\alpha \rightarrow \beta)] = V$ se e só se $\nu[\alpha] = F$ ou $\nu[\beta] = V$
e) $\nu[(\alpha \leftrightarrow \beta)] = V$ se e só se $\nu[\alpha] = \nu[\beta]$.

Como veremos, as valorações booleanas atribuem arbitrariamente valores às letras sentenciais e os valores atribuídos às sentenças complexas são univocamente determinados pelos valores atribuídos aos componentes mais simples. Em vista disso, uma valoração booleana é totalmente determinada pelos valores que atribui às letras sentenciais de $\mathbb{S}$. Isso se deve ao fato de que, em vista da definição acima, cada uma das cinco formas de fórmulas complexas é entendida como uma função (ou operação) cujos argumentos e valores estão em $\{V, F\}$. Em outras palavras, cada um dos conectivos é entendido como a expressão de uma *função de verdade*.

Assim, a negação representa a função de $\{V, F\}$ em $\{V, F\}$ definida pela tabela:

Argumento	Valor
α	$\neg\alpha$
V	F
F	V

Os quatro conectivos binários expressam funções de $\{V, F\}^2$ em $\{V, F\}$ definidas pelas tabelas abaixo:

Argumentos		Valores			
α	β	$(\alpha \wedge \beta)$	$(\alpha \vee \beta)$	$(\alpha \rightarrow \beta)$	$(\alpha \leftrightarrow \beta)$
V	V	V	V	V	V
V	F	F	V	F	F
F	V	F	V	V	F
F	F	F	F	V	V

Vamos, neste momento, abrir espaço para uma pequena incursão na teoria das funções de verdade. De modo geral, uma função de verdade n-ária é uma operação n-ária no conjunto $\{V, F\}$ ou seja, uma função $f: \{V, F\}^n \rightarrow \{V, F\}$.

Podemos verificar que há quatro funções de verdade unárias (abaixo indicadas por f_1^1, f_2^1, f_3^1, f_4^1):

	f_1^1		f_2^1		f_3^1		f_4^1
V	V	V	V	V	F	V	F
F	V	F	F	F	V	F	F
(tautologia)		*(identidade)*		*(negação)*		*(contradição)*	

Essas quatro funções recebem nomes, que aparecem acima entre parênteses. Podemos observar que pelo que foi estabelecido anteriormente, nas valorações booleanas o conectivo '$\neg$' expressa a função f_3^1, acima.

As funções de verdade binárias são dezesseis, as ternárias são 256; de modo geral, o número das funções de verdade n-árias é dado por $(2^2)^n$.

Vamos relacionar, em seguida, as 16 funções de verdade binárias. Para economizar espaço, usaremos uma mesma tabela, com quatro linhas, uma para cada combinação de valores dos argumentos e dezesseis colunas, uma para cada função binária, aqui denotadas por $f_1^2 \dots f_{16}^2$:

		f_1^2	f_2^2	f_3^2	f_4^2	f_5^2	f_6^2	f_7^2	f_8^2	f_9^2	f_{10}^2	f_{11}^2	f_{12}^2	f_{13}^2	f_{14}^2	f_{15}^2	f_{16}^2
V	V	V	V	V	V	F	V	V	V	F	F	F	V	F	F	F	F
V	F	V	V	V	F	V	V	F	F	V	V	F	F	V	F	F	F
F	V	V	V	F	V	V	F	V	F	V	F	V	F	F	V	F	F
F	F	V	F	V	V	V	F	F	V	F	V	V	F	F	F	V	F

Podemos observar que:

'$\wedge$' expressa a função f_{12}^2
'$\vee$' expressa a função f_2^2
'$\rightarrow$' expressa a função f_4^2
'$\leftrightarrow$' expressa a função f_8^2

Entre as funções unárias e binárias, f_3^1, f_2^2, f_4^2, f_8^2, e f_{12}^2 são as funções diretamente expressáveis. As outras, no entanto, são também expressáveis por meio de composição, a partir dessas cinco. Assim, por exemplo, a função f_5^2 é obtida pela composição de f_3^1 com f_{12}^2, como se vê na tabela abaixo:

		f_{12}^2	$f_3^1 f_{12}^2$
V	V	V	F
V	F	F	V
F	V	F	V
F	F	F	V

Na verdade, é possível mostrar que qualquer função de verdade n-ária pode ser obtida por meio de uma composição reiterada, a partir de algumas funções unárias e binárias, tomadas como primitivas. Um resultado conhecido é de que com a função f_5^2 é possível compor qualquer função n-ária de verdade; e o mesmo vale para a função f_{15}^2. Também podemos obter todas as funções de verdade usando os elementos do conjunto $\{f_3^1, f_{12}^2\}$; o mesmo vale pare os conjuntos $\{f_3^1, f_2^2\}$ e $\{f_3^1, f_4^2\}$. Como, na nossa linguagem podemos expressar f_3^1, f_{12}^2, f_2^2, f_4^2, para cada função de verdade n-ária, teremos na nossa linguagem uma fórmula contendo n letras sentencias que expressa essa função.

Para exemplificar, sejam α, β, γ fórmulas quaisquer (eventualmente letras sentenciais). Então:

1) As quatro funções unárias podem ser expressas pelas fórmulas $(\alpha \vee \neg\alpha)$, α, $\neg\alpha$, $(\alpha \wedge \neg\alpha)$, como mostra a tabela abaixo:

f_2^1	f_3^1	f_1^1	f_4^1
α	$\neg\alpha$	$(\alpha \vee \neg\alpha)$	$(\alpha \wedge \neg\alpha)$
V	F	V	F
F	V	V	F

2) a) A função f_1^2 pode ser expressa por $((\alpha \vee \neg\alpha) \vee (\beta \vee \neg\beta))$;

b) As funções f_2^2, f_3^2, f_4^2 e f_5^2 podem ser expressas por $(\alpha \wedge \beta)$, $(\alpha \wedge \neg\beta)$, $(\neg\alpha \wedge \beta)$, $(\neg\alpha \wedge \neg\beta)$, como mostra a tabela:

				f_2^2	f_3^2	f_4^2	f_5^2
α	β	$\neg\alpha$	$\neg\beta$	$(\alpha \wedge \beta)$	$(\alpha \wedge \neg\beta)$	$(\neg\alpha \wedge \beta)$	$(\neg\alpha \wedge \neg\beta)$
V	V	F	F	V	F	F	F
V	F	F	V	F	V	F	F
F	V	V	F	F	F	V	F
F	F	V	V	F	F	F	V

c) As funções $f_6^2, f_7^2, f_8^2, f_9^2, f_{10}^2, f_{11}^2$ podem ser expressas por

				f_6^2	f_7^2	f_8^2	f_9^2	f_{10}^2	f_{11}^2
α	β	$\neg\alpha$	$\neg\beta$	$(\alpha \wedge (\beta \vee \neg\beta))$	$((\alpha \vee \neg\alpha) \wedge \beta)$	$(\alpha \leftrightarrow \beta)$	$\neg(\alpha \leftrightarrow \beta)$	$((\alpha \vee \neg\alpha) \wedge \neg\beta)$	$(\alpha \wedge (\beta \vee \neg\beta))$
V	V	F	F	V	V	V	F	F	F
V	F	F	V	V	F	F	V	V	F
F	V	V	F	F	V	F	V	F	V
F	F	V	V	F	F	V	F	V	V

d) As funções $f_{12}^2, f_{13}^2, f_{14}^2$ e f_{15}^2 podem ser expressas pelas negações das fórmulas que expressam f_5^2, f_4^2, f_3^2, f_2^2, respectivamente;

e) A função f_{16}^2 pode ser expressa pela negação de uma função que expressa f_1^2.

3) Não vamos aqui examinar cada uma das funções ternárias ou n-árias para $n > 3$. A título de exemplo, apenas, consideremos duas funções ternárias, *g* e *h*, tais que:

g recebe o valor V para a trinca {V, F, V} e recebe o F nos demais casos;

h recebe V para {V, F, V} e para {F, F, F} e recebe F nos demais casos.

Então ao formular $((\alpha \wedge \beta) \wedge \gamma)$ e $(((\alpha \wedge \neg\beta) \wedge \gamma) \vee ((\neg\alpha \wedge \neg\beta) \wedge \neg\gamma))$ expressam, respectivamente, as funções *g* e *h*, como se vê na tabela abaixo:

								g		h
α	β	γ	$\neg\alpha$	$\neg\beta$	$\neg\gamma$	$(\alpha\wedge\neg\beta)$	$(\neg\alpha\wedge\neg\beta)$	$((\alpha\wedge\neg\beta)\wedge\gamma)$	$((\neg\alpha\wedge\neg\beta)\wedge\neg\gamma)$	$(((\alpha\wedge\neg\beta)\wedge\gamma)\vee((\neg\alpha\wedge\neg\beta)\wedge\neg\gamma))$
V	V	V	F	F	F	F	F	F	F	F
V	V	F	F	F	V	F	F	F	F	F
V	F	V	F	V	F	V	F	V	F	V
V	F	F	F	V	V	V	F	F	F	F
F	V	V	V	F	F	F	F	F	F	F
F	V	F	V	F	V	F	F	F	F	F
F	F	V	V	V	F	F	V	F	F	F
F	F	F	V	V	V	F	V	F	V	V

4) Nos exemplos anteriores, as fórmulas apontadas como expressões das funções indicadas eram apenas algumas entre classes de fórmulas que cumprem o mesmo papel. Assim, a função f_{12}^{2}, por exemplo, pode ser expressa por uma infinidade de diferentes fórmulas, entre as quais figuram: $(\alpha \wedge \beta)$, $\neg(\neg\alpha \vee \neg\beta)$, $\neg(\alpha \rightarrow \neg\beta)$, etc.

Nossa linguagem dispõe, então, de cinco conectivos que expressam diretamente uma função unária, f_3^{1}, e quatro funções binárias, f_2^{2}, f_4^{2}, f_8^{2}, f_{12}^{2}. Mostremos, agora, que, embora úteis, alguns desses conectivos não seriam em princípio necessários. Para isso, sejam:

1) $\mathcal{L}_{\neg\wedge}$: a linguagem que contém letras sentenciais e os conectivos '$\neg$', '$\wedge$';

2) $\mathcal{L}_{\neg\vee}$: a linguagem que contém letras sentenciais e os conectivos '$\neg$', '$\vee$';

3) $\mathcal{L}_{\neg\rightarrow}$: a linguagem que contém letras sentenciais e os conectivos '$\neg$', '$\rightarrow$'.

Mostremos que cada uma dessas linguagens pode definir os demais conectivos. Assim:

1) Em $\mathcal{L}_{\neg\wedge}$, $(\alpha \vee \beta) \overset{\text{df}}{\equiv} \neg(\neg\alpha \wedge \neg\beta)$
$(\alpha \rightarrow \beta) \overset{\text{df}}{\equiv} \neg(\alpha \wedge \neg\beta)$
$(\alpha \leftrightarrow \beta) \overset{\text{df}}{\equiv} \neg(\neg\alpha \wedge \beta) \wedge \neg(\neg\alpha \wedge \neg\beta))$

2) Em $\mathcal{L}_{\neg\vee}$, $(\alpha \wedge \beta) \overset{\text{df}}{\equiv} \neg(\neg\alpha \vee \neg\beta)$
$(\alpha \rightarrow \beta) \overset{\text{df}}{\equiv} (\neg\alpha \vee \beta)$
$(\alpha \leftrightarrow \beta) \overset{\text{df}}{\equiv} \neg(\neg\alpha \vee \neg\beta) \vee \neg(\alpha \vee \beta))$

3) Em $\mathcal{L}_{\neg\rightarrow}$, $(\alpha \wedge \beta) \overset{\text{df}}{\equiv} \neg(\alpha \rightarrow \neg\beta)$
$(\alpha \vee \beta) \overset{\text{df}}{\equiv} (\neg\alpha \rightarrow \beta)$
$(\alpha \leftrightarrow \beta) \overset{\text{df}}{\equiv} \neg((\alpha \rightarrow \beta) \rightarrow \neg(\neg\alpha \rightarrow \neg\beta))$

A razão pela qual usamos os cinco conectivos está ligada ao fato de que com eles abreviamos muitas fórmulas (veja-se por exemplo o comprimento das ddefinições de uma bicondicional nas linguagens acima!). Pode, assim, ser mais cômodo trabalhar com um número maior de conectivos. No entanto, em vez de tomar como primitivos os cinco conectivos - ou seja, em vez de adotar a linguagem $\mathcal{L}_{\neg,\wedge,\vee,\rightarrow,\leftrightarrow}$, poderíamos ter preferido trabalhar com $\mathcal{L}_{\neg\wedge}$, com $\mathcal{L}_{\neg\vee}$ ou com $\mathcal{L}_{\neg\rightarrow}$.

Retornemos às valorações booleanas. Seja α uma fórmula CS com n letras sentenciais, $\Pi_1, ..., \Pi_n$. Então:

1) Como todos os conectivos se comportam como funções de verdade se conhecermos os valores que uma valoração booleana atribui às letras sentenciais $\Pi_1, ..., \Pi_n$, podemos calcular o valor que ela atribui a cada uma das fórmulas e conectivos; consequentemente, podemos calcular o valor de α nessa valoração.

Exemplo:

Seja $\alpha = (P \rightarrow (\neg Q \vee R))$ e seja v a valoração que atribui V a 'P' e a 'Q' e F a 'R'.

	P	Q	R	$\neg Q$	$(\neg Q \vee R)$	$(P \rightarrow (\neg Q \vee R))$
v	V	V	F	F	F	F

2) Notemos que esse será também o comportamento de qualquer outra valoração v' que concorda com v nas letras sentenciais que ocorrem em α. Podemos, então, enunciar um princípio geral:

Para quaisquer valorações booleanas, v, v', se elas concordam nos valores que atribuem a cada uma das letras sentenciais que ocorrem em uma sentença CS α, então elas concordam no valor que atribuem a α.

3) Ora, toda fórmula CS α tem um número finito de letras sentenciais, digamos, *n*. Como há apenas dois valores de verdade, existem exatamente 2^n combinações de {V, F} em grupos de *n*, ou seja, há exatamente 2^n n-uplas formadas com os valores do conjunto {V, F}. Se quisermos, portanto, estudar o comportamento de uma fórmula no conjunto de todas as valorações booleanas, será suficiente considerarmos 2^n valorações que diferem entre si por pelo menos um dos valores atribuídos a uma letra sentencial de α - pois qualquer valoração booleana concordará nas letras sentenciais de α com alguma dessas 2^n valorações consideradas.

Por exemplo:

Tomemos $\alpha = (P \leftrightarrow (Q \vee \neg P))$. Como temos duas letras sentenciais envolvidas, há 2^2 possibilidades distintas de atribuir os elementos de {V, F} aos elementos de {P, Q}, esquematizados abaixo:

	P	Q
1	V	V
2	V	F
3	F	V
4	F	F

Cada uma das letras, de 1 a 4, representa uma atribuição possível dos valores de {V, F} aos elementos de {P, Q}.

Com respeito ao conjunto {P, Q}, as valorações booleanas dividem-se portanto em quatro grupos (ou classes de equivalência em {P, Q}), cada um deles representado por uma das linhas da tabela acima. Nela, a linha 1 representa todas as valorações que atribuem V a 'P' e a 'Q'; a linha 2, todas as valorações que atribuem V a 'P' e F a 'Q'; e assim por diante.

4) Podemos, então, definir o que é uma tabela de verdade para uma fórmula α. Se todas as letras sentenciais de uma fórmula CS α estão no conjunto $\{\Pi_1, ..., \Pi_n\}$, uma tabela de verdade para α é uma construção contendo $1 + 2^n$ linhas tais que: a primeira linha leva a numeração 0 (zero) e contém as letras sentenciais $\Pi_1, ..., \Pi_n$; essa linha é dita o *cabeçalho* da tabela. As demais linhas, numeradas de 1 a 2^n, contêm as 2^n n-uplas que podem ser formadas com os elementos de {V, F}, obtidas pelo seguinte método:

- na coluna correspondente a Π_1, há 2^{n-1} ocorrências de 'V' seguidas de 2^{n-1} ocorrências de 'F';
- na coluna correspondente a Π_2, há 2^{n-2} ocorrências de 'V', seguidas por 2^{n-2} ocorrências de 'F', seguidas por 2^{n-2} ocorrências de 'V', seguidas por 2^{n-2} ocorrências de 'F'.
- na coluna correspondente a Π_n, há 2^{n-n} ocorrências de 'V', seguidas por 2^{n-n} ocorrências de 'F', seguidas por 2^{n-n} ocorrências de 'V', ... , seguidas por 2^{n-n} ocorrências de 'F'.

O procedimento acima poderia ainda ser resumido pela seguinte receita: para obter a coluna da i-ésima letra sentencial, Π_i, onde $1 \leq i \leq n$, procedemos da seguinte maneira: preenchemos a coluna com 2^{n-i} ocorrências de 'V' seguidas por 2^{n-i} ocorrências de 'F'; aplicando 2^{i-1} vezes este procedimento, preencheremos as 2^n vagas da coluna. Assim, por exemplo, se as letras sentenciais são os elementos do conjunto {P, Q, R}, construímos:

0	P	Q	R
1	V	V	V
2	V	V	F
3	V	F	V
4	V	F	F
5	F	V	V
6	F	V	F
7	F	F	V
8	F	F	F

A linha 0 é o cabeçalho da tabela.

A coluna 1 foi obtida por 2^{1-1} (i.e. 2^0, i.e. 1) aplicações do procedimento: escrever 2^{3-1} vezes o 'V' e 2^{3-1} vezes o 'F'.

A coluna 2 foi obtida por 2^{2-1} (i.e. 2^1, i.e. 2) aplicações do procedimento: escrever 2^{3-2} vezes o 'V' e 2^{3-2} o 'F'.

A coluna de 'R' foi obtida por 2^{3-1} (i.e. 2^2, i.e. 4) aplicações do procedimento: escrever 2^{3-3} (i.e. 1) vezes o valor 'V' e 2^{3-3} vezes o valor 'F'.

5) Numa tabela de verdade para α podemos, então, calcular o comportamento de α em todas as valorações booleanas. Para tanto, basta construir a tabela e ir estendendo o cabeçalho até que todas as subfórmulas de α dele constem. Em seguida, procedemos ao cálculo do comportamento das subfórmulas de α segundo as regras para os conectivos definidas pelas valorações booleanas.

Exemplo:

Seja α a fórmula $(((\neg P \wedge Q) \rightarrow (S \rightarrow R)) \vee \neg Q)$.

0	P	Q	R	S	$\neg P$	$\neg Q$	$(\neg P \wedge Q)$	$(S \rightarrow R)$	$((\neg P \wedge Q) \rightarrow (S \rightarrow R))$	$(((\neg P \wedge Q) \rightarrow (S \rightarrow R)) \vee Q)$
1	V	V	V	V	F	F	F	V	V	V
2	V	V	V	F	F	F	F	V	V	V
3	V	V	F	V	F	F	F	F	V	V
4	V	V	F	F	F	F	F	V	V	V
5	V	F	V	V	F	V	F	V	V	V
6	V	F	V	F	F	V	F	V	V	V
7	V	F	F	V	F	V	F	F	V	V
8	V	F	F	F	F	V	F	V	V	V
9	F	V	V	V	V	F	V	V	V	V
10	F	V	V	F	V	F	V	V	V	V
11	F	V	F	V	V	F	V	F	F	V
12	F	V	F	F	V	F	V	V	V	V
13	F	F	V	V	V	V	F	V	V	V
14	F	F	V	F	V	V	F	V	V	V
15	F	F	F	V	V	V	F	F	V	V
16	F	F	F	F	V	V	F	V	V	V

Analogamente, seja Γ um conjunto finito de fórmulas CS e T uma tabela de verdade que contém em seu cabeçalho todas as letras sentenciais que ocorrem nos elementos de Γ. Então dizemos que T é uma tabela de verdade adequada para Γ ou, simplesmente, que T é uma tabela para Γ.

Introduziremos, agora, quatro conceitos básicos para o cálculo sentencial: os conceitos de modelo funcional-veritativo de um conjunto de fórmulas CS; o de tautologia; o de consistência funcional-veritativa; e o de conseqüência tautológica.

Seja α uma fórmula CS, Γ um conjunto de fórmulas CS.

1) Dizemos que uma valoração booleana v é um *modelo funcional-veritativo* de Γ se v atribui o valor de verdade V a todos os elementos de Γ.

   Exemplo:

   Seja Γ o conjunto $\{(P \leftrightarrow (\neg Q \wedge R)), (P \vee Q), (S \rightarrow \neg R)\}$. Então, qualquer valoração booleana que dê o valor V a 'P' e a 'R', e o valor F a 'Q' e a 'S' é um modelo funcional-veritativo de Γ, como se pode ver no cálculo abaixo:

P	Q	R	S	$\neg Q$	$\neg R$	$(\neg Q \wedge R)$	$(P \leftrightarrow (\neg Q \wedge R))$	$(P \vee Q)$	$(S \rightarrow \neg R)$
V	F	V	F	V	F	V	V	V	V

2) Dizemos que um conjunto de fórmulas CS Γ é *funcional-veritativamente consistente* se Γ tiver pelo menos um modelo funcional-veritativo. Assim, o conjunto Γ do exemplo acima é um conjunto funcional-veritativamente consistente.

   Observe-se que há conjuntos que não são veritativamente consistentes, i.e. que não têm nenhum modelo, i.e.: não há nenhuma valoração booleana que atribua o valor V a cada um dos seus elementos.

   Exemplos:

   O conjunto $\{P, \neg P\}$ não é funcional-veritativamente consistente pois nenhuma valoração booleana atribui V a 'P' e V a '$\neg$P'. Analogamente, o conjunto $\{P, (P \rightarrow \neg Q), Q\}$ não é funcional-veritativamente consistente pois toda valoração booleana que atribui V a 'P' e a 'Q' atribui F a '$(P \rightarrow \neg Q)$'.

   Se o conjunto Γ é finito, podemos usar uma tabela de verdade adequada para Γ com o intuito de testar a consistência funcional-veritativa de Γ: Γ será consistente se houver uma mesma linha da tabela na qual todos os elementos de Γ recebem V; uma linha como essa representa uma classe de equivalência de valorações booleanas que são modelos de Γ. Tomemos, por exemplo, o conjunto $\{(P \vee Q), (Q \rightarrow \neg R), (\neg P \rightarrow R)\}$:

0	P	Q	R	$\neg P$	$\neg R$	$(P \vee Q)$	$(Q \rightarrow \neg R)$	$(\neg P \rightarrow R)$	
1	V	V	V	F	F	V	F	V	
2	V	V	F	F	V	V	V	V	x
3	V	F	V	F	F	V	V	V	x
4	V	F	F	F	V	V	V	V	x
5	F	V	V	V	F	V	F	V	
6	F	V	F	V	V	V	V	F	
7	F	F	V	V	F	F	V	V	
8	F	F	F	V	V	F	V	F	

   Como vemos, as linhas 2, 3, 4 satisfazem a condição estipulada; assim, cada uma delas representa uma classe de modelos funcionais-veritativos de Γ.

Observações:

a) Para qualquer valoração booleana v, a sentença abaixo é verdadeira: para todo $\alpha \in \varnothing$, $v(\alpha) = V$. Ou seja, qualquer valoração booleana é modelo funcional-veritativo do conjunto vazio. Assim, $\varnothing$ é um conjunto consistente funcional-veritativamente.
b) O conjunto de todas as sentenças CS é um conjunto inconsistente, uma vez que contém fórmulas e suas negações.

3) Dizemos que uma fórmula CS é uma *tautologia* se α recebe o valor de verdade V em todas as valorações booleanas. Assim, α é uma tautologia se e só se a coluna correspondente a α, numa tabela de verdade que contenha todas as letras sentenciais de α, é formada apenas por ocorrências de 'V'.

Há um resultado simples e importante concernindo as tautologias: seja α uma fórmula, Π uma letra sentencial que ocorre em α e seja α' a fórmula que resulta de α pela substituição de todas as ocorrências de Π por ocorrências de β (indicamos essa última condição escrevendo: $\alpha' = \alpha^{\Pi}/_{\beta}$); <u>se α for uma tautologia, então α' também será uma tautologia</u>.

É fácil compreender por que essa afirmação vale. Se α' não for uma tautologia, existe uma valoração booleana ν'que falsificará α'; seja agora ν a valoração booleana que atribui a Π o valor que ν' atribui a β e que concorda com ν nos valores atribuídos às demais letras sentenciais de α; então, $\nu(\alpha) = \nu(\alpha') = F$. Assim, se existe ν' tal que $\nu'(\alpha') = F$, então existe ν tal que $\nu(\alpha) = F$. Portanto, se não existe ν tal que $\nu(\alpha) = F$, também não existe ν' tal que $\nu'(\alpha') = F$, ou seja: se α é tautologia, então α' também é tautologia.

Por causa do resultado que acabamos de expor, em vez de dar exemplos de fórmulas que são tautologias, costumamos apresentar esquemas de fórmulas tautológicas, i.e. esquemas cujas especificações são, todas, tautologias. Assim, por exemplo, o esquema '$(\alpha \vee \neg\alpha)$' é um esquema tautológico: todas as fórmulas da forma desse esquema, i.e. todas as fórmulas obtidas pela especificação da fórmula α, como '$(P \vee \neg P)$', $((P \wedge Q) \vee \neg(P \wedge Q))$, etc., são tautologias.

Há infinitos esquemas tautológicos. Alguns deles, no entanto, por serem simples e representarem leis lógicas fundamentais, recebem nomes especiais. Vamos apresentar três grupos de esquemas tautológicos.

<u>Grupo I</u>: Equivalências Tautológicas

$(\neg\neg\alpha \leftrightarrow \alpha)$	Dupla negação
$((\alpha * \beta) \leftrightarrow (\beta * \alpha))$	Comutação para $* \in \{\wedge, \vee, \leftrightarrow\}$
$(((\alpha * \beta) * \gamma) \leftrightarrow (\alpha * (\beta * \gamma))$	Associação para $* \in \{\wedge, \vee, \leftrightarrow\}$
$((\alpha \rightarrow \beta) \leftrightarrow (\neg\beta \rightarrow \neg\alpha))$	Contraposição
$(\neg(\alpha * \beta) \leftrightarrow (\neg \alpha \circ \neg\beta))$	de Morgan onde se $* = \wedge$ então $\circ = \vee$; se $* = \vee$ então $\circ = \wedge$
$(\neg(\neg\alpha * \neg\beta) \leftrightarrow (\alpha \circ \beta))$	
$((\alpha \circ (\beta * \gamma)) \leftrightarrow ((\alpha \circ \beta) * (\alpha \circ \gamma)))$	distributiva onde se $* = \wedge$ então $\circ = \vee$; se $* = \vee$ então $\circ = \wedge$
$((\alpha * \alpha) \leftrightarrow \alpha)$	Contração onde $* \in \{\wedge, \vee\}$
$(((\alpha \wedge \beta) \rightarrow \gamma) \leftrightarrow (\alpha \rightarrow (\beta \rightarrow \gamma)))$	Exportação/Importação
$((\alpha \rightarrow \beta) \leftrightarrow \neg(\alpha \wedge \neg\beta))$	Definição da implicação em $\mathcal{L}_{\neg\wedge}$
$((\alpha \rightarrow \beta) \leftrightarrow (\neg \alpha \vee \beta))$	Definição da implicação em $\mathcal{L}_{\neg\vee}$
$((\alpha \wedge \beta) \leftrightarrow \neg(\alpha \rightarrow \neg\beta))$	Definição da conjunção em $\mathcal{L}_{\neg\rightarrow}$
$((\alpha \vee \beta) \leftrightarrow (\neg\alpha \rightarrow \beta))$	Definição da disjunção em $\mathcal{L}_{\neg\rightarrow}$
$((\alpha * (\alpha * \beta)) \leftrightarrow (\alpha * \beta))$	Absorção para $* \in \{\wedge, \vee\}$
$((\alpha * (\alpha \circ \beta)) \leftrightarrow \alpha)$	onde se $* = \wedge$ então $\circ = \vee$; se $* = \vee$ então $\circ = \wedge$
$((\alpha \leftrightarrow \beta) \leftrightarrow ((\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)))$	
$((\alpha \leftrightarrow \beta) \leftrightarrow ((\alpha * \beta) \circ (\neg\alpha * \neg\beta)))$	onde se $* = \wedge$ então $\circ = \vee$; se $* = \vee$ então $\circ = \wedge$
$((\alpha \leftrightarrow \beta) \leftrightarrow (\neg\alpha \leftrightarrow \neg\beta))$	
$(\neg(\alpha \leftrightarrow \beta) \leftrightarrow (\neg\alpha \leftrightarrow \beta)$	

Grupo II: Implicações Tautológicas

$(\alpha \to (\beta \to \alpha))$	Prefixação por implicação
$((\alpha \land \beta) \to \alpha)$	Eliminação do segundo conjuntivo
$((\alpha \land \beta) \to \beta)$	Eliminação do primeiro conjuntivo
$(\alpha \to (\alpha \lor \beta))$	Introdução do segundo disjuntivo
$(\beta \to (\alpha \lor \beta))$	Introdução do primeiro disjuntivo
$(\neg\alpha \to (\alpha \to \beta))$	Lei de Duns Scott
$(\alpha \to (\neg\, \alpha \to \beta))$	Lei de Duns Scott
$((\alpha \to \beta) \to ((\alpha \to \neg\beta) \to \neg\alpha))$	Lei de Redução ao Absurdo
$((\neg\alpha \to \beta) \to ((\neg\alpha \to \neg\beta) \to \alpha))$	Lei de Redução ao Absurdo
$(\alpha \to (\beta \to (\alpha \land \beta)))$	Introdução da conjunção
$((\alpha \to \gamma) \to ((\beta \to \gamma) \to ((\alpha \lor \beta) \to \gamma)))$	Eliminação da disjunção
$((\alpha \to \neg\alpha) \to \neg\alpha)$	Redução de casos
$((\neg\alpha \to \alpha) \to \alpha)$	Redução de casos

Grupo III:

$(\alpha \lor \neg\alpha)$	Lei do Terceiro Excluído
$\neg(\alpha \land \neg\alpha)$	Lei da Não Contradição
$(\alpha \to \alpha)$	Lei da Identidade Proposicional

Ao lado das tautologias, consideremos aquelas fórmulas CS que são falsas em qualquer valoração booleana (como, por exemplo, as especificações do esquema '$(\alpha \land \neg\alpha)$'). Tais fórmulas são ditas *contradições*. Se, agora, representarmos por $\mathcal{T}$ um esquema tautológico e por $\mathbb{C}$ um esquema contraditório, teremos, ainda, a possibilidade de formular as seguintes leis proposicionais:

$$(\mathcal{T} \leftrightarrow \neg\mathcal{C})$$
$$((\alpha \land \mathcal{T}) \leftrightarrow \alpha)$$
$$((\alpha \lor \mathcal{T}) \leftrightarrow \mathcal{T})$$
$$((\alpha \land \mathcal{C}) \leftrightarrow \mathcal{C})$$
$$((\alpha \lor \mathcal{C}) \leftrightarrow \alpha)$$
$$((\alpha \to \mathcal{T}) \leftrightarrow \mathcal{T})$$
$$((\alpha \to \mathcal{C}) \leftrightarrow \neg\alpha)$$
$$((\mathcal{T} \to \alpha) \leftrightarrow \alpha)$$
$$((\mathcal{C} \to \alpha) \leftrightarrow \mathcal{T})$$

4) Dizemos que uma fórmula α é *conseqüência tautológica* de um conjunto de sentenças Γ se todo modelo funcional veritativo de Γ atribui o valor V a α.

Exemplo:

'Q' é uma conseqüência tautológica do conjunto $\{P, (P \to Q)\}$ pois: se ν é um modelo de $\{P, (P \to Q)\}$, então $\nu(P) = V$ e $\nu((P \to Q)) = V$; nesse caso, necessariamente, $\nu(Q) = V$.

Contra-exemplo:

'Q' não é uma conseqüência tautológica de $\{P, (Q \to P)\}$, pois existe uma valoração booleana ν tal que $\nu(P) = V$ e $\nu((P \to Q)) = V$, mas $\nu(Q) = F$; assim, ν é modelo de $\{P, (Q \to P)\}$, mas falsifica 'Q'.

Uma propriedade análoga à que expressamos acima, quando falamos das tautologias, vale para a relação de conseqüência tautológica.

Seja $\Gamma^{\Pi}/_{\beta}$ o resultado da substituição de todas as ocorrências da letra sentencial Π nos elementos de Γ por ocorrências da fórmula CS β. Então a seguinte propriedade vale para todo conjunto de fórmulas CS Γ e quaisquer fórmulas CS α, β:

se α é conseqüência tautológica de Γ, então $\alpha^{\Pi}/_{\beta}$ é conseqüência tautológica de $\Gamma^{\Pi}/_{\beta}$.

A justificação dessa propriedade é semelhante à justificação apresentada no caso das tautologias. Ela nos motiva, igualmente, a falar em esquemas da conseqüência tautológica.

Enunciamos, abaixo, algumas propriedades da relação de conseqüência tautológica (usaremos a notação $\Gamma\mathbb{T}\alpha$ para abreviar a sentença 'α é conseqüência tautológica de Γ').

(1) $\mathbb{T}\alpha$ se e só se α é uma tautologia;

(2) se $\alpha \in \Gamma$ então $\Gamma\mathbb{T}\alpha$;

(3) se $\Gamma\mathbb{T}\alpha$ e $\Gamma\subset\Delta$ então $\Delta\mathbb{T}\alpha$;

(4) $\Gamma\mathbb{T}(\alpha \wedge \neg\alpha)$ se e só se Γ é funcional-veritativamente inconsistente;

(5) $\{\alpha\}\mathbb{T}\beta$ se e só se $(\alpha \rightarrow \beta)$ é uma tautologia;

(6) $\{\alpha_1, ..., \alpha_n\}\mathbb{T}\beta$ se e só se $(((\alpha_1 \wedge ...) \wedge \alpha_n) \rightarrow \beta)$ é uma tautologia.

Em virtude da propriedade (6), podemos sempre decidir se uma fórmula β é conseqüência tautológica de um conjunto finito $\{\alpha_1, ..., \alpha_n\}$ de fórmulas. Para isso é suficiente tomar a conjunção generalizada dos elementos do conjunto como antecedente de uma condicional cujo consequente é β e testar esta fórmula numa tabela de verdade: se for tautológica, β é conseqüência tautológica de $\{\alpha_1, ..., \alpha_n\}$; se não for, cada linha em que a implicação é falsa corresponde a uma classe de valorações que são modelos de $\{\alpha_1, ..., \alpha_n\}$ que falsificam β.

Casos importantes de relação de conseqüência tautológica:

a) Para cada um dos esquemas $(\mathcal{E} \leftrightarrow \mathcal{E}')$ de equivalências tautológicas dadas na seção anterior temos: $\{\mathcal{E}\}\mathbb{T}\mathcal{E}'$ e $\{\mathcal{E}'\}\mathbb{T}\mathcal{E}$;
b) Para cada um dos esquemas $(\mathcal{E} \rightarrow \mathcal{E}')$ de implicações tautológicas dadas na seção anterior temos: $\{\mathcal{E}\}\mathbb{T}\mathcal{E}'$;
c) $(\mathcal{E}_1 \rightarrow (\mathcal{E}_2 \rightarrow ... \rightarrow (\mathcal{E}_n \rightarrow \mathcal{E}') ...))$ é um esquema tautológico se e só se $\{\mathcal{E}_1, \mathcal{E}_2, ..., \mathcal{E}_n\}\mathbb{T}\mathcal{E}'$.

Destaquemos alguns exemplos importantes:

$\{\alpha, (\alpha \rightarrow \beta)\}\mathbb{T}\beta$	Modus Ponens
$\{\neg\beta, (\alpha \rightarrow \beta)\mathbb{T}\neg\alpha$	Modus Tollens
$\{(\alpha \rightarrow \beta), (\beta \rightarrow \gamma)\}\mathbb{T}(\alpha \rightarrow \gamma)$	Silogismos Hipotético
$\{(\alpha \vee \beta), \neg\alpha\}\mathbb{T}\beta$	Silogismo Disjuntivo
$\{(\alpha \vee \beta), \neg\beta\}\mathbb{T}\alpha$	Silogismo Disjuntivo
$\{\alpha, \neg\alpha\}\mathbb{T}\beta$	Duns Scott
$(\alpha \rightarrow \gamma), (\beta \rightarrow \gamma, (\alpha \vee \beta)\}\mathbb{T}\gamma$	Prova por Casos

Observações:

Quando queremos saber se β é conseqüência tautológica de um conjunto finito de fórmulas $\{\alpha_1, ..., \alpha_n\}$, se não quisermos construir toda a tabela para $\{\alpha_1, ..., \alpha_n\} \cup \{\beta\}$, podemos proceder de uma das seguintes maneiras:

a) Considerar apenas **todas** as linhas que verificam todos os elementos de $\{\alpha_1, ..., \alpha_n\}$ e verificar se β é verdadeira em todas elas (caso em que β será conseqüência tautológica de $\{\alpha_1, ..., \alpha_n\}$);
b) Considerar apenas **todas** as linhas que falsificam β e verificar se alguma delas é modelo de $\{\alpha_1, ..., \alpha_n\}$ (caso em que β não é conseqüência tautológica $\{\alpha_1, ..., \alpha_n\}$) ou se nenhuma delas é modelo de $\{\alpha_1, ..., \alpha_n\}$ (caso em que β é conseqüência tautológica de $\{\alpha_1, ..., \alpha_n\}$).

1. Aplicação dos conceitos proposiciomais à linguagem $\mathcal{L}$

Vamos retornar à linguagem $\mathcal{L}$ do Cálculo de Predicados para aplicar os conceitos proposicionais que acabamos de estudar. Para tanto, seja Γ um conjunto de sentenças de $\mathcal{L}$, α uma sentença de $\mathcal{L}$(?) e h uma função de sentenças básicas (i.e. atômicas e gerais) em letras sentenciais. Então:

- α^h indicará o resultado da substituição de cada componente básico β, de α, por $h(\beta)$;
- Γ^h indicará o resultado da substituição de cada componente básico β, que ocorra nos elementos de Γ, por $h(\beta)$.

Assim, α^h é uma CS-associada de α, pela associação h e Γ^h num conjunto de CS-associadas dos elementos de Γ, pela associação h. Nessas condições:

1. dizemos que uma valoração ν é um modelo funcional-veritativo de Γ se para alguma função de sentenças básicas em letras sentenciais, h, ν é um modelo funcional-veritativo de Γ^h;
2. dizemos que Γ é funcional-veritativamente consistente se há pelo menos um modelo funcional-veritativo;
3. dizemos que α é tautológica se para algum h (como acima) α^h é tautológica;
4. dizemos que α é conseqüência tautológica de Γ se, para algum h (como acima), $\Gamma^h \mathbb{T} \alpha^h$.

Observemos que:

a) Para algum h, α^h é tautologia se e só se para toda h', $\alpha^{h'}$ é tautologia; e
b) Para algum h, α^h é conseqüência tautológica de Γ^h se e só se para todo h', $\alpha^{h'}$ é conseqüência tautológica de $\Gamma^{h'}$.

Assim, se quisermos saber se α é uma tautologia, encontramos uma CS-associada de α e testamos essa sentença CS numa tabela de verdade. O resultado obtido é aplicável a α. Da mesma forma, se quisermos saber se um conjunto finito $\{\alpha_1, ..., \alpha_n\}$ tem como conseqüência tautológica β, encontramos um conjunto de CS-associadas de $\alpha_1, ..., \alpha_n, \beta$ (segundo uma mesma associação) e testamos se a associada de β é conseqüência tautológica das associadas de $\{\alpha_1, ..., \alpha_n\}$.

Dessa forma,

$\{(\forall x F^1 x \to \neg G^1 a), \forall x F^1 x\} \mathbb{T} \neg G^1 a$ pois $\{P, (P \to \neg Q)\} \mathbb{T} \neg Q$

'$(\forall x F^1 x \vee \neg \forall x F^1 x)$' é uma tautologia pois '$(P \vee \neg P)$' é uma tautologia.

Da mesma forma,

'$\{(\exists x F^1 x \to \neg(F^1 a \vee G^1 b)), \neg F^1 a, (G^1 b \to \forall x F^1 x)\}$' é funcional-veritativamente consistente, pois $\{(P \to \neg(Q \vee R)), \neg Q, (R \to S)\}$ é funcional-veritativamente consistente, uma vez que toda valoração booleana ν tal que $\nu(P) = \nu(Q) = F$ e $\nu(R) = \nu(S) = V$, por exemplo, verifica '$(P \to \neg(Q \vee R))$', '$\neg Q$' e '$(R \to S)$'.