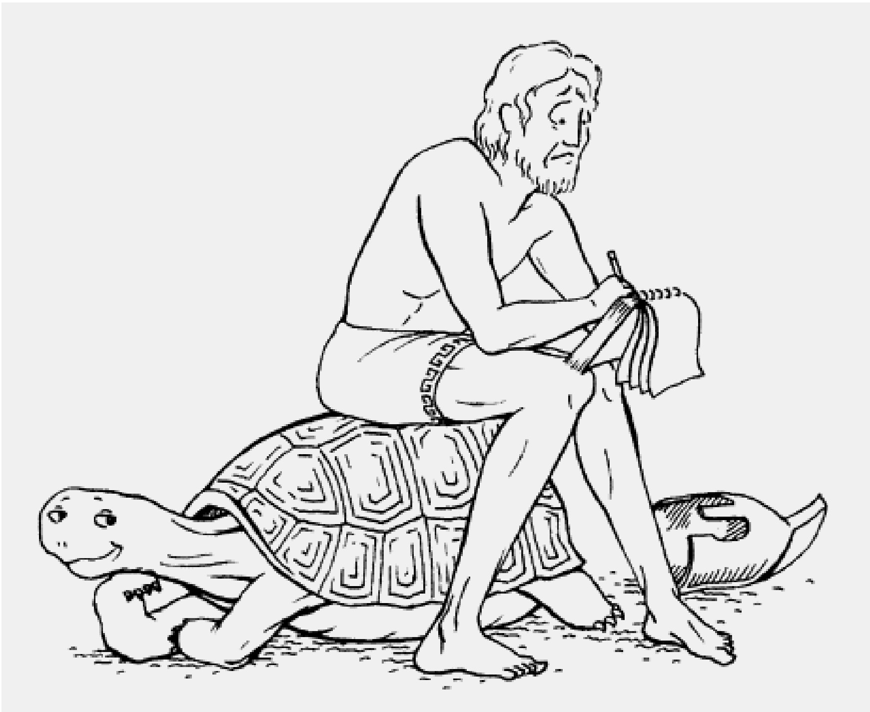
UNIVERSIDADE FEDERAL DO PIAUÍ
UNIVERSIDADE ABERTA DO PIAUÍ
Programa de Educação a Distância



LÓGICA PARA A COMPUTAÇÃO

Francisco Vieira de Souza

PRESIDENTE DA REPÚBLICA
Luiz Inácio Lula da Silva

MINISTRO DA EDUCAÇÃO

Fernando Haddad

UNIVERSIDADE FEDERAL DO PIAUÍ-REITOR
Luiz de Sousa Santos Júnior

SECRETÁRIO DE EDUCAÇÃO A DISTÂNCIA DO MEC
Carlos Eduardo Bielschowsky

DIRETOR DE POLITICAS PUBLICAS PARA EAD
Hélio Chaves

UNIVERSIDADE ABERTA DO BRASIL-COORDENADOR GERAL
Celso Costa

CENTRO DE EDUCAÇÃO ABERTA A DISTÂNCIA DA UFPI
Coordenador Geral de EaD na UFPI
Gildásio Guedes Fernandes

CENTRO DE CIENCIAS DA NATUREZA
Helder Nunes da Cunha

COORDENADOR DO CURSO de Sistema de Informação na Modaliade de EaD
Luiz Cláudio Demes da Mata Sousa

DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA-CHEFE DO DEPARTAMENTO

Paulo Sérgio Marques dos Santos

EQUIPE DE APOIO

Profº. Arlino Araújo

Liana Cardoso

Luana Monteiro

Cleidinalva Oliveira

Lana Grasiela Marques

DIAGRAMAÇÃO

Samuel Falcão Silva

S729l Souza, Francisco Vieira de
Lógica para a Computacional/Francisco Vieira de Souza. – Teresina: UFPI/UAPI. 2008.

152p.

Inclui bibliografia

1 – Lógica Proposicional. 2 – Álgebra Booleana. 3 – Lógica de Predicados. I. Universidade Federal do Piauí/Universidade Aberta do Piauí. II. Título.

CDD: 511.3

APRESENTAÇÃO

Este texto é destinado aos estudantes do programa de Educação a Distância da Universidade Aberta do Piauí (UAPI) vinculada ao consórcio formado pela Universidade Federal do Piauí (UFPI), Universidade Estadual do Piauí (UESPI), Centro Federal de Ensino Tecnológico do Piauí (CEFET-PI), com apoio do Governo do estado do Piauí, através da Secretaria de Educação. O texto é composto de três unidades, contendo itens e subitens que discorrem sobre a Lógica Proposicional e a Lógica de Predicados, evidenciando como estas estruturas podem ser utilizadas no estudo da Informática.

Na **Unidade 1**, é analisada a Lógica Proposicional e as suas principais estruturas, abordando as sentenças e diversas formas de construção, buscando encontrar formas e metodologias de provas da validade ou falsidade de argumentos.

Na **Unidade 2** são feitas comparações com teorias conhecidas como a Teoria dos Conjuntos e a Álgebra de George Boole, além de ser feita uma justificativa sobre o princípio da indução finita.

A **Unidade 3** é dedicada ao estudo da Lógica de Predicados como forma alternativa para a construção de expressões cujos significados não podem ser capturados pelos construtores da Lógica Proposicional. Na unidade também é apresentado o problema da indecibilidade do Cálculo de Predicados de Primeira Ordem, fazendo alusão à ligação existente entre esta teoria e a conhecida “Tese de Church”.

Sumário Geral

UNIDADE 01 - LÓGICA PROPOSICIONAL

UNIDADE 02 - A LÓGICA E OUTRAS TEORIAS

UNIDADE 03 - LÓGICA DE PREDICADOS

Unidade 1

Lógica Proposicional

RESUMO

O objetivo principal desta unidade é apresentar os principais conceitos e estruturas da Lógica Proposicional bem como ela pode ser utilizada no ordenamento do raciocínio humano, na busca de soluções para os problemas que ocorrem na natureza.Na unidade é mostrada a evolução histórica desde a sua utilização apenas como formulação correta de argumentos, utilizada apenas pelas Ciências Sociais, até o seu emprego atual na Ciência da Computação. A unidade também contém vários exemplos, e exercícios resolvidos tentando proporcionar ao leitor o entendimento pleno dos conceitos envolvidos, além de serem propostos vários exercícios para sedimentar a teoria apresentada. A forma de apresentação utilizada é de acordo com o exigido para o ensino à distância, ou seja, tendo em vista sempre esta nova modalidade de ensino.

Sumário Unidade

LÓGICA PROPOSICIONAL

1.1 Introdução

A Lógica teve seu início com o grego Aristóteles (384-322 a. C.) quando outros filósofos, também gregos, passaram a utilizar seus enunciados resultando em grande simplificação e clareza para a Matemática.

Aristóteles

Por volta de 1666, Gottfried Wilhelm Leibniz (1646-1716) usou em vários trabalhos o que chamou de *Calculus Rationator*, ou *Lógica Matemática* ou ainda *Logística*. Estas idéias nunca foram teorizadas por Leibniz, porém seus escritos trouxeram a idéia da Lógica Matemática.

Leonhard Euler.

No século XVIII, Leonhard Euler (1707-1783) introduziu a representação gráfica das relações entre sentenças ou proposições, mais tarde ampliada por John Venn (1834 - 1923), E. W. Veitch em 1952 e M. Karnaugh em 1953. Em 1847, Augustus DeMorgan (1806-1871) publicou um tratado *Formal Logic* envolvendo-se em uma discussão pública com o filósofo escocês William Hamilton, conhecido por sua aversão à Matemática, que escreveu "*A Matemática congela e embota a mente; um excessivo estudo da Matemática incapacita a mente para as energias que a filosofia e a vida requerem.*" George Boole (1815-1864), ligado a DeMorgan, tomou as dores do amigo e escreveu "*The Mathematical Analysis of Logic*" em 1848. Em 1854 ele escreveu o livro "*An Investigation of the Laws of Thoutght*" e em 1859 escreveu "*Treatise on Defferantial Equations*" no qual discutiu o método simbólico geral.

O trabalho de George Boole foi amplliado por Lewis Carroll em 1896, por Whitehead em 1898, por Huntington em 1904 e em 1933, por Sheffer em 1913, entre outros. Este período de desenvolvimento da Lógica culminou com a publicação do “*Principia Mathematica”* por Alfred North-Whitehead (1861-1947) e por Bertand Russel (1872-1970), que representou uma grande ajuda para completar o programa sugerido por Leibniz, que visava dar uma base lógica para toda a Matemática.

1.2 Primeiros passos

Para facilitar o nosso entendimento sobre a Lógica, vamos analisar de que forma ela é importante e que papel ela desempenha no ordenamento do raciocínio humano, na busca de soluções para os problemas que ocorrem na natureza.

Antigamente, a Lógica era utilizada apenas como forma de ordenamento de argumentos, conhecidos como premissas, para se chegar a uma conclusão que representasse o resultado de alguma questão. Desta forma, a Lógica era tão somente uma técnica utilizada principalmente pela Filosofia e pelas ciências humanas.

Atualmente, o computador representa uma máquina capaz de processar muitos cálculos de forma correta e a grande velocidade. Desta forma, eles passaram a ser utilizados na busca de soluções para os mais diversos problemas da natureza. Estes problemas para serem processados pelos computadores devem ser tratados de forma adequada para que as soluções encontradas sejam representativas.

Analisemos uma situação clássica, em que dois amigos, Francisco e Jorge, se encontram em uma sexta-feira à noite em um bar para jogar conversa fora, acompanhados de *chopp* gelado e peixe frito. Francisco argumenta euforicamente que “*Kaká é o melhor jogador de futebol do mundo*”, instante em que

Jorge refuta, sob o calor da “*água que passarinho não bebe*”, que “*o melhor jogador de futebol do mundo é, sem qualquer sombra de dúvidas, Ronaldinho Gaúcho*”. Deve-se facilmente depreender que esta discussão deve ter continuado até a madrugada, principalmente porque chegou à mesa um argentino chamado Celso.

Por outro lado, imaginemos agora a afirmação: “*5 é um número natural primo*”. Esta proposição não desperta qualquer dúvida, uma vez que sabemos o que significa um número ser primo e sabemos verificar, de forma inequívoca, que o número 5 obedece às exigências para que um número seja primo.

Analisando estas expressões, verifica-se que é possível decidir de forma irrefutável se o número 5 é primo mas não podemos decidir se o gol feito por um jogador é mais bonito que o feito por outro, porque esta decisão varia de pessoa para pessoa.

Com estes dois exemplos, pretendemos caracterizar a necessidade de um rigor na linguagem natural corrente. Isto nos leva a uma linguagem matemática, estudando os princípios da chamada “Lógica Matemática”.

1.3 Construção de sentenças

As expressões analisadas na seção anterior, tanto na linguagem matemática quanto na linguagem corrente, envolvem certas entidades, cada uma delas representadas convenientemente por símbolos. Na oração “*Kaká é o melhor jogador de futebol do mundo*” aparecem as entidades “*Kaká*”, “*jogador de futebol*”, “*melhor jogador de futebo*l” e “*melhor jogador de futebol do mundo*”. Não se deve confundir uma entidade com a sua representação, uma vez que uma mesma entidade pode ter mais de uma representação. por exemplo, "*3*" e "*raiz quadrada de nove*" representam a mesma entidade.

As sentenças são expressões da linguagem às quais pode-se atribuir um valor verdadeiro ou falso. Por exemplo, "*Brasília é a capital do Brasil*" é uma sentença verdadeira, enquanto "*5 é um número par*" é uma sentença falsa. Normalmente, o valor verdadeiro é simbolizado pela letra maiúscula ***V***, enquanto o valor falso é simbolizado pela letra maiúscula ***F***. Num primeiro estudo da Lógica Matemática será admitido que toda proposição tenha apenas dois valores possíveis: ***V*** ou ***F***. Esta idéia deve ser abandonada ao estudarmos outras teorias onde as sentenças possam ter valores distintos de ***F*** e ***V***. As sentenças, por sua vez, podem ser representadas pelas letras minúsculas *p, q, r,* etc. Desta forma, podemos definir uma sentença da seguinte forma:

Definição 1.1. Uma *sentença* ou *proposição* é uma expressão de uma dada linguagem, que pode ser classificada como verdadeira ou falsa, de maneira exclusiva, em um contexto.

Exemplo 1.1. As expressões a seguir são sentenças.

a) Teresina é uma cidade quente.

b) A Amazônia é um deserto.

c) O Papa é romano ou não.

d) Barack Obama é e não é americano.

e) Se algo é igual a si próprio e tudo é igual a si próprio, então Sócrates e Obama são iguais.

Exemplo 1.2. As sentenças imperativas, interrogativas ou exclamativas não serão consideradas em nosso estudo. As expressões a seguir não são exemplos de sentenças:

a) Estude para a prova de Lógica!

b) Qual foi a sua nota na prova de Lógica?

c) Que saudade de Amélia!

As expressões do ***Exemplo 1.1*** são sentenças porque pode-se atribuir a cada uma delas um valor verdadeiro ou falso, o que não é possível com as expressões do ***Exemplo 1.2*** porque elas representam uma frase imperativa, uma interrogativa e uma exclamativa, respectivamente.

1.4 Conectivos lógicos

Uma característica importante das sentenças é que elas podem ser combinadas para construir outras sentenças, envolvendo expressões como **e, ou, se .. então, se e somente se**, que são aplicadas a duas sentenças. As sentenças do ***Exemplo 1.1*** podem ser combinadas para formar as seguintes sentenças, além de outras:

a) Barack Obama é e não é americano e o Papa é romano ou não..
b) A Amazônia é um deserto mas Barack Obama não é americano.
c) Se o Papa é romano então Teresina é uma cidade quente.

Também é possível construir sentenças aplicando as expressões **não** e **é possível que.** Neste caso elas são aplicadas a uma única sentença, diferente da forma como se constroem sentenças utilizando outras expressões que necessitam de duas sentenças. Como exemplo, podemos construir as sentenças:

a) A Amazônia não é um deserto

b) É possível que o Papa seja romano

Embora as expressões **não, e, ou, se .. então, se e somente se** e **é possível que** não possuam a mesma classificação gramatical, do ponto de vista da Lógica, todas elas possuem a mesma função que é construir sentenças a partir de uma ou mais sentenças previamente dadas. No estudo da Lógica, estas expressões têm uma denominação especial.

Definição 1.2. Um conectivo é uma expressão de uma dada linguagem, utilizada para formar sentenças a partir de sentenças dadas.

Exemplo 1.3. Dadas as sentenças "*5 é ímpar*" e "*4 < 2*", podemos formar as seguintes sentenças:

a) 5 é ímpar ***e*** *4 < 2.*

b) 5 é ímpar ***ou*** *4 < 2.*

c) ***Se*** *5 é ímpar,* ***então*** *4 < 2.*

d) 5 é ímpar ***se, e somente se*** *4 < 2*

e) *5* ***não*** *é ímpar*.

Uma forma interessante de se analisar os conectivos é considerá-los como as operações aritméticas da Matemática. Uma operação aritmética é uma forma de combinar elementos para formar novos elementos. Por exemplo, a operação de multiplicação processa os números 3 e 4 e associa o resultado deste processamento ao valor 12. No caso dos conectivos, os elementos a serem operados são sentenças e o resultado obtido é uma nova sentença.

Peso de um conectivo

Embora existam muitos conectivos que podem ser utilizados na construção de sentenças, só nos interessam para o estudo da Lógica um pequeno subconjunto deles. Nosso objetivo é estudar certos aspectos lógicos da atividade matemática. Desta forma, serão considerados apenas os conectivos ***não, e, ou, se...então*** e ***se, e somente se***.

Podemos verificar que, na construção da sentença "*5 não é ímpar*", o conectivo ***não*** foi aplicado a uma única sentença, mas para construir a sentença "*5 é ímpar* ***e*** *4 < 2*" o conectivo ***e*** foi aplicado a duas sentenças. Os conectivos são classificados de acordo com o número de sentenças que eles precisam para formar uma nova sentença.

Definição 1.3. O peso ou *aridade* de um conectivo é o número exato de sentenças utilizadas para formar uma nova sentença, por meio deste conectivo.

A Tabela 1.1 mostra os pesos dos principais conectivos. **(prox. Página)**

Tabela 1.1 - Os conectivos e seus pesos.

CONECTIVO	PESO (ARIDADE)
Não	**1**
E	**2**
Ou	**2**
se ... então	**2**
se, e somente se	**2**

1.5 Sentenças atômicas e sentenças moleculares

As sentenças podem ser classificadas de acordo com o fato de terem sido, ou não, obtidas de outras sentenças utilizando conectivos.

Definição 1.4. Uma sentença é chamada *atômica* se ela não tem qualquer conectivo.

A partir desta definição, verifica-se que as expressões negativas de uma linguagem não são sentenças atômicas. As sentenças atômicas são consideradas as sentenças básicas, ou seja, a partir delas são construídas outras sentenças.

Exemplo 1*.4.* As sentenças seguintes são atômicas:

a) 5 é um número primo.

b) *O rio Parnaíba é perene.*

c) *João Pessoa é a capital da Paraíba*.

d) *A terra é um planeta*.

Definição 1.5. Uma sentença é molecular se ela não for atômica, ou seja, se tiver pelo menos um conectivo.

Exemplo 1.5. As sentenças a seguir são moleculares:

a) *5 não é um número primo*.

b) João Pessoa é a capital da Paraíba e a terra é um planeta.

c) O sol é uma estrela ou a terra é um planeta.

d) Se a terra é um planeta então o sol é uma estrela.

Neste texto, as letras romanas minúsculas representarão as sentenças atômicas, por exemplo, *p, q, r*, etc. As letras gregas minúsculas serão usadas para representar tanto as sentenças atômicas quanto as sentenças moleculares. As letras gregas minúsculas serão utilizadas quando se deseja atribuir um grau de generalidade, ou seja, representar uma sentença que pode ser atômica ou molecular

Classificação das sentenças moleculares

As sentenças moleculares podem ser classificadas de acordo com o conectivo utilizado em sua construção. Nesta seção, serão mostradas diversas definições que fundamentam a classificação das sentenças moleculares.

Definição 1.6. Uma sentença é uma *negação* se ela for obtida a partir de outra utilizando o conectivo ***não***.

Exemplo 1.6. A sentença "*5 não é um número primo*" pode ser considerada uma negação obtida pela aplicação do conectivo ***não*** à sentença "*5 é um número primo*".

Definição 1.7. A sentença utilizada na construção de uma negação é chamada de *sentença negada* ou *componente* da negação.

No caso do *Exemplo 1.6*, a sentença “*5 é um número primo*” é a sentença negada ou a componente da negação “*5 não é primo*”.

Definição 1.8. Uma sentença é uma *conjunção* se ela for obtida a partir de duas outras sentenças através do conectivo ***e***.

Exemplo 1.7. A sentença “*4 não divide 7 nem 9*” deve ser re-escrita como “*4 não divide 7 e 4 não divide 9*” para que se possa entender o que realmente ela quer significar. Neste caso, a sentença é a conjunção das sentenças “*4 não divide 7*” e “*4 não divide 9*” aplicando o conectivo ***e***. Por sua vez, a sentença “*4 não divide 7*” é a negação da sentença “*4 divide 7*” e a sentença “*4 não divide 9*” é a negação da sentença “*4 divide 9*”. O leitor deve observar que neste caso, foi necessário reescrever a sentença original de forma que ela pudesse ser analisada corretamente. Este tema será abordado brevemente.

O leitor atento deve ter observado que o conectivo ***e*** foi aplicado a duas sentenças, uma vez que ele é um conectivo de peso 2. A primeira sentença é chamada *primeira componente* e a segunda sentença é chamada *segunda componente* da conjunção.

Voltando ao *Exemplo 1.7,* a sentença “*4 não divide 7*” é a *primeira componente* e a sentença “*4 não divide 9*” é a *segunda componente* da conjunção.

Definição 1.9. Uma sentença é uma *disjunção* se ela for obtida a partir de duas outras sentenças, através do conectivo ***ou***.

Exemplo 1.8. A sentença "*5 é ímpar ou 13 é par*" é obtida a partir das sentenças "*5 é ímpar*" e "*13 é par*" aplicando-se o conectivo ***ou***.

Como observado para o conectivo ***e***, que tem peso 2 e forma conjunções, o conectivo ***ou*** também tem peso 2 e por isso a disjunção também precisa de duas sentenças para que ela seja formada, conforme pode ser observado no ***Exemplo 1.8***, onde a primeira sentença é denominada *primeira componente* e a segunda sentença é chamada *segunda componente* da disjunção.

Definição 1.10. Uma sentença é uma *implicação* se ela for obtida a partir de duas outras, através do conectivo ***se...então***.

Exemplo 1.9**.** A sentença "*Célia viajou para Campina Grande se Tarcísio comprou um carro novo*" deve ser reescrita como "*se Tarcísio comprou um carro novo, então Célia viajou para Campina Grande*". Esta sentença é uma implicação formada pelas sentenças "*Tarcísio comprou um carro novo*" e "*Célia viajou para Campina Grande*" utilizando o conectivo **se...então**.

As duas (o conectivo ***se...então*** tem peso 2) sentenças são chamadas de *componentes* da implicação, onde a sentença "*Tarcísio comprou um carro novo*" é denominada *antecedente* e a sentença "*Célia viajou para Campina Grande*" é denominada *conseqüente* da implicação.

Definição 1.11. Uma sentença é uma *biimplicação* se ela for obtida a partir de duas outras sentenças, através do conectivo ***se, e somente se***.

Exemplo 1*.10*. A sentença “Vou a Parnaíba se, e somente se, a estrada foi consertada” é a biimplicaçào das sentenças “Vou a Parnaíba” e “a estrada foi consertada”, aplicando-se o conectivo **se, e somente se**.

Estas duas sentenças são chamadas de *componentes* da biimplicação, sendo “*Vou a Parnaíba*” a *primeira componente* e “*a estrada foi consertada*” a *segunda componente*.

1.6 Reescrita de sentenças

Em alguns dos exemplos mostrados anteriormente foi verificado que para se analisar corretamente a sentença, foi necessário reescrevê-la para que a análise pudesse ser feita corretamente. Serão analisados, a seguir, alguns exemplos onde esta técnica se torna necessária.

Exemplo 1.11. Seja a sentença: “*5 não é ímpar*”. Um número natural que não é ímpar é obrigatoriamente um número par. Assim, a sentença acima pode ser entendida como “*5 é par*”. Neste caso, a sentença não apresenta qualquer conectivo e, portanto, ela é uma sentença atômica. Por outro lado, podemos analisar a sentença como sendo a negação da sentença “*5 é ímpar*”. Neste caso, ela é uma sentença molecular.

Exemplo 1.12. Seja a sentença “*Emiliano e Chagas são casados*”. Qual é a intenção da sentença? Não está claro que o objetivo desta sentença seja informar que Emiliano tem uma esposa e Chagas tem outra. Da forma como a sentença está escrita, pode-se também deduzir que Emiliano e Chagas são casados um com o outro.

Estes exemplos mostram que algumas sentenças podem ter mais de uma interpretação. Neste caso, diz-se que elas são ambíguas e isto pode acarretar análises lógicas incompatíveis. É necessário que a formação de sentenças obedeça regras precisas para que estas ambigüidades não aconteçam.

Regras de reescrita

Existem algumas regras básicas que devem ser seguidas na construção de sentenças. Em uma primeira etapa, deve-se explicitar as sentenças atômicas que compõem a sentença final, atribuindo a elas um símbolo, por exemplo, as letras romanas minúsculas, p, q, r, s, t, etc, transformando-as em sentenças simbolizadas. Em uma etapa posterior, devem ser atribuídos símbolos aos conectivos. Os conectivos são representados pelos símbolos mostrados na Tabela 1.4. A seguir será mostrado um conjunto de regras que devem ser utilizadas nestas reescritas.

Tabela 1.2 - Reescritas de algumas sentenças atômicas.

SENTENÇA ATÔMICA	SENTENÇA ATÔMICA REESCRITA
5 é um número ímpar	***(5 é um número ímpar)***
10 é maior que 100	***(10 é maior que 100)***
Thaís é psicóloga	***(Thaís é psicóloga)***
Mariane é uma boa aluna	***(Mariane é uma boa aluna)***

Regra 1. *Uma sentença atômica p deve ser reescrita entre parênteses, ou seja, (p).*

***Exemplo 1.13*.** Para facilitar a compreensão, algumas sentenças atômicas estão mostradas na primeira coluna da Tabela 1.2 e suas reescritas estão mostradas na segunda coluna.

***Regra 2**. A negação de uma sentença p deve ser reescrita como (¬(p)), onde p é a sentença negada.*

Exemplo 1.14. Utilizando as mesmas sentenças mostradas na Tabela 1.2 foi construída a Tabela 1.3 que mostra cada uma destas sentenças reescritas em suas formas negadas. Deve ser observado que, apesar do exemplo mostrar apenas a negação de sentenças atômicas, ela pode ser aplicada também a sentenças moleculares.

Tabela 1.3 - Negação de sentenças.

SENTENÇA	NEGAÇÃO REESCRITA
5 é ímpar	**(¬ (5 é ímpar))**
10 é maior que 100	**(¬(10 é maior que 100))**
Thaís é psicóloga	**(¬(Thaís é psicóloga))**
Mariane é uma boa aluna	**(¬(Mariane é uma boa aluna))**

***Regra 3**. Uma conjunção de duas sentenças p e q, previamente reescritas, deve ser reescrita como ((p) Λ (q)).*

Exemplo 1**.15.* A conjunção *“5 não é ímpar, nem é maior que 0”* deve ser entendida como a conjunção *“5 não é ímpar* ***e *5 não é maior que 0”.* Neste caso, ela deve ser reescrita como *((¬(5 é ímpar)) Λ (¬(5 é maior que 0))).*

***Exemplo 1.16**.* Já sabemos que a conjunção “*Nathalie e Natasha foram à Barras*” é ambígua, uma vez que pode ser interpretada como “*Nathalie foi a Barras e Natasha foi a Barras*” ou como “*Nathalie e Natasha foram a Barras, juntas*”. Esta ambigüidade de significado provoca também ambigüidade de construção porque não é possível decidir se esta sentença é atômica (na segunda interpretação) ou se ela é molecular, como a primeira interpretação. Como solução, foi feita uma convenção de que onde ocorra este tipo de problema, a sentença deve ser considerada em sua forma molecular. Desta forma, a sentença acima deve ser reescrita por *(Nathalie foi a Barras* ∧ *Natasha foi a Barras)*.

Regra 4*. Uma disjunção de duas sentenças p e q, previamente reescritas, deve ser reescrita como ((p)* ∨ *(q)).*

Exemplo 1.17*.* A disjunção *"20 não é maior que 10 ou, 4 é primo e 1 é maior que 4"* deve ser reescrita como *((¬(20 é maior que 10))* ∨ *((4 é primo) ∧ (1 é maior que 4))).*

Regra 5*. Uma implicação de duas sentenças p e q, sendo p a sentença antecedente e q a sentença conseqüente, previamente reescritas, deve ser reescrita como ((p) →(q)).*

Exemplo 1.18*.* A sentença *"Às sextas-feiras Antônio vai ao bar da Miúda"* pode ser entendida como *"se for sexta-feira, então Antônio vai ao bar da Miúda"* que deve ser reescrita por *((é sexta-feira) →(Antônio vai ao bar da Miúda)).*

Regra 6*.* Uma *biimplicação* de duas sentenças componentes *p* e *q,* previamente reescritas, deve ser reescrita como *((p) ↔(q)).*

Exemplo 1.19*.* A biimplicação *"Zefinha emagrecerá se, e somente se não beber refrigerante nem comer macarrão"* deve ser reescrita como *((Zefinha emagrece) ↔ ((¬(Zefinha bebe refrigerante)) ∧ (¬(Zefinha come macarrão)))).*

Sempre que possível, as sentenças devem ser reescritas no presente do indicativo, ou seja, não deve ser levado em conta o tempo verbal.

1.7 Simbologia das sentenças

Deve ser levado em conta que nosso objetivo é determinar se uma determinada sentença *p* é, ou não, uma verdade lógica. Para isto, deve-se verificar se *p* é verdadeira, ou não, em todos os contextos. O fato de uma sentença ser verdadeira em todos os contextos depende da forma como ela foi construída e não de seu conteúdo. As regras de reescritas permitem explicitar como as sentenças são formadas, mas é necessário analisar como as sentenças são simbolizadas. Este é o passo final da reescrita porque permite “esconder” o conteúdo e explicitar a forma. A Tabela 1.4, a seguir, mostra um resumo dos conectivos e seus respectivos símbolos.

Tabela 1.4 - Os conectivos e seus símbolos.

CONECTIVO	SÍMBOLO
não	$\neg$
e	$\wedge$
ou	$\vee$
se...então	$\rightarrow$
se, e somente se	$\leftrightarrow$

Sejam as sentenças:

a) *5 é ímpar ou 5 não é ímpar.*

b) *Einstein é brasileiro ou Einstein não é brasileiro*.

c) *O conjunto dos números perfeitos possui um maior elemento ou o conjunto dos números perfeitos não possui um maior elemento.*

Pode-se observar, sem qualquer dificuldade, que todas estas sentenças são disjunções e, de acordo com o que foi visto anteriormente, devem ser reescritas da seguinte forma:

a) *((5 é ímpar)* V *(¬(5 é ímpar)))*.

b) ((Einstein é brasileiro) V *(¬(Einstein é brasileiro))).*

c) ((O conjunto dos números perfeitos tem um maior elemento) V *(¬(O conjunto dos números perfeitos tem um maior elemento))).*

Vamos examinar cada uma destas sentenças em separado.

a) A sentença "*((5 é ímpar)* V *(¬(5 é ímpar)))"* é a disjunção entre a sentença atômica "*5 é ímpar*" e a sentença "*(¬(5 é ímpar))"* . Esta, por sua vez, é a negação da sentença "*5 é ímpar*". Como a sentença apresenta duas alternativas, das quais a primeira é verdadeira, a sentença representada pela disjunção, também é verdadeira.

b) A sentença "*((Einstein é brasileiro)* V *(¬(Einstein é brasileiro)))*" também é uma disjunção entre a sentença atômica "*(Einstein é brasileiro)*" e a sentença "*(¬(Einstein é brasileiro))"* . Esta, por sua vez, é a negação da sentença *"(Einstein é brasileiro)*"*.* Neste caso, a sentença também apresenta duas alternativas, das quais a segunda é verdadeira. Logo a sentença representada pela disjunção também é verdadeira.

c) A sentença "((O conjunto dos números perfeitos tem um maior elemento) V (¬(O conjunto dos números perfeitos tem um maior elemento)))" também é a disjunção entre a sentença atômica "(O conjunto dos números perfeitos tem um maior elemento)" e a sentença "(¬(O conjunto dos números perfeitos tem um maior elemento))". Esta, por sua vez é a negação da sentença "(O conjunto dos números perfeitos tem um maior elemento)". Neste caso, não sabemos se a primeira ou a segunda sentença da disjunção é verdadeira, porque este é um problema matemático ainda em aberto. No entanto, a sentença apresenta duas alternativas excludentes, mas complementares, ou seja, se a primeira for verdadeira a segunda é falsa e vice-versa. Assim, pode-se afirmar, com certeza, que a sentença representada pela disjunção é verdadeira.

Pode ser facilmente observado que a explicação dada na sentença da letra c) anterior também pode ser aplicada às sentenças das letras a) e b). Isto é verdade porque as sentenças componentes são expressões alternativas excludentes e complementares. Dito de outra forma, se as sentenças de uma disjunção expressam alternativas excludentes e complementares, a disjunção é verdadeira, independente do conhecimento antecipado sobre a verdade ou falsidade das componentes.

Dada uma sentença simbolizada *p*, nosso objetivo é determinar se *p* é, ou não, uma verdade lógica. Isto significa verificar se *p* é verdadeira, ou não, em todos os contextos. Pelo exposto, o fato de uma sentença ser verdadeira em todos os contextos depende da maneira como a sentença foi formada e não de seu conteúdo. As regras de reescritas nos permitem verificar como as sentenças foram formadas, faltando apenas uma forma de simbolizar as sentenças, permitindo esconder seus conteúdos, uma vez que eles não são importantes.

Esta tarefa não é tão simples em um primeiro momento. É necessária alguma prática para que seja feita de forma adequada. Para isto, será introduzida uma metodologia a ser seguida, pelo menos enquanto não adquiramos uma prática consolidada nesta área. Esta metodologia consiste da divisão das sentenças em passos. São eles:

1. Classificar a sentença como atômica ou molecular.
2. Classificar todos os conectivos que ocorrem na sentença (se for molecular).
3. Classificar o tipo da sentença em negação, conjunção, disjunção, implicação ou biimplicação (se for molecular).
4. Reescrever a sentença de acordo com as regras de reescritas.
5. Simbolizar a sentença reescrita, substituindo as sentenças atômicas pelas letras *p, q, r* ou *s* (indexadas ou não), de modo que cada ocorrência de uma mesma sentença seja substituída sempre pela mesma letra e que sentenças atômicas distintas sejam substituídas por letras também distintas.

Será apresentada a seguir uma seqüência de exemplos de aplicação desta metodologia, para que o leitor possa fixá-la com facilidade.

Exemplo 1.20.

a) "*Francisco é feliz*".

Aplicando a metodologia proposta para a simbolização, temos:

1) Atômica.
2) Não tem conectivos por ser atômica.
3) Não pode ser classificada por ser atômica.
4) (*Francisco é feliz*).

5) A sentença pode ser simbolizada por *p*, onde *p: Francisco é feliz*.

b) “*Francisco é feliz e Cecília o ama*”.

A sentença deve ser reescrita como “*Francisco é feliz e Cecília ama Francisco*”*.* Aplicando a metodologia proposta, temos:

1) Molecular.
2) Possui o conectivo ***e***.
3) Trata-se de uma conjunção.
4) *((Francisco é feliz) Λ (Cecília ama Francisco))*.
5) Sendo *p : Francisco é feliz* e *q : Cecília ama Francisco*, então: (*(p) ˄ (q))*.

c) “*Francisco é feliz caso Cecília o ame*”.

A sentença deve ser reescrita como “***Se*** *Cecília ama Francisco**,* **então*** *Francisco é feliz*”. Aplicando a metodologia proposta, temos:

1) Molecular.
2) Possui o conectivo ***se ... então***.
3) Trata-se de uma implicação.
4) *((Cecília ama Francisco) → (Francisco é feliz))*.
5) Sendo *p : Cecília ama Francisco* e *q : Francisco é feliz*, então: (*(p) →(q))*.

d) “*Francisco é feliz pois Cecília o ama*”.

A sentença deve ser reescrita como “*Cecília ama Francisco,* ***e se*** *Cecília ama Francisco,* ***então*** *Francisco é feliz*”. Aplicando a metodologia proposta, temos:

1) Molecular.
2) Possui os conectivos ***e*** e ***se ... então***.
3) Trata-se de uma conjunção onde a segunda componente é uma implicação.

4) *(Cecília ama Francisco* ˄ *((Cecília ama Francisco)* → *(Francisco é feliz)))*.
5) Sendo *p : Cecília ama Francisco* e *q : Francisco é feliz*, então a sentença deve ser representada por *((p* ˄ *((p)* → *(q)))*.

e) “*Francisco é feliz porque Cecília o ama e ela é feliz*”.

A sentença deve ser reescrita como “*Cecília ama Francisco e Cecília é feliz, e se Cecília ama Francisco e Cecília é feliz. então Francisco é feliz*”. Aplicando a metodologia proposta, temos:

1) Molecular.
2) Possui os conectivos ***e*** e ***se ... então***.
3) Trata-se de uma conjunção onde a primeira componente é uma conjunção e a segunda componente é uma implicação.
4) *(((Cecília ama Francisco)* ∧ *(Cecília é feliz))* ∧ *(((Cecília ama Francisco)* ∧ *(Cecília é feliz))* → *(Francisco é feliz)))*.
5) Sendo *p : Cecília ama Francisco, q : Cecília é feliz* e *r : Francisco é feliz*, então a sentença deve ser representada por *(((p)* ∧ *(q))* ∧ *(((p)* ∧ *(q))* → *(r)))*.

Nos exemplos a seguir, não serão mais mostrados todos os passos definidos na metodologia apresentada, uma vez que, neste ponto, já se supõe que o leitor tenha adquirido alguma prática, o que torna desnecessária a colocação de todos estes passos. Neste caso, serão mostrados apenas os resultados finais.

a) “10 + 10 ≠ 20”.

Sendo *p : 10 + 10 = 20*, então a sentença deve ser simbolizada por *(¬(p))*.

b) “3 e 5 são ímpares”.

Sendo *p : 3 é ímpar* e *q : 5 é ímpar*, então a sentença deve ser simbolizada por *((p) Λ (q))*.

c) “Pelo menos um dos números inteiros 2, 5 e 7 primo”.

Sendo *p : 2 é um número primo, q : 5 é um número primo* e *r : 7 é um número primo*, então a sentença deve ser simbolizada por *(((p)* V *((q)* V *®)))* ou por *((((p)* V *(q))* V *(r)))*.

d) “Exatamente um dos números 1, 2 e 3 é primo”.

Sendo p : 1 é primo, q : 2 é primo e r : 3 é primo, então a sentença deve ser simbolizada por (((p) Λ (¬(q) Λ ¬(r))) V ((q) Λ (¬(p) Λ ¬(r))) V ((r) Λ (¬(p) Λ ¬(q)))).

Simplificação de sentenças

Até este ponto, seguimos algumas convenções, por exemplo, colocando parênteses cercando cada sentença, seja ela atômica ou molecular. Estas convenções foram necessárias para que as estruturas das sentenças fossem entendidas de forma pedagogicamente mais fácil. No entanto, esta convenção provoca a existência de muitos parênteses chegando, em algumas situações, a confundir visualmente o leitor. Neste ponto, imaginamos que o leitor já tenha maturidade suficiente para simplificar algumas regras, por exemplo, eliminando alguns parênteses redundantes ou substituindo alguns conectivos por outros. As regras de simplificação são as seguintes:

Regra 1. *Os parênteses externos podem ser retirados.*

Neste caso, a sentença simbolizada *((p Λ (p→q))→q)* será escrita por *(p Λ (p→q))→q*.

Regra 2. *Os parênteses em torno da negação podem ser retirados.*

Neste caso, a sentença simbolizada $((\neg q) \land (p \rightarrow q)) \rightarrow (\neg q)$ será escrita por $(\neg q \land (p \rightarrow q)) \rightarrow \neg q$.

Regra 3. *Os conectivos* $\rightarrow$ *e* $\leftrightarrow$ *têm precedência sobre os conectivos* $\land$ *e* $\lor$ *.*

Neste caso, a sentença simbolizada $\neg q \land ((p \rightarrow q) \rightarrow \neg p)$ será escrita por $\neg q \land (p \rightarrow q) \rightarrow \neg p$.

Regra 4. *O conectivo* $\neg$ *se aplica à menor sentença que o sucede.*

Neste caso, deve-se escrever $\neg(p \land q)$ se a intenção for explicitar que o conectivo $\neg$ deve ser aplicado à sentença completa, porque se for escrito $\neg p \land q$ o conectivo $\neg$ será aplicado apenas à sentença p.

Regra 5. Os conectivos $\rightarrow$, $\leftrightarrow$ e $\land$ podem ser substituídos pelos conectivos $\neg$ e $\lor$ da seguinte forma:

- $p \rightarrow q$ pode ser substituído por $\neg p \lor q$
- $p \leftrightarrow q$ pode ser substituído por $(\neg p \lor q) \land (\neg q \lor p)$ e
- p ^ q pode ser substituído por $\neg (\neg p \lor \neg q)$.

A ***Regra 5*** informa que os únicos conectivos necessários para se construir qualquer sentença são **¬** e ***V***, ou seja, todos os outros podem ser construídos em função destes dois. Neste caso, diz-se que o conjunto {**¬, *V***} destes dois conectivos é "*completo*".

1.8 Função verdade

De acordo com o que foi visto até aqui, uma sentença é verdadeira ou falsa, de forma exclusiva, em um dado contexto.

Vamos agora analisar formas de avaliação de sentenças, ou seja, dada uma sentença *p*, vamos determinar se *p* é verdadeira ou falsa em um dado contexto. O nosso objetivo final é encontrar uma forma de determinar se uma determinada sentença é verdadeira ou falsa em todos os contextos possíveis.

Para facilitar este estudo, é necessário adotar uma notação. Já foi visto que uma sentença só pode ter um de dois valores: verdadeiro ou falso. Estes dois valores serão conhecidos como "*valores verdade*". Este termo pode gerar alguma dúvida, uma vez que ao nos referirmos aos valores verdade poder-se-ia prejulgar que a sentença fosse indubitavelmente verdadeira. No entanto, este não é o caso. Quando nos referimos ao valor verdade de uma função teremos como resultado o valor falso ou o valor verdadeiro, de forma excludente. Um valor verdadeiro será simbolizado pela letra maiúscula ***V*** e um valor falso será simbolizado pela letra maiúscula ***F.*** Desta forma, avaliar uma função consiste em determinar seu valor verdade: ***V*** ou ***F***.

O valor verdade de uma sentença atômica depende exclusivamente do contexto ao qual a sentença está associada. Por exemplo, apenas as pessoas com algum grau de conhecimento de Física sabem que a sentença "*a velocidade da luz, no vácuo, é de 300.000 quilômetros por segundo*" é uma sentença verdadeira. Já nas sentenças moleculares, várias situações podem acontecer, ou seja, seus valores verdade podem depender, ou não, dos valores verdade das sentenças componentes. Vamos analisar as duas situações através de exemplos.

Seja a sentença *((4 é par)* ∧ *(6 é ímpar))*. A conjunção faz alusão às duas sentenças, mas sabemos que a segunda delas é falsa. Desta forma, o valor verdade da sentença é falso (**F**). Neste caso, o valor verdade da sentença dependeu dos valores verdade de suas componentes.

Agora vejamos a sentença *((Zefinha é feia)* V *(Zefinha não é feia))*. Esta disjunção também é composta de duas sentenças, mas, nada podemos afirmar sobre a veracidade ou falsidade de cada uma delas. Se Zefinha for feia, a primeira componente é verdadeira e a segunda é falsa. Se Zefinha não for feia, então a segunda componente da disjunção é verdadeira e a primeira é falsa. Em ambas as situações, a sentença final é verdadeira. Neste caso, o valor verdade da sentença não dependeu dos valores verdade de suas componentes.

Para resolver este dilema, é necessário adotarmos uma convenção. Só nos interessa, enquanto estudantes de Lógica, analisar sentenças que possam ter seus valores verdade determinados apenas em função das suas componentes. Para isto, dizemos que um conectivo *é por função verdade* se o valor verdade das sentenças moleculares obtidas por seu intermédio for determinado, única e exclusivamente, baseado nos valores verdade de suas sentenças componentes. É importante destacar que os conectivos ***não, e, ou, se ... então*** e ***se, e somente se*** são por função verdade.

1.9 Regras de avaliação das sentenças

Para avaliar sentenças moleculares, é necessário que sejam definidas regras a serem aplicadas aos conectivos que as compõem. Isto significa que, para cada conectivo, será definida uma regra para encontrar o valor verdade da sentença composta por ele. Isto é o que será feito a seguir.

Negação

O conectivo ***não (¬)*** é utilizado quando desejamos negar o conteúdo de uma sentença. Na teoria dos conjuntos, ele tem outra notação para determinar a complementação de conjuntos, ou seja, uma barra horizontal sobre o símbolo do conjunto. Esta operação associa a cada conjunto ***A***, de um dado conjunto universo ***U***, um outro conjunto $\bar{A}$, chamado de *complemento* de ***A***, constituído pelos elementos de ***U*** que não pertencem a ***A***. Dado ***u*** em ***U***, a condição para que ***u*** esteja em $\bar{A}$ é que ***u*** não esteja em ***A*** e a condição para que ***u*** esteja em ***A*** é que ***u*** não esteja em $\bar{A}$.

Regra 1. Uma negação é verdadeira se a sentença negada for falsa e uma sentença é falsa se a sentença negada for verdadeira.

Esta regra está resumida na Tabela 1.5, a seguir, chamada de *tabela verdade do conectivo **não**.*

Tabela 1.5 - Tabela verdade do conectivo *não (¬)*.

p	*¬ p*
F	*V*
V	*F*

Em uma cidade em que cada habitante ou fala verdade ou é mentiroso, Félix encontrou Teresa e Nazareth quando perguntou: alguém de vocês é mentirosa? Teresa respondeu: "pelo menos uma de nós duas é mentirosa." Teresa estava, ou não, falando a verdade?

Conjunção

Na linguagem da Lógica, o conectivo ***e* (∧)** é utilizado quando queremos afirmar a ocorrência simultânea de dois fatos. Na teoria dos conjuntos, isto equivale à interseção de conjuntos.

Regra 2. Uma conjunção é verdadeira se suas duas componentes forem, simultaneamente, verdadeiras e é falsa se pelo menos uma delas for falsa. Esta regra permite construir a tabela verdade da conjunção, que está mostrada na Tabela 1.6, a seguir.

Tabela 1.6 - Tabela verdade do conectivo *e (Λ).*

p	*Q*	*p Λ q*
F	*F*	*F*
F	*V*	*F*
V	*F*	*F*
V	*V*	*V*

Disjunção

Na linguagem da Lógica, o conectivo ***ou (***V***)*** é utilizado quando se deseja apresentar alternativas. Na teoria dos conjuntos isto equivale à união dos conjuntos.

Regra 3. Uma disjunção é falsa se suas componentes forem, simultaneamente, falsas e é verdadeira se pelo menos uma de suas componentes for verdadeira.

Esta regra é utilizada no sentido inclusivo, ou seja, o fato das duas sentenças componentes da disjunção serem ambas verdade tornam a sentença também verdade. Esta situação é um pouco diferente do uso corriqueiro na língua portuguesa. Por exemplo, na sentença "*Félix vai de carro ou de avião*" queremos informar que ou Félix vai de carro ou ele vai de avião, uma vez que ele não pode ir de carro e de avião ao mesmo tempo. Nos circuitos digitais também existem implementações do operador ***ou exclusivo***, conhecido como ***Xor***, mas este caso não será aqui considerado. Será considerado apenas o ***ou inclusivo***. . Esta regra permite construir a tabela verdade da disjunção, como mostrado na Tabela 1.7, a seguir.

Tabela 1.7 - Tabela verdade do conectivo ou (V)

p	*q*	*p* V *q*
F	*F*	*F*
F	*V*	*V*
V	*F*	*V*
V	*V*	*V*

Implicação

Na Lógica, usa-se o conectivo de implicação (→) quando se deseja indicar uma relação de causa e efeito entre a sentença antecedente e a sentença consequente. Neste caso, uma interpretação gráfica possível em um diagrama de Venn só é possível se a implificação for transformada antes em uma dusjunção, ou seja transformando *p.→ q em ¬p* V *q.*

Regra 4. Uma implicação é falsa se sua antecedente for verdadeira e a conseqüente for falsa. Caso contrário, a sentença será verdadeira.

Esta regra permite construir a tabela verdade da implicação, que está mostrada na Tabela 1.8, a seguir.

Tabela 1.8 - Tabela verdade do conectivo *se .. então (→)*.

p	*Q*	*p → q*
F	*F*	*V*
F	*V*	*V*

V	***F***	***F***
V	***V***	***V***

Biimplicação

No estudo da Lógica, usa-se o conectivo ***se e somente se (↔)*** quando se deseja explicitar que duas sentenças têm o mesmo conteúdo. Neste caso, também não existe uma interpretação gráfica direta correspondente na teoria dos conjuntos, devendo antes ser transformada em outra possível.

Regra 5. Uma biimplicação é verdadeira se suas componentes possuírem os mesmos valores verdade e é falsa se elas apresentam valores verdade distintos.

Isto está mostrado na Tabela 1.9, a seguir.

Tabela 1.9 - Tabela verdade do conectivo *se e somente se (*↔

p	***q***	***p ↔ q***
F	***F***	***V***
F	***V***	***F***
V	***F***	***F***
V	***V***	***V***

Formalização conceitual

Os conceitos até aqui vistos foram introduzidos de maneira que o leitor pudesse entendê-los, antes que eles fossem formalizados. Esta formalização, apesar de necessária, pode confundir didaticamente o leitor que está tendo os primeiros contactos com o estudo da Lógica. Esta seção é dedicada à formalização destes conceitos para facilitar a continuação deste aprendizado e tornar este trabalho autocontido.

Definição 1.12 (**Alfabeto**). O alfabeto da Lógica Proposicional é constituído por:

- Símbolos de pontuação: (e).
- Símbolos de verdade: *true* e *false*.
- Símbolos proposicionais: p, q, r. s, p_1, q_1, r_1, s_1, p_2,...
- Conectivos proposicionais: $\neg$, $\vee$, $\wedge$, $\rightarrow$ e $\leftrightarrow$

Como pode ser verificado, o alfabeto da linguagem da Lógica Proposicional é constituído de infinitos símbolos. Isto não ocorre no alfabeto da língua portuguesa que é composto de apenas 26 letras e mais alguns poucos. Os símbolos de verdade são as palavras da língua inglesa *true* e *false* que, no presente contexto, são consideradas apenas símbolos. Em outros contextos, estes símbolos podem ser representados de forma diferente.

Como pode ser observado, na definição de Alfabeto da Lógica, ele é formado por símbolos. Estes símbolos devem ser concatenados para formar estruturas que serão tratadas como fórmulas bem formadas, ***fbfs***. A construção das fórmulas bem formadas obedecem a leis de formação que serão mostradas a seguir.

Definição 1.13 (**fbf – fórmula bem formada**). As fórmulas bem formadas da linguagem da Lógica Proposicional são as ***fbfs proposicionais*** construídas a partir dos símbolos de alfabeto conforme as regras a seguir:

- Todo símbolo de verdade é uma fórmula bem formada.
- Todo símbolo proposicional é uma fórmula bem formada.
- Se p for uma fórmula bem formada, então $(\neg p)$, a negação de p, também é uma fórmula bem formada.
- Se p e q forem fórmulas bem formadas, então $(p \vee q)$ também será uma fórmula bem formada, a disjunção de p e q.
- Se p e q forem fórmulas bem formadas, então $(p \wedge q)$ também será uma fórmula bem formada, a conjunção de p e q.
- Se p e q forem fórmulas bem formadas, então $(p \rightarrow q)$ também será uma fórmula bem formada, a implicação de p para q, onde p é o antecedente e q é o consequente.
- Se p e q forem fórmulas bem formadas, então $(p \leftrightarrow q)$ também será uma fórmula bem formada, a bimplicação de p para q, onde p é o lado esquerdo e q é o lado direito.

Interpretações

Já foi visto neste estudo que existem sentenças, ou fbfs, que são sempre verdadeiras, independente dos valores verdade de suas componentes. Outras há que são sempre falsas e existe ainda um terceiro tipo que apresentam sentenças verdadeiras ou falsas, dependendo do contexto em que elas estejam inseridas, Por exemplo, as sentenças:

a) *Amanhã vai chover ou não vai chover.*
b) *Chove e não chove hoje.*
c) *Emiliano come muito e Maria gosta de bananas.*

A sentença a) será sempre verdadeira, independente se chover ou não. A sentença b) é sempre falsa, independente de São Pedro gostar ou não. Já a sentença c) pode ser verdadeira ou falsa. Estas três situações estão detalhadas nas tabelas seguintes onde para cada uma destas sentenças, é construída a sua tabela verdade, para fins de comparação e permitir um completo domínio por parte do leitor.

1. *Amanhã vai chover ou não vai chover.*

p : amanhã vai chover.

p	*(¬p)*	*(p ∨ (¬p))*
F	*V*	*V*
V	*F*	*V*

2. *Chove e não chove hoje.*

p : Hoje chove

p	*(¬p)*	*(p ∧ (¬p))*
F	*V*	*F*
V	*F*	*F*

3. *Emiliano come muito e Maria gosta de bananas.*

p : Emiliano come muito
q : Maria gosta de bananas.

p	*Q*	*(p ∧ q)*
F	*F*	*F*
F	*V*	*F*
V	*F*	*F*
V	*V*	*V*

As sentenças construídas através dos conectivos mostrados até aqui podem ser analisadas apenas utilizando a tabela verdade das sentenças componentes.

Definição 1.14. *Uma interpretação para uma sentença simbolizada α é uma atribuição de valores verdade às letras sentenciais que ocorrem em α, de modo que a cada letra seja atribuído um único valor verdade.*

No exemplo mostrado, nas letras a) e b) só existe uma letra sentencial, *p*, portanto ela só pode ter dois valores verdade: ***F*** ou ***V***. Isto significa que só temos duas interpretações para ela. Já no caso c), existem duas letras sentenciais, portanto existem 4 interpretações para esta sentença. De forma geral, uma sentença simbolizada onde ocorrem ***m*** letras sentenciais distintas, possui $\boldsymbol{2^m}$ interpretações.

Tautologias, contradições e contingências

Nesta seção, as sentenças serão classificadas de acordo com as suas interpretações. Para isto, será necessário definir antes o que se entende por interpretação.

Definição 1.15. Uma interpretação para duas sentenças simbolizadas α e β é uma atribuição de valores às letras sentenciais que ocorrem em α e em β. Isto significa que cada linha da tabela verdade representa uma interpretação para as letras sentenciais constantes desta tabela.

Definição 1.16. Uma sentença simbolizada α é chamada *tautologia* se, para todas as interpretações, seus valores verdade forem todos verdadeiros (***V***).

Vejamos a sentença α : $(p \rightarrow q) \vee \neg (p \rightarrow q)$. A Tabela 1.10 mostra a sua tabela verdade.

Tabela 1.10. Tabela verdade da sentença α.

p	q	$p \rightarrow q$	$\neg (p \rightarrow q)$	$(p \rightarrow q) \vee \neg (p \rightarrow q)$
F	***F***	***V***	***F***	***V***
F	***V***	***V***	***F***	***V***
V	***F***	***F***	***V***	***V***
V	***V***	***V***	***F***	***V***

Como pode ser observado, para todas as interpretações de p e q, a sentença α tem valor verdade ***V***. Assim, a sentença α é uma tautologia.

Para informar que uma sentença α é uma tautologia, será utilizada a notação $\models \alpha$. Esta notação será utilizada a partir deste instante e será importante na formulação de provas, um tema a ser analisado mais adiante.

Definição 1.17. Uma sentença simbolizada α é chamada de <u>*contradição*</u> se, para todas as interpretações, seus valores verdade forem falsos (***F***).

Seja, por exemplo, a sentença α : $(p \wedge q) \leftrightarrow (\neg p \vee \neg q)$. Sua tabela verdade é a seguinte:

p	q	$p \wedge q$	$\neg p$	$\neg q$	$\neg p \vee \neg q$	$p \wedge q \leftrightarrow \neg p \vee \neg q$
F	***F***	***F***	***V***	***V***	***V***	***F***
F	***V***	***F***	***V***	***F***	***V***	***F***
V	***F***	***F***	***F***	***V***	***V***	***F***
V	***V***	***V***	***F***	***F***	***F***	***F***

Pode-se observar que a última coluna desta tabela verdade só contém valores verdade ***F***. Portanto, a sentença é uma contradição.

Definição 1.18. Uma sentença simbolizada α é chamada de <u>*contingência*</u> se algum ou alguns de seus valores verdade forem verdadeiros (***V***), enquanto outro ou outros têm valores verdade (***F***).

Se uma sentença for uma contingência, diz-se que ela é uma fórmula ***satisfatível,*** ou seja, uma sentença é ***satisfatível*** se existe pelo menos uma interpretação para a qual o valor verdade da sentença é verdadeiro.

O exemplo a seguir mostra uma sentença classificada como uma contingência, uma vez que, em sua última coluna, se verificam valores ***F*** e também ***V***. Seja a sentença α : $(q \wedge (p \to q)) \to p$.

Sua tabela verdade é a seguinte.

p	q	$p \to q$	$q \land (p \to q)$	$(q \land (p \to q)) \to p$
F	***F***	***V***	***F***	***V***
F	***V***	***V***	***V***	***F***
V	***F***	***F***	***F***	***V***
V	***V***	***V***	***V***	***V***

Pelo que foi visto até aqui, uma forma prática de classificar uma determinada sentença como tautologia, contingência ou contradição é analisar seus valores verdade, utilizando sua tabela verdade.

Equivalência Tautológica

Até o momento, aprendemos como classificar uma sentença como tautologia, contradição ou contingência. No entanto, é também importante saber comparar duas sentenças e analisar o que há de comum entre elas. Assim, temos:

Definição 1.19. Duas sentenças simbolizadas α e β são tautologicamente equivalentes se, para cada interpretação de α e de β, os valores de α e β forem iguais.

Duas sentenças α e β, tautologicamente equivalentes, serão denotadas por $\alpha \models\!\mid \beta$.

Exemplo 1.22. As sentenças, a seguir, são tautologicamente equivalentes. No entanto suas verificações são deixadas como exercício para o leitor.

a) $(p \lor q) \rightarrow r \;\models\!\!\dashv\; \neg r \rightarrow (\neg p \land \neg q)$

b) $(p \lor q) \land \neg(p \land q) \;\models\!\!\dashv\; \neg((p \lor \neg q) \land (\neg p \lor q))$

c) $(p \lor \neg q) \land (\neg p \lor q) \;\models\!\!\dashv\; (p \leftrightarrow q)$

Proposição. Sendo α e β duas sentenças simbolizadas, então as seguintes condições são equivalentes:

a) $\alpha \;\models\!\!\dashv\; \beta$

b) $\models \alpha \leftrightarrow \beta$

Prova: O sistema de prova utilizado para esta demonstração baseia-se na verificação de que a primeira sentença é equivalente a segunda e que a segunda também é equivalente à primeira. Este sistema é conhecido como *ida e volta*.

(➔) Suponhamos que $\alpha \;\models\!\!\dashv\; \beta$.

Então, em cada interpretação para α e β, elas assumem o mesmo valor verdade, pela definição dada. Observando a tabela verdade de $\alpha \leftrightarrow \beta$, verifica-se também que, em cada linha, α e β assumem valores iguais. Isto significa que as sentenças α e β são tautologicamente equivalentes, ou seja, $\models \alpha \leftrightarrow \beta$.

(🡐) Suponhamos agora que $\models \alpha \leftrightarrow \beta$.

Então, em cada linha da tabela verdade de $\alpha \leftrightarrow \beta$ ocorre o valor verdade ***V***. Isto significa que, em cada linha, os valores verdade de α e de β são iguais. Como cada linha da tabela verdade inicia com uma interpretação para α e β, as sentenças α e β assumem o mesmo valor verdade em cada interpretação, ou seja, $\alpha \;\models\!\!\dashv\; \beta$

As três leis do pensamento

Alguns pesquisadores definiram a Lógica como a ciência das leis do pensamento. Estas pessoas defenderam que existem exatamente três fundamentais do pensamento, as quais são necessárias e suficientes para que o pensamento se desenvolva de forma correta.

Estas leis do pensamento receberam, tradicionalmente, as denominações de *Princípio da Identidade*, *Princípio da Contradição* (algumas vezes tratado como Princípio da Não-Contradição) e o *Princípio do Terceiro Excluído*. Existem formulações alternativas para estes princípios, de acordo com os diferentes contextos. Em nosso caso, as formulações são as seguintes:

- *O Princípio da Identidade* afirma que se qualquer enunciado for verdadeiro, então ele será verdadeiro.

- *O Princípio da Contradição* afirma que nenhum enunciado pode ser verdadeiro e falso ao mesmo tempo.

- *O princípio do Terceiro Excluído* afirma que um enunciado ou é verdadeiro ou falso.

O Princípio da Identidade afirma que todo enunciado da forma $p \rightarrow p$ é verdadeiro, ou seja, todo enunciado deste tipo é uma tautologia.

O Princípio da Contradição afirma que todo enunciado da forma $p \wedge \neg p$ é falso, ou seja, todo enunciado deste tipo é uma contradição.

O Princípio do Terceiro Excluído afirma que todo enunciado da forma $p \vee \neg p$ é verdadeiro, ou seja, todo enunciado deste tipo é uma tautologia.

Estes princípios tem sido criticados ao longo dos tempos, mas, em sua grande maioria, estas críticas parecem basear-se em falta de entendimento correto. O Princípio da Identidade foi criticado com fundamento em que as coisas mudam, visto que, o que era verdadeiro em relação determinado fato no passado pode não sê-lo em um momento futuro. Por exemplo, o Brasil era uma colônia de Portugal até 1822, quando deixou de sê-lo. Esta observação é correta, mas esse sentido não é aquele do qual a Lógica se ocupa. Os “enunciados” cujos valores verdade mudam com o tempo são expressões *elípticas* ou incompletas de proposições que não mudam e são destas que a Lógica se ocupa. Desta forma, o enunciado “o Brasil é uma colônia de Portugal” pode ser considerada uma expressão elíptica ou parcial de “o Brasil foi uma colônia de Portugal até 1922”, o que é tão verdadeiro no século XIX, quanto atualmente. Quando limitamos nossa atenção aos enunciados *não elípticos* ou completos, o Princípio da Identidade é perfeitamente verdadeiro e indiscutível.

De forma similar, o Princípio da Contradição foi criticado por semânticos, em geral por marxistas com o fundamento de que há contradições, ou situações nas quais forças contraditórias ou conflitantes estejam em ação. É verdade que existem situações com forças conflitantes, e isto é verdadeiro no domínio da mecânica e nas esferas social e econômica. No entanto, é uma terminologia vaga e inconveniente chamar de “contraditórias” essas forças conflitantes. O calor aplicado a um gás contido tende a provocar a expansão e o recipiente que tende a conter a expansão desse gás podem ser descritos como um conflito mútuo, mas nenhum deles é a negação ou a contradição do outro. O proprietário de uma fábrica que tem milhares de operários trabalhando em conjunto para o seu funcionamento pode opor-se ao sindicato e ser combatido por este, que jamais teria se organizado se seus filiados não tivessem se reunidos para trabalhar nessa fábrica, mas nem o proprietário nem o sindicato são a negação ou o contraditório um do outro. Quando entendido no sentido em que se considera correto, o Princípio da Contradição é perfeitamente verdadeiro e igualmente indiscutível.

O Princípio do Terceiro Excluído tem sido objeto de mais ataques do que os outros dois. Afirma-se insistentemente que a sua aceitação leva a uma "orientação bivalente" que implica, entre outras coisas, que tudo seja branco ou preto, excluindo todos os outros domínios intermediários. Mas ainda que o enunciado "isto é branco" (em que a palavra "isto" se refere, exatamente, à mesma coisa em ambos os enunciados), um não é a negação ou o contraditório do outro. Indubitavelmente, não podem ser ambos verdadeiros, mas podem ser ambos falsos. São contrários, mas não contraditórios. A negação ou contradição de "isto é branco" é "isto não é branco" e um destes enunciados deve ser verdadeiro – se a palavra "branco" for usada nos dois enunciados, exatamente no mesmo sentido. Quando restrito a enunciados que contém termos totalmente isentos de ambiguidades e absolutamente rigorosos, o Princípio do Terceiro Excluído é verdadeiro.

Embora os três princípios sejam verdadeiros, poder-se-á duvidar, contudo, de que possuam o status privilegiado e fundamental que tradicionalmente lhes é atribuído. O primeiro e o terceiro não são as únicas formas de tautologia; nem a contradição explícita $p \land \neg p$ é a única forma de contradição. Na realidade, as três leis do pensamento representam os princípios básicos que governam a construção das tabelas verdade.

Regras de inferência

A Tabela 1.11, a seguir, mostra as principais equivalências tautológicas que, por serem muito utilizadas, ficaram conhecidas de forma generalizada como "*regras de inferência",* cada uma delas com uma denominação particular.

Tabela 1.11. Principais regras de inferência.

Regras	**Fórmula**
Modus ponens	$(p \land (p \rightarrow q)) \rightarrow q$
Modus tollens	$(\neg q \land (p \rightarrow q)) \rightarrow \neg p$
Silogismo hipotético	$((p \rightarrow q) \land (q \rightarrow r)) \rightarrow (p \rightarrow r)$
Silogismo disjuntivo	$((p \lor q) \land \neg p) \rightarrow q$
Simplificação	$(p \land q) \rightarrow p$
Adição	$p \rightarrow (p \lor q)$
Eliminação	$((p \rightarrow q) \lor (r \land \neg q)) \rightarrow (p \rightarrow r)$
Prova por casos	$(((p \rightarrow r) \land (q \rightarrow r)) \rightarrow (p \lor q)) \rightarrow r$
Contraposição	$(p \rightarrow q) \rightarrow (\neg q \rightarrow \neg p)$

Existem outras equivalências tautológicas, algumas delas mostrando alguma propriedade, por exemplo, a associatividade e a comutatividade de alguns conectivos. Incentiva-se que o leitor verifique a veracidade de cada uma delas como exercício.

a) $\neg\neg p \;\models\!\!\!\dashv\; p$
b) $p \land q \;\models\!\!\!\dashv\; q \land p$
c) $p \lor q \;\models\!\!\!\dashv\; q \lor p$
d) $(p \land q) \land r \;\models\!\!\!\dashv\; p \land (q \land r)$
e) $(p \lor q) \lor r \;\models\!\!\!\dashv\; p \lor (q \lor r)$
f) $(p \land q) \lor r \;\models\!\!\!\dashv\; (p \lor r) \land (q \lor r)$
g) $(p \lor q) \land r \;\models\!\!\!\dashv\; (p \land q) \lor (q \land r)$

h) $\neg(p \wedge q) \models\!\!\dashv \neg p \vee \neg q$
i) $\neg(p \vee q) \models\!\!\dashv \neg p \wedge \neg q$
j) $p \wedge p \models\!\!\dashv p$
k) $p \vee p \models\!\!\dashv p$
l) $p \vee (p \wedge q) \models\!\!\dashv p$
m) $p \wedge (p \vee q) \models\!\!\dashv p$

Formas normais conjuntivas e disjuntivas

Algumas equivalências tautológicas permitem transformar qualquer sentença em uma outra, logicamente equivalente, mas que não contenha os conectivos $\rightarrow$ ou $\leftrightarrow$. Neste caso, a sentença resultante conterá apenas os conectivos **¬**, $\vee$ e $\wedge$. Diz-se que esta sentença está em sua *forma normal,* que pode ser *disjuntiva* (***FND***) ou *conjuntiva* (***FNC***). Estas formas têm aplicação muito importante na construção otimizada de circuitos digitais, um campo de aplicação da Lógica de Boole, um tema a ser analisado mais adiante, ainda nesta Unidade.

O algoritmo para fazer esta transformação é o seguinte:

1. substituem-se as fórmulas: $p \rightarrow q$ por $\neg p \vee q$ e $p \leftrightarrow q$ por $(\neg p \vee q) \wedge (p \vee \neg q)$.
2. eliminam-se as negações que precedem os parênteses, substituindo $\neg(p \wedge q)$ por $\neg p \vee \neg q$ e $\neg(p \vee q)$ por $\neg p \wedge \neg q$.
3. eliminam-se as negações múltiplas, substituindo-se $\neg(\neg p)$ por p.
4. para se obter a ***FNC***, substituem-se as fórmulas do tipo $p \vee (q \wedge r)$ por $(p \vee q) \wedge (p \vee r)$.
5. para se obter a ***FND***, substituem-se as fórmulas do tipo $p \wedge (q \vee r)$ por $(p \wedge q) \vee (p \wedge r)$.

Exemplo 1.23. As fórmulas a seguir estão em suas formas normais:

*a)**FNC**: (¬p V q) ∧ (r V s V p)*

*b)***FND**: *p V (q ∧ r) V (¬s ∧ p)*

Exercícios

1. Encontre as formas normais conjuntivas (FNC) das seguintes sentenças:

a. *¬ (p V q) ∧ (r → ¬p)*

b. *(p V ¬q) V (r → ¬p)*

c. *(p Λ ¬q) V (r ↔ ¬p)*

2. Encontre as formas normais disjuntivas (FND) das mesmas sentenças do exercício anterior.

Emil Leon Post

O problema de Post

Até aqui foi visto como construir sentenças a partir de sentenças componentes. Podemos perguntar se é possível fazer o processo inverso, ou seja, encontrar as sentenças componentes a partir da sentença final. Este problema foi analisado pelo pesquisador Emil Leon Post (1888-1995) que chegou à conclusão de que era possível encontrar as componentes a partir das formas normais disjuntivas ou conjuntivas utilizando o método da tabela verdade.

Obtendo a forma normal disjuntiva

Neste caso, teremos de seguir o algoritmo a seguir:

1. Observam-se todas as linhas da tabela verdade que possuem valores verdade V para a sentença final .

2. Para cada linha de valor verdade V, constrói-se a conjunção dos valores verdade de cada sentença atômica componente.
3. Faz-se a disjunção das conjunções anteriores.

Exemplo 1.24. Encontrar a forma normal disjuntiva que satisfaça a seguinte tabela verdade:

p	***q***	***função***	***Conjunções***
F	***F***	***V***	$(\neg p \land \neg q)$
F	***V***	***F***	
V	***F***	***F***	
V	***V***	***V***	$(p \land q)$

Na coluna da função aparecem valores ***V*** na primeira e na quarta linhas. Na primeira linha p tem valor verdade ***F***, logo vai entrar na conjunção como $\neg p$. Da mesma forma, q. Já na quarta linha, p e q têm valores verdade ***V***. Logo, entram na conjunção como p e q. Isto significa que a forma normal disjuntiva para esta função é $(\neg p \land \neg q) \lor (p \land q)$.

Obtendo a forma normal conjuntiva

Para a obter a forma normal conjuntiva, o algoritmo deve ser o mesmo, substituindo os valores ***V*** por ***F*** e ***F*** por ***V*** e as conjunções por disjunções e vice-versa.

Exemplo 1.25. Encontrar a forma normal conjuntiva que satisfaça a seguinte tabela verdade (a mesma do *Exemplo anterior*)*:*

p	*q*	*função*	*conjunções*
F	*F*	*V*	
F	*V*	*F*	$p \vee \neg q$
V	*F*	*F*	$\neg p \vee q$
V	*V*	*V*	

Na coluna da função aparecem valores F na segunda e na terceira linhas. Na segunda linha *p* tem valor verdade ***F***, logo vai entrar na disjunção como *p* e *q* tem o valor verdade V, logo vai entrar na disjunção como $\neg q$. Já na terceira linha, *p* tem o valor verdade V, logo vai entrar na disjunção como $\neg p$ e a variável *q* tem o valor verdade F, logo entra na disjunção como *q*. Isto significa que a forma normal conjuntiva deve ser $(p \vee \neg q) \wedge (\neg p \vee q)$.

Exercício. Verifique se a forma normal conjuntiva do Exemplo anterior é equivalente à forma normal disjuntiva do mesmo exemplo.

Argumento válido

O principal objetivo do estudo da Lógica na computação é encontrar formas de se verificar se uma sentença, que depende de seus conectivos e pode ser grande, é, ou não, verdadeira. Resumidamente, estamos interessados em encontrar formas de verificar se uma determinada argumentação é logicamente verdadeira ou falsa.

Definição 1.20. Chama-se argumento toda seqüência de proposições $p_0, p_1, ..., p_{n-1}, p_n$, com $n \in \mathbf{N}$, $n \geq 0$, onde as proposições $p_0, p_1, ..., p_{n-1}$ são chamadas "***premissas***" e a proposição p_n é chamada **"*conclusão*"**.

Definição 1.21. Diz-se que um argumento $p_0, p_1, ..., p_{n-1}, p_n$ é "***válido***", se e somente se, sendo as premissas verdadeiras a conclusão também é verdadeira, ou seja, se e somente se, a fórmula $(p_0 \wedge p_1 \wedge ... \wedge p_{n-1}) \rightarrow p_n$ for uma tautologia.

As seguintes afirmações são formas distintas de se expressar a mesma coisa:

- $p_0 \wedge p_1 \wedge ... \wedge p_{n-1} \models p_n$
- $p_0, p_1, ..., p_{n-1}$ acarreta p_n
- p_n decorre de $p_0, p_1, ..., p_{n-1}$
- p_n se deduz de $p_0, p_1, ..., p_{n-1}$
- p_n se infere de $p_0, p_1, ..., p_{n-1}$

Uma forma de se verificar se uma seqüência de proposições é, ou não, um argumento válido é utilizar a tabela verdade.

Exemplo 1.26. A sequência $p, q \rightarrow r, \neg r, \neg q$ é um argumento válido. Para verificar isto, verifiquemos que a tabela verdade da proposição $(p \wedge q \rightarrow r \wedge \neg r) \rightarrow \neg q$ é uma tautologia.

p	q	r	$q \rightarrow r$	$\neg r$	$p \wedge q \rightarrow r \wedge \neg r$	$\neg q$	$p \wedge q \rightarrow r \wedge \neg r \rightarrow \neg q$
F	F	F	V	V	F	V	V
F	F	V	V	F	F	V	V
F	V	F	F	V	F	F	V
F	V	V	V	F	F	F	V

V	F	F	V	V	V	V	V
V	F	V	V	F	F	V	V
V	V	F	F	V	F	F	V
V	V	V	V	F	F	F	V

Demonstração de validade de argumentos

Até este ponto foi vista uma técnica de demonstração sobre sentenças proposicionais utilizando sempre a tabela verdade. Esta técnica tem sido utilizada com sucesso, dada a naturalidade e facilidade como ela é feita. No entanto, a técnica da tabela verdade não é interessante quando existem muitos símbolos proposicionais, porque a tabela se torna muito grande. Assim, se torna importante encontrar outras formas de demonstração, sendo este o objetivo desta seção.

Uma sequência de demonstração é uma sequência de fbfs nas quais cada fbf é uma hipótese ou o resultado da aplicação de uma ou mais regras de dedução do sistema formal às fbfs anteriores na sequência.

Entre estas formas, podem ser citadas:

- demonstração direta,
- demonstração indireta condicional,
- demonstração indireta por absurdo e
- demonstração indireta por árvores de refutação.

Demonstração direta

Esta forma de demonstração ou de dedução de uma conclusão p_n a partir de um conjunto de premissas consiste em aplicar-se as equivalências tautológicas e as regras de inferências vistas anteriormente. Vamos verificar isto através de dois exemplos.

***Exemplo 1.27*.** Demonstrar a validade do argumento $p, q \rightarrow r, \neg r, \neg q$ do *Exemplo anterior.*

Demonstração:

1. p premissa
2. $q \rightarrow r$ premissa
3. $\neg r$ premissa
4. $\neg q$ Conclusão: verdade por Modus Tollens entre as premissas 2 e 3.

Exemplo 1.28. Demonstrar a validade do argumento $\neg p \rightarrow q$, $q \rightarrow \neg r$, $r \vee s$, $\neg s \rightarrow p$. Demonstração:

1.	$\neg p \rightarrow q$	premissa
2.	$q \rightarrow \neg r$	premissa
3.	$r \vee s$	premissa
4.	$\neg p \rightarrow \neg r$	por Silogismo hipotético entre 1 e 2
5.	$\neg r \rightarrow s$	pela substituição de V por → em 3.
6.	$\neg p \rightarrow s$	por Silogismo hipotético entre 4 e 5
7.	$\neg s \rightarrow \neg\neg p$	por Contraposição de 6
8.	$\neg s \rightarrow p$.	Conclusão: pela dupla negação de 7.

Demonstração indireta condicional

Para demonstrar a validade de argumentos cuja conclusão é uma fórmula condicional do tipo $p \rightarrow q$, considera-se o antecedente p como uma premissa adicional e o conseqüente q será a conclusão a ser demonstrada. De fato,

1. $p_0, p_1, \ldots, p_{n-1}, p, q$ sendo um argumento válido, então
2. $p_0, p_1, \ldots, p_{n-1}, p \models q$, ou seja, isto significa que
3. $((p_0 \wedge p_1 \wedge \ldots \wedge p_{n-1}) \wedge p) \rightarrow q$ é uma tautologia. Isto significa que
4. $(p_0 \wedge p_1 \wedge \ldots \wedge p_{n-1}) \rightarrow (p \rightarrow q)$ é uma tautologia. Isto quer dizer

5. $p_0,\ p_1,\ \ldots,\ p_{n-1} \models p \rightarrow q$, ou seja, que
6. $p_0,\ p_1,\ \ldots,\ p_{n-1},\ p \rightarrow q$ é um argumento válido.

Exemplo 1.29. Usando o esquema de demonstração indireta condicional, demonstrar a validade do argumento $\neg p \rightarrow q,\ q \rightarrow \neg r,\ r \vee s,\ \neg s \rightarrow p.$

Demonstração:

1. $\neg p \rightarrow q$ premissa
2. $q \rightarrow \neg r$ premissa
3. $r \vee s$ premissa
4. $\neg s$ premissa adicional
5. r por silogismo disjuntivo entre 3 e 4
6. $\neg p \rightarrow \neg r$ por silogismo hipotético entre 1 e 2
7. $r \rightarrow p$ por contraposição de 6
8. p Conclusão: *Modus Ponens* entre 5 e 7.

Demonstração indireta por absurdo

Para se construir um esquema de demonstração por absurdo de um argumento $p_0,\ p_1,\ \ldots,\ p_{n-1},\ p$ considera-se a negação da conclusão, $\neg p$, como premissa adicional e conclui-se por uma contradição. De fato,

1. $p_0, p_1, \ldots, p_{n-1}, \neg p$ uma contradição, então
2. $p_0, p_1, \ldots, p_{n-1}, \models \neg p \rightarrow \boldsymbol{F}$, ou seja, isto significa que
3. $p_0, p_1, \ldots, p_{n-1}, \models \neg\neg p \vee \boldsymbol{F}$ pela definição de implicação, ou seja,
4. $p_0, p_1, \ldots, p_{n-1}, \models p \vee \boldsymbol{F}$ pela idempotência. Isto significa que
5. $p_0, p_1, \ldots, p_{n-1}, \models p$ pela propriedade do conectivo V (ou). Isto significa que
6. $p_0, p_1, \ldots, p_{n-1}, p$ é um argumento válido.

Exemplo 1.30. Usando o esquema de demonstração por absurdo, demonstrar a validade do argumento $\neg p \rightarrow q$, $q \rightarrow \neg r$, $r \vee s$, $\neg s \rightarrow p$.

1.	$\neg p \rightarrow q$	premissa
2.	$q \rightarrow \neg r$	premissa
3.	$r \vee s$	premissa
4.	$\neg(\neg s \rightarrow p)$	premissa adicional
5.	$\neg p \rightarrow \neg r$	por silogismo hipotético entre 1 e 2
6.	$\neg r \rightarrow s$	pela substituição de V por → em 3
7.	$\neg p \rightarrow s$	por silogismo hipotético entre 5 e 6
8.	$\neg s \rightarrow p$	pela contraposição de 7
9.	$\neg(\neg s \rightarrow p) \wedge (\neg s \rightarrow p)$	pela conjunção de 4 e 8.
10.	***F***	Conclusão: pela contradição de 9.

Isto significa que quando se supõe que a conclusão de um argumento dado é uma contradição chega-se a uma contradição. Isto significa que a suposição inicial não é válida. Portanto, o argumento é válido.

Demonstração indireta por árvores de refutação

A árvore de refutação, também conhecida como *Tableau Semântico*, é um outro método empregado para se analisar a validade de um argumento. O método é adequado para ser implementado em computadores e é baseado na demonstração por absurdo.

O processo de construção de árvores de refutação é baseado em regras que dependem dos tipos dos conectivos que compõem as sentenças que vão gerar uma derivação. Assim, torna-se necessário definir um conjunto de regras de derivação para cada conectivo.

Regra da conjunção R1 (^). Uma fórmula do tipo *p* ^ *q* gera duas linhas e escrevem-se as fórmulas *p* e *q* em cada uma delas. Procede-se assim em todos os ramos abertos aos quais a fórmula *p* ^ *q* pertence pois *p* ^ *q* assume o valor ***V*** se, e somente se, as fórmulas *p* e *q* forem ambas verdadeiras. O sinal ∠ significa que a sentença $p \land q$ foi substituída e não deve ser mais usada.

1. $p \land q$ ∠
2. p
3. q

Regra da disjunção R2 (∨***).*** Uma fórmula do tipo *p v q* gera uma linha e dois ramos, escrevendo-se na linha e, em cada ramo, as fórmulas *p* e q, respectivamente. Procede-se assim em todos os ramos abertos aos quais a fórmula *p v q* pertence, pois *p v q* assume o valor ***V*** se, e somente se, a fórmula *p* for verdadeira ou se *q* for verdadeira.

```
1.  p ∨ q  ∠
        / \
2.  p    q
```

Regra da implicação R3 (→). Uma fórmula do tipo $p \to q$ gera uma linha e dois ramos e escreve-se, na linha e em cada ramo, as fórmulas $\neg p$ e *q*. Precede-se assim em todos os ramos abertos aos quais a fórmula $p \to q$ pertence pois, $p \to q$ assume valores ***V*** se, e somente se, a fórmula $\neg p$ for verdadeira ou se *q* for verdadeira, ou seja, $p \to q = (\neg p \lor q)$.

```
1.p → q ∠
      / \
2.¬p   q
```

Regra da biimplicação R4 ($\leftrightarrow$). Uma fórmula do tipo $p \leftrightarrow q$ gera duas linhas e dois ramos e escreve-se, nas linhas as fórmulas p e q em um ramo e as fórmulas $\neg p$ e $\neg q$ no outro ramo. Procede-se assim em todos os ramos abertos aos quais a fórmula $p \leftrightarrow q$ pertence pois, $p \leftrightarrow q$ assume o valor ***V*** se, e somente se, a fórmula $p \land q$ for verdadeira ou se a fórmula $\neg p \land \neg q$ for verdadeira, ou seja, $p \leftrightarrow q = (p \land q) \lor (\neg p \land \neg q)$.

1. $p \leftrightarrow q$ ✓
        / \
2. p $\neg p$
3. q $\neg q$

Regra da dupla negação R5 ($\neg\neg$). Uma fórmula do tipo $\neg\neg p$ gera uma linha e escreve-se p na linha. Procede-se assim em todos os ramos abertos aos quais a fórmula $\neg\neg p$ pertence, uma vez que $\neg\neg p$ é verdadeira se e somente se p também a for.

1. $\neg\neg p$ ✓
2. p

Regra da negação da conjunção R6 ($\neg \land$). Uma fórmula do tipo $\neg(p \land q)$ gera uma linha e dois ramos escrevem-se na linha e em cada ramo as fórmulas $\neg p$ e $\neg q$, respectivamente. Procede-se assim em todos os ramos abertos aos quais a fórmula $\neg(p \land q)$ pertence, pois $\neg(p \land q)$ assume o valor ***V*** se, e somente se, a fórmula $\neg p$ for verdadeira ou se a fórmula $\neg q$ for verdadeira, ou seja, $\neg(p \land q) = \neg p \lor \neg q$.

1. $\neg(p \land q)$ ✓
        / \
2. $\neg p$ $\neg q$

Regra da negação da disjunção R7 ($\neg\vee$). Uma fórmula do tipo $\neg(p \vee q)$ gera duas linhas e escrevem-se as fórmulas $\neg p$ e $\neg q$ em cada linha. Procede-se assim em todos os ramos abertos aos quais a fórmula $\neg(p \vee q)$ pertence, pois $\neg(p \vee q)$ assume o valor ***V*** se, e somente se, as fórmulas $\neg p$ e $\neg q$ forem verdadeiras, ou seja, $\neg(p \vee q) = \neg p \wedge \neg q$

1. $\neg(p \vee q)$ ✓
2. $\neg p$
3. $\neg q$

Regra da negação da implicação R8 ($\neg\rightarrow$).Uma fórmula do tipo $\neg(p \rightarrow q)$ gera duas linhas e escrevem-se as fórmulas p e $\neg q$ em cada linha. Procede-se assim em todos os ramos abertos aos quais a fórmula $\neg(p \rightarrow q)$ pertence, pois $\neg(p \rightarrow q)$ assume o valor ***V*** se, e somente se, as fórmulas p e $\neg q$ forem verdadeiras, ou seja, . $\neg(p \rightarrow q) = \neg(\neg p \vee q) = p \wedge \neg q$.

1. $\neg(p \rightarrow q)$ ✓
2. p
3. $\neg q$

Regra da negação da biimplicação R9 ($\neg\leftrightarrow$).Uma fórmula do tipo $\neg(p \leftrightarrow q)$ gera duas linhas e dois ramos e escrevem-se nas linhas, as fórmulas $\neg p$ e q em um ramo e as fórmulas p e $\neg q$ no outro ramo. Procede-se assim em todos os ramos abertos aos quais a fórmula $\neg(p \leftrightarrow q)$ pertence, pois $\neg(p \leftrightarrow q)$ assume o valor V se, e somente se, a fórmula $(\neg p \wedge q)$ for verdadeira ou se a fórmula $(p \wedge \neg q)$ for verdadeira, ou seja,

$\neg(p \leftrightarrow q) = (\neg p \wedge q) \vee (p \wedge \neg q)$.

```
1. ¬(p ↔ q) ✓
        /  \
2. ¬p   p
3. q  ¬q
```

Definição 1.22 (Ramo fechado). Um ramo é dito *fechado* se nele existirem uma fórmula p e sua negação $\neg p$ e escreve-se X no final do ramo. Um ramo é aberto quando não é fechado.

Definição 1.23 (Tableau fechado). Um tableau é fechado quando todos os seus ramos forem fechados. Caso contrário, ele é aberto.

Construção de tableaux semânticos

Para utilizar este método para testar a validade de um argumento, constrói-se uma lista composta por suas premissas e pela negação da conclusão. Esta lista compõe a raiz da árvore de refutação que, como toda estrutura de árvore utilizada na computação, cresce para baixo. O processo de construção dos ramos da árvore é feito pela aplicação das regras descritas anteriormente para a construção de novos nós da árvore. O processo termina quando todas as fórmulas forem apenas sentenças atômicas simbolizadas ou suas negações, ou quando forem encontradas contradições.

Se forem encontrados apenas valores falsos, ***F***, em todos os ramos da árvore, significa que a tentativa de refutação falhou e isto significa que o argumento é válido. Se em algum nó da árvore não tiver valor falso, este argumento deve ser refutado, ou seja, não é válido. Vejamos alguns exemplos.

Exemplo 1.31*.* Analisar a validade do argumento $p \wedge q \models \neg\neg p$, usando o processo de construção de árvore de refutação ou tableau semântico.

Para isso, serão desenvolvidos os seguintes passos:

Constrói-se a lista das premissas e da negação da conclusão:

1. $p \land q$
2. $\neg\neg\neg p$

- Sabemos que a sentença $p \land q$ só é verdadeira se p e q forem, ambas, verdadeiras. Deste modo, $p \land q$ pode ser substituída por p e q, gerando as linhas 3 e 4, respectivamente. Neste caso, a sentença $p \land q$ é marcada com o sinal ✓ para indicar que ela não deve ser mais utilizada na construção da árvore.

1. $p \land q$ ✓
2. $\neg\neg\neg p$
3. p
4. q

- Sabemos também que $\neg\neg\neg p$ é equivalente a $\neg p$. Assim ela será marcada e substituída por esta.

1. $p \land q$ ✓
2. $\neg\neg\neg p$ ✓
3. p
4. q
5. $\neg p$

- Neste ponto, observa-se que a árvore está composta apenas pelas sentenças atômicas, p e q, e pela negação de p. Isto significa que o processo de construção da árvore de refutação acabou. Observa-se também que as sentenças das linhas 3 e 5 formam uma contradição. Este fato será denotado pela letra ***X*** na próxima linha, 6.

1. $p \land q$ ✓
2. $\neg\neg\neg p$ ✓
3. p

4. q

5. $\neg p$

6. ***X***

Isto significa que a nossa tentativa de refutação da sentença falhou, ou seja, o argumento é válido.

***Exemplo 1.32*. ** Analisar, usando o processo de árvore de refutação, a validade do argumento $p \vee q, \neg p \models q$.

- Construindo a árvore de refutação com a lista de premissas e com a negação da conclusão, tem-se:

1. $p \vee q$
2. $\neg p$
3. $\neg q$

- Sabemos que $p \vee q$ será uma sentença verdadeira se p for verdadeira ou se q for verdadeira. Para representar isto, a fórmula será marcada e será substituída pelos dois ramos p e q.

1. $p \vee q$ ✓
2. $\neg p$
3. $\neg q$
  /   \
4. p q

Neste ponto, o processo de construção da árvore terminou porque a árvore só contém variáveis proposicionais ou suas negações. No ramo da árvore formado pelas linhas 2 e 4, ($\neg p \wedge p$) encontramos uma fórmula ***F*** e no ramo formado pelas linhas 3 e 4 ($\neg q \wedge q$) encontramos outra contradição. Isto significa que a nossa tentativa de refutar o argumento falhou nos dois ramos da árvore. Portanto ele é verdadeiro. Isto será expresso escrevendo-se um X no final de cada ramo da lista, gerando a linha 5 e fechando os dois ramos da árvore

1. $p \lor q$ ✓
2. $\neg p$
3. $\neg q$

   / \

4. p q
5. X   X

Exemplo 1.33. Analisar a validade do argumento $p \lor q, p \models \neg q$.

- Vamos construir a lista das premissas e da negação da conclusão.

1. $p \lor q$
2. p
3. $\neg\neg q$

- A dupla negação $\neg\neg q$ deve ser substituída por q e marcada.

1. $p \lor q$
2. p
3. $\neg\neg q$ ✓
4. q

- Como no exemplo anterior, a sentença $p \lor q$ será marcada e substituída:

1. $p \lor q$ ✓
2. p
3. $\neg\neg q$ ✓
4. q

   / \

5. p q

Neste ponto, a construção da árvore termina e não foi encontrada qualquer contradição. Isto significa que o argumento não é válido.

***Exemplo 1.34*.** Vamos construir um exemplo mais completo. Sejam as seguintes sentenças:

- Ronaldo é determinado.
- Ronaldo é inteligente.
- Se Ronaldo é inteligente e atleta, ele não é um perdedor.
- Ronaldo é um atleta se é um amante do futebol.
- Ronaldo é amante do futebol se é inteligente.

Usando o método do tableau semântico ou árvore de refutação, a sentença "***Ronaldo não é um perdedor***" é uma conseqüência lógica dos argumentos acima?
Vamos considerar as seguintes correspondências:

p : Ronaldo é determinado.
q : Ronaldo é inteligente.
r : Ronaldo é atleta.
s : Ronaldo é um perdedor.
t : Ronaldo é amante do futebol.

A partir destas correspondências, os argumentos são traduzidos para a Lógica Proposicional da seguinte forma:

$$h = (p \wedge q \wedge ((q \wedge r) \rightarrow \neg s) \wedge (t \rightarrow r) \wedge (q \rightarrow t)) \rightarrow \neg s$$

Deve-se verificar se h é, ou não, uma tautologia. Em outras palavras, deve=se verificar se $\vdash h$. Vamos construir este tableau.

1	p	premissa
2	q	premissa
3	$(q \wedge r) \rightarrow \neg s$	premissa

4	$t \rightarrow r$	premissa
5	$q \rightarrow t$	premissa
6	$\neg\neg s$	negação da conclusão
7	s	idempotência em 6
8	$q \rightarrow r$	silogismo hipotético 5,4
9	$s \rightarrow \neg (q \wedge r)$	por contraposição em 3
10	$\neg(q \wedge r)$	por MP 7,9
	/ \	
11	$\neg q$ $\neg r$	por R6 em 10
12	**X**	contradição 2,11
13	$\neg q$ r	pela def de $\rightarrow$ em 8
14	**X** **X**	contradição 2,13 e 11,13

Como o tableau é fechado, ou seja, só existem sentenças atômicas ou suas negações e contradições, então h é uma tautologia porque a suposição de que ela não era válida nos conduziu a uma contradição. Portanto é verdade que **¬**s é uma conseqüência lógica dos argumentos dados, ou seja, Ronaldo não é um perdedor.

O caminho das pedras

Não existe um algoritmo perfeito para a construção de tableaux semânticos. No entanto, algumas regras servem para facilitar a construção de tais árvores. São elas:

- ✓ Modus ponens é a regra de inferência mais intuitiva. Tente usá-la muitas vezes
- ✓ Fbfs da forma $\neg(P \wedge Q)$ ou $\neg(P \vee Q)$ dificilmente são úteis em uma sequência de provas. Use o teorema de De Morgan para convertê-las em $\neg P \vee \neg Q$ e $\neg P \wedge \neg Q$, respectivamente, separando as componentes
- ✓ Fbfs da forma $P \vee Q$ dificilmente são úteis em uma sequência de provas, já que não implicam P nem Q. Use a dupla negação para converter $P \vee Q$ em $\neg(\neg P) \vee Q$ e depois use a regra do condicional para obter $\neg P \rightarrow Q$

- ✓ Aplicar primeiro as regras que não bifurquem, ou seja, adie as bifurcações o máximo possível.

RESUMO

Esta unidade consistiu de um estudo inicial envolvendo a Lógica Proposicional e seus fundamentos e propriedades. Além da teoria, vários exemplos e exercícios resolvidos e foram mostrados para que o leitor pudesse sedimentar e entender melhor os conceitos e definições envolvidas. Vários exercícios também foram propostos para este fim.

De posse deste conhecimento, o leitor está capacitado a entender o relacionamento existente entre a Lógica e outras teorias matemáticas, um tema a ser visto na próxima unidade.

Exercícios

1. Demonstrar a validade do argumento $\neg p \rightarrow q$, $q \rightarrow \neg r$, $r \vee s$, $\neg s \rightarrow p$.

2. Considere o seguinte argumento: "*Se as taxas de juros caírem, o mercado imobiliário vai melhorar. A taxa federal de descontos vai cair ou o mercado imobiliário não vai melhorar. As taxas de juros vão cair. Por tanto, a taxa federal de descontos vai cair*." Verifique, usando prova direta, se este argumento é, ou não, válido.

3. Considere o seguinte argumento: "*Meu cliente é canhoto, mas se o diário não tiver sumido, então meu cliente não é canhoto. Portanto, o diário sumiu*". Verifique, usando prova direta, se este argumento é, ou não, válido.

4. Considere o seguinte argumento: "Descosos são cor de rosa mas, se Gincoso não gostar de pereques, então

descosos não são cor de rosa. Portanto, Gincoso não gosta de pereques". Verifique, usando prova direta, se este argumento é, ou não, válido e compare este argumento com o do problema anterior.

5. Sejam as seguintes sentenças:

Raquel é rica.
Raquel é inteligente.
Se Raquel é inteligente e bonita, ela vai ser freira.
Raquel é bonita se ela freqüenta a Academia.
Raquel freqüenta a Academia se é inteligente.

Usando o método do tableau semântico ou árvore de refutação, a sentença "Raquel vai se casar" é uma conseqüência lógica dos argumentos acima?

6. Considere as três afirmações a seguir:

H_1: Se Alírio toma vinho e o vinho está ruim, ele fica com ressaca.

H_2: Se Alírio fica com ressaca, então ele fica triste e vai para casa.

H_3: Se Alírio vai ao seu encontro romântico com Virgínia então ele fica triste e vai para casa.

Suponha que as três afirmações anteriores são verdadeiras. A partir deste fato, quais das afirmações a seguir também são verdadeiras?

G_1: Se Alírio toma vinho e este está ruim, então ele perde seu encontro romântico com Virgínia.

G_2: Se Alírio fica com ressaca e vai para casa, então ele não perde seu encontro romântico co Virgínia.

G_3: Se o vinho está ruim, então Alírio não o toma ou ele não fica com ressaca.

G_4: Se o vinho está ruim ou Alírio fica com ressaca, então ele fica triste.

G_5: Se Alírio toma vinho e vai para casa, então ele não fica triste se o vinho está ruim.

SAIBA MAIS

Existem muitos bons textos e alguns deles estão listados na Bibliografia colocada ao final da Unidade 2. Outros estão na Internet à disposição . Estes estão listados a seguir.

REFERÊNCIAS NA WEB

www.ufpi.br/uapi **(A página da UAPI)**

www.uab.gov.br **(O Site da Universidade Aberta do Brasil-UAB)**

www.seed.mec.gov.br **(A Homepage da Secretaria de Educação a Distância do MEC - SEED)**

www.abed.org.br **(O Sitio da Associação Brasileira de Educação a Distância - ABED)**

http://pt.wikipedia.org/ **O site da Wikipedia.**

www.pucsp.br/~logica/

www.inf.ufsc.br/ine5365/introlog.html

www.gregosetroianos.mat.br/logica.asp

Unidade 2

A Lógica e outras Teorias

RESUMO

O objetivo principal desta Unidade é fazer uma comparação entre a Lógica e outras teorias já conhecidas. Entre elas, a Teoria dos Conjuntos e a Álgebra de George Boole. Estas teorias são bastante utilizadas em todos os campos do conhecimento e devem ser justificadas as suas estruturas à luz da Lógica.

Outro tema bastante utilizado diz respeito com a prova de propriedades utilizando o princípio da indução finita. Este tipo de prova tem sido utilizado em muitos casos. Faz-se necessário, no entanto, verificar a validade e corretude deste tipo de prova e verificar por que este princípio realmente tem sua validade.

. A unidade também contém vários exemplos, e exercícios resolvidos tentando proporcionar ao leitor o entendimento pleno dos conceitos envolvidos, além de serem propostos vários exercícios para sedimentar a teoria apresentada.

A forma de apresentação utilizada é de acordo com o exigido para o ensino à distância, ou seja, tendo em vista sempre esta nova modalidade de ensino.

SUMARIO

A LÓGICA E OUTRAS TEORIAS

1.10 Introdução

Esta unidade se faz necessária face as diversas teorias conhecidas e utilizadas com freqüência em diversos campos do conhecimento. Por exemplo, a teoria dos conjuntos é bastante conhecida e utilizada na Matemática. Nesta unidade ela será vista analisando a sua consistência à luz da Lógica.

Uma outra teoria bastante utilizada na Eletrônica e em outras ciências diz respeito a Álgebra de Boole. Este tema tem interesse para os estudantes de Computação dada a sua utilização nos circuitos digitais que são os elementos que compõem todos os computadores eletrônicos.

Não menos importante que estes dois temas, o princípio da indução finita tem sido mostrado como técnica de prova de muitas propriedades matemáticas. É necessário entender porque este princípio funciona corretamente, bem como utilizá-lo em diversas situações. Estes temas são o objeto de estudo desta unidade.

1.11 O Cálculo Proposicional e a Teoria dos Conjuntos

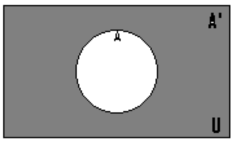
É importante notar a existência de uma relação intrínseca entre o cálculo proposicional e a Teoria dos Conjuntos. Esta relação permite que a verificação dos valores verdade de algumas sentenças da Lógica Proposicional seja feita utilizando técnicas da Teoria dos Conjuntos. Esta possibilidade pode facilitar as demonstrações, uma vez que a Teoria dos Conjuntos já é bastante conhecida e suas técnicas, normalmente, já são dominadas pelas pessoas que estão iniciando seus estudos sobre a Lógica.

Um exemplo disto é a verificação gráfica de propriedades de operações da Lógica Proposicional usando Diagramas de Euler-Venn (John Venn 1834-1923), apesar de observar que esta metodologia não deve ser considerada como instrumento rigoroso de prova. Mesmo assim, estes diagramas podem ser utilizados como ferramentas de verificação visual e já são bastante conhecidos. Isto significa que quaisquer sentenças do Cálculo Proposicional têm expressões correspondentes na Teoria dos conjuntos e estas podem ser representadas como diagramas de Euler-Venn. Estas correspondências são verificadas utilizando as mesmas regras mostradas para a obtenção das formas normais conjuntivas e disjuntivas, mostradas anteriormente, na Unidade 1. Estas correspondências são especificadas da seguinte forma:

- a negação de uma sentença A da Lógica, ou seja $\neg A$, corresponde ao complemento de A na Teoria dos Conjuntos, ou seja, $\bar{A}$;
- a conjunção de duas sentenças A e B da Lógica, ou seja $A \land B$, corresponde à interseção dos conjuntos A e B na Teoria dos Conjuntos, ou seja, $A \cap B$;
- a disjunção de duas proposições A e B da Lógica, ou seja $A \lor B$, corresponde à união dos conjuntos A e B na Teoria dos Conjuntos, ou seja, $A \cup B$;
- a sentença $A \rightarrow B$ da Lógica não tem correspondente direto na Teoria dos Conjuntos, mas pode ser substituída pela sentença $\neg A \lor B$ e esta tem correspondência na Teoria dos Conjuntos;

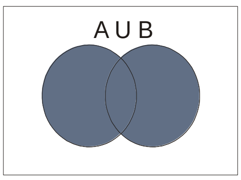
- a sentença A ↔ B da Lógica também não tem correspondente na Teoria dos Conjuntos, mas pode ser substituída pela sentença (¬A V B) Λ (A V ¬B), e esta tem correspondência na Teoria dos Conjuntos;
- as negações que precedem os parênteses nas sentenças da Lógica podem ser substituídas por outras sentenças também da Lógica, mas com correspondentes na Teoria dos Conjuntos. Estas substituições são:
  - ¬ (A Λ B) por ¬A V ¬B e
  - ¬ (A V B) por ¬A Λ ¬B;
- as negações múltiplas ¬(¬A) da Lógica podem ser substituídas por A e
- elimina-se o alcance dos conectivos Λ e V da Lógica, substituindo-se
  - A V (B Λ C) por (A V B) Λ (A V C) e
  - A Λ (B V C) por (A Λ B) V (A Λ C)

Desta forma, podemos ter as seguintes correspondências para as áreas hachuradas:



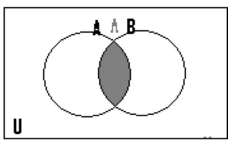
- **Negação (¬A)**

A área hachurada corresponde ao complemento de A, ou seja, Ā, que corresponde a ¬A da Lógica.



Disjunção (A V B)

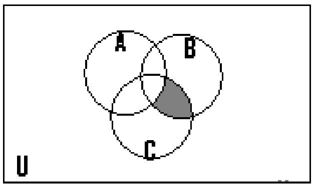
A área hachurada corresponde à união A U B, que corresponde a A V B da Lógica.



Conjunção (A Λ B)

A área hachurada corresponde à interseção A ∩ B, que corresponde a A Λ B da Lógica.

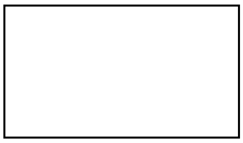
Exemplo 2.1*.* Seja o diagrama de Euler-Venn mostrado ao lado:



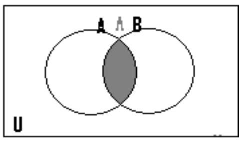
A área hachurada corresponde a $(B \cap C) \cap \bar{A}$ na Teoria dos Conjuntos. Pelas regras dadas anteriormente, esta área corresponde a $(B \land C) \land \neg A$ da Lógica e que corresponde a $\neg(\neg(B \land C) \lor A)$ que, por sua vez, corresponde a $\neg((B \land C) \to A)$.

Como decorrência direta destas relações, podemos verificar que os seguintes resultados são verdadeiros:

- **Tautologia.** Em uma tautologia, a área hachurada é o conjunto Universo U. Por exemplo, a sentença $A \lor \neg A$ da figura ao lado é uma tautologia.



- **Contradição.** Uma contradição é representada pela ausência de área hachurada. Por exemplo, a sentença $A \land \neg A$ da figura ao lado é uma contradição.



- **Contingência.** Uma contingência é representada por uma área que apresenta uma parte hachurada e outra não hachurada. Por exemplo, a sentença $A \land B$ da figura ao lado representa uma contingência

1.12 Cálculo Proposicional e a Álgebra de Boole

Augustus de Morgan

Nesta seção, será será analisada a relação existente entre o Cálculo Proposicional e a Álgebra booleana, desenvolvida por George Boole em 1948 Esta relação é muito importante para a fundamentação da Eletrônica Digital responsável pela construção dos computadores eletrônicos.

Uma Álgebra Booleana é uma sêxtupla B da forma B = {A, +, ., ‘, 0, 1}, onde A é um conjunto de variáveis, +, e . são operações binárias entre os elementos de A, ‘ é uma operação unária em A e os elementos 0 e 1 são elementos distintos de B, onde são verdadeiras as seguintes propriedades mostradas na Tabela 1.11.

Tabela 1.11. Propriedades da Álgebra Booleana.

PROPRIEDADE	OPERAÇÃO	OPERAÇÃO DUAL
Associatividade	**(p+q)+r = p+(q+r)**	**(p.q).r = p.(q.r)**
Comutatividade	**p+q = q+p**	**p.q = q.p**
Idempotência	**p+p = p**	**p.p = p**
Absorção	**(p.q)+p = p**	**(p+q).p = p**
Distribuição	**p+(q.r) = (p+q).(p+r)**	**p.(q+r) = (p.q).+ (p.r)**
Prop de 0	**p+0 = p**	**p.1 = p**
Prop de 1	**p+1 = 1**	**p.0 = 0**
Complemento	**p+p’ = 1**	**p.p’ = 0**

Na Tabela 1.11 anterior, cada operação da coluna Operação dual pode ser obtida da coluna Operação substituindo-se a operação + por . e . por +, além de trocar o 0 por 1 e o 1 por 0, sendo esta a definição de operação dual de uma outra.

As seguintes observações da Álgebra de Boole devem ser atendidas para tornar as operações nesta teoria mais fáceis de serem realizadas e compreendidas:

- a operação p . q normalmente é denotada por pq,
- a operação p + q é a disjunção de p com q,
- a operação pq é a conjunção de p com q,
- p’ é o complemento de p,
- 0 é o elemento zero (complemento de 1) e
- 1 é o complemento de 0.

Exemplo 2.2. As seguintes expressões são equivalentes, na Álgebra Booleana e na Lógica Proposicional: $(p' + (qr))' \equiv \neg(\neg p \vee (q \wedge r))$.

Uma expressão booleana representa uma função onde as variáveis são os parâmetros e a expressão é o resultado da função. As expressões booleanas podem ser transformadas em expressões booleanas mais simples para serem implementadas como circuitos eletrônicos. O objetivo é conseguir circuitos mais simples e portanto mais baratos e menores.

Exemplo 2.3. Simplificar a sentença proposicional $(\neg p \wedge \neg q \wedge r) \vee (\neg p \wedge q \wedge \neg r) \vee (\neg p \wedge q \wedge r) \vee (p \wedge q \wedge \neg r) \vee (p \wedge q \wedge r)$.

A expressão correspondente a esta na Álgebra de Boole é

$p'q'r + p'qr' + p'qr + pqr' + pqr$

$= p'q'r + p'q(r' + r) + pq(r' + r)$ pela propriedade da distribuição

$= p'q'r + p'q + pq$ pela prop do complemento

$=p'(q'r + q) + pq$ pela propriedade da distribuição

=p’(r + q) + pq	pela propriedade da absorção
=p’r + p’q + pq	pela propriedade da distribuição
=p’r + (p’ + p)q	pela propriedade da distribuiçào
=p’r + q	pela prop do complemento

A expressão acima tem correspondente na Lógica que é $(\neg p \wedge r) \vee q$, significando que ambas realizam a mesma função. A sentença proposicional inicial que era bem maior e mais complexa foi transformada em outra expressão bem menor e mais simples e, exatamente por este motivo, pode ser implementada de forma bem mais econômica. Esta técnica é objeto de estudo dos sistemas digitais. O objetivo aqui é apenas mostrar como a Álgebra booleana se fundamenta e é justificada pela Lógica.

Existem muitas outras técnicas que podem ser utilizadas na simplificação de funções na Álgebra de Boole.

1.13 O Principio da Indução Finita e a Lógica

O princípio da indução finita é um dos principais métodos utilizados na demonstração de resultados em diversas áreas da Matemática e da Teoria da Computação. Na Matemática, ele é utilizado na demonstração de várias propriedades dos números e na Ciência da Computação é empregado para demonstrar resultados na área das Linguagens Formais, na Teoria dos Algoritmos, na Teoria dos Códigos e na Lógica. Esta é a principal motivação da inclusão deste tema em nosso estudo, uma vez que ele é bastante utilizado pelos profissionais destas áreas do conhecimento humano e deve ser analisada a relação existente entre ele e a Lógica, justificando sua adoção como metodologia de prova.

Necessidade e suficiência de condições

Estes dois termos são também muito utilizados em demonstrações na Lógica e na Matemática para denotar a implicação e a equivalência que são temas já conhecidos e vamos introduzi-los através de um exemplo, para melhor compreensão.

Considerando o conjunto dos professores da Universidade, sabemos que, em termos funcionais, existem os professores *Auxiliares*, os professores *Assistentes*, os professores *Adjuntos* e os professores *Associados*. Vamos analisar em que condições um professor da Universidade se torna um professor Associado. Vamos analisar que condições são exigidas a um profissional para que ele se torne um professor Associado, além de seu desejo pessoal, é claro. A primeira condição necessária é que ele tenha cursado algum curso superior. Este fato pode ser representado na Lógica por:

$$\text{associado} \rightarrow \text{graduado}$$

Isto significa que se um profissional é professor Associado então ele é graduado em algum curso superior. No entanto, apenas ser graduado não é uma condição suficiente para ser um professor Associado. Para isso é também necessário que o profissional tenha realizado o curso de Mestrado. Isto significa que se alguém é professor Associado ele deve ser graduado e também ser mestre. Isto é representado na Lógica por:

$$\text{associado} \rightarrow \text{graduado} \land \text{mestre}$$

As duas condições são necessárias, mas ainda não são suficientes para ser um professor Associado,. Além dessas duas, é necessário também que o profissional tenha feito um curso de Doutorado, ou seja,

$$\text{associado} \rightarrow \text{graduado} \wedge \text{mestre} \wedge \text{doutor}$$

Estas condições são necessárias, mas ainda não são suficientes para que um profissional se torne um professor Associado, Para tal é necessário que ele se submeta a um Concurso Público de Provas e Títulos. Isto implica que

$$\text{associado} \rightarrow \text{graduado} \wedge \text{mestre} \wedge \text{doutor} \wedge \text{concursado}$$

Apenas os professores Adjuntos do nível IV podem se candidatar ao cargo de professor Associado. Isto significa que

$$\text{associado} \rightarrow \text{graduado} \wedge \text{mestre} \wedge \text{doutor} \wedge \text{concursado} \wedge \text{adjunto IV}$$

Estas condições são necessárias, mas ainda não são suficientes. Para que um professor Adjunto IV seja promovido ao cargo de Associado. Além de todas estas condições, ele tem que ser avaliado por uma Comissão para esse fim nomeada. Caso essa avaliação seja aprovada, ele então será promovido ao cargo de professor Associado. Isto significa que

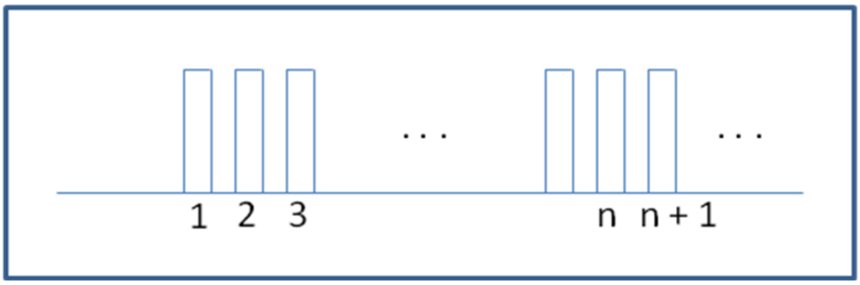
$$\text{associado} \rightarrow \text{graduado} \wedge \text{mestre} \wedge \text{doutor} \wedge \text{concursado} \wedge \text{adjunto IV} \wedge \text{avaliado}$$

Portanto para ser um professor Associado, o profissional tem de ser graduado em algum curso, tem de ter feito Mestrado e Doutorado, ter feito um concurso Público de Provas e Títulos, ser Adjunto IV e ser avaliado por uma Comissão. Neste caso, as condições necessárias são também suficientes, ou seja, o sentido da implicação pode ser invertido.

$$\text{graduado} \wedge \text{mestre} \wedge \text{doutor} \wedge \text{concursado} \wedge \text{adjunto IV} \wedge \text{avaliado} \rightarrow \text{associado}$$

Neste caso, a condição de suficiência é o antecedente da proposição e a de necessidade é o conseqüente.

Exemplo 2.3. Um exemplo, baseado em (Souza, 2002), se refere a um conjunto infinito de pedras de dominó, enfileiradas conforme a figura a seguir.



Os dominós são enumerados e dispostos de forma que se o dominó número n for derrubado para a direita, então o dominó subseqüente (número n + 1) também será derrubado para a direita. Considere a seguinte questão

Que condição é suficiente para que o dominó não caia?

Se qualquer um dos dominós que precedem o dominó número n cair para a direita, então este também será derrubado. Portanto, há várias condições suficientes para que o dominó número n seja derrubado. Uma delas é a seguinte:

Uma condição suficiente para que o dominó número n caia é que o dominó 1 seja derrubado para a direita.

Basta que o dominó número 1 caia para a direita e isto será suficiente para que o mesmo ocorra com o dominó número n. Em uma linguagem da Lógica, isto pode ser representado por

se "<u>*dominó número 1 for derrubado para a direita*</u>" então "<u>*dominó número n irá cair*</u>"

Pode-se verificar, portanto, que em uma Linguagem Lógica o antecedente de uma implicação é uma condição ***suficiente*** para o conseqüente. Considere uma outra questão.

Qual é uma condição necessária para que o dominó número 1 possa ser derrubado para a direita?

Em outras palavras, o que deve ser permitido ocorrer para que o dominó número 1 seja derrubado para a direita? Se não for permitido derrubar o dominó número n, por exemplo, não será possível derrubar o dominó número 1. Portanto a queda do dominó número n deve ser permitida para que o dominó número 1 seja derrubado para a direita. Logo, uma condição necessária para que o dominó número 1 caia para direita é que seja permitida a queda do dominó número n. Considerando a implicação

Se <u>*o dominó número 1 for derrubado para a direita*</u>, então <u>*o dominó número n irá cair.*</u>

O conseqüente da implicação é uma condição necessária para que o antecedente possa ocorrer. Isto é, a condição necessária é aquela sem a qual nada pode ocorrer. A ocorrência da condição necessária no conseqüente deve ser permitida para que o antecedente ocorra.

A condição suficiente é o antecedente da implicação e a condição necessária é o conseqüente.

Considere duas fórmulas p e q tais que p implica q. Por definição, **p implica q** ⇔ para toda interpretação de I, se I[p] = T, então I[q] = T.

Neste caso, I[p] = T é uma condição suficiente para se ter I[q] = T e I[q] = T é uma condição necessária para se ter I[p] = T.

Definição 2.1 (condição suficiente e condição necessária). Dadas duas fórmulas p e q tais que p implica q, então p é uma condição suficiente para q e q é uma condição necessária para p.

No caso em que p equivale a q, tem-se que p implica q e q implica p. Logo, p é uma condição necessária e suficiente para q. Da mesma forma, q também é uma condição necessária e suficiente para p.

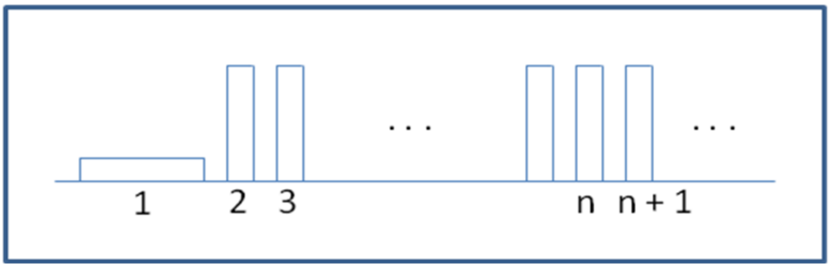
O princípio da indução

Consideremos novamente o conjunto infinito de dominós, visto anteriormente, numerados e enfileirados, como mostrado na figura a ele associada. Um conjunto de condições suficientes para que todos os dominós sejam derrubados é indicado a seguir. Observe que existem outros conjuntos de condições suficientes. A definição a seguir é denominada primeira forma do princípio da indução finita.

***Definição* 2.2** (condições suficientes, primeira forma).

1. *Condição básica*. O dominó número 1 é derrubado para a direita.
2. *Condição indutiva*. Seja n um número arbitrário. Se o dominó número n for derrubado para a direita, então o dominó número (n + 1) também será derrubado para a direita.

Deve ser observado que as duas condições acima são imprescindíveis para garantir que todos os dominós sejam derrubados. Se apenas a condição básica 1 for verdadeira não se pode garantir que todos os dominós sejam derrubados. Pode ocorrer, por exemplo, a situação descrita na figura a seguir.



Neste caso, o dominó número 1 é derrubado, mas ele está longe do dominó número 2, que não é derrubado. Desta forma, apenas o dominó número 1 é derrubado. O mesmo ocorre quando apenas a condição indutiva 2 é verdadeira. Neste caso, as distâncias entre os dominós é tal que se um dominó qualquer for derrubado, então o subseqüente também será derrubado. Entretanto, se o primeiro dominó não for derrubado, não se pode garantir a derrubada dos demais. Além das condições básica e indutiva, indicadas na primeira forma em 1 e 2, há outros tipos de condições que também são suficientes para a derrubada de todos os dominós. Um outro conjunto de condições suficientes é indicado a seguir.

***Definição* 2.3** (condições suficientes, segunda forma).

1. *Condição básica*. O dominó número 1 é derrubado para a direita.
2. *Condição indutiva*. Seja n um número arbitrário. Se todos os dominós até o número n forem derrubados para a direita, então o dominó número (n + 1) também será derrubado para a direita.

Observe que as condições básicas da primeira e da segunda forma coincidem. Entretanto, a condição indutiva é ligeiramente modificada. Na segunda forma, se todos os dominós até o número n forem derrubados, então o dominó número (n + 1) também será derrubado. Na primeira forma, é considerada apenas a derrubada do dominó número n, o que determina a derrubada do dominó número (n + 1). Por outro lado, na segunda forma, o que provoca a derrubada do dominó número (n + 1) é a queda dos dominós de 1 a n.

O princípio da indução finita possui duas formas correspondentes às condições suficientes consideradas nesta seção, sendo a primeira conhecida como "*primeira forma do princípio da indução finita*", algumas vezes conhecida como "*princípio da indução fraca*" e a segunda conhecida como "*segunda forma do princípio da indução finita*", também conhecida como "*princípio da indução forte*".

Princípio da indução fraca

Já foi aqui afirmado que o princípio da indução finita é bastante utilizado em provas matemáticas, na Lógica e na Teoria da Computação. Vamos anunciá-lo formalmente:

Definição 2.4 (primeira forma do princípio da indução finita). Suponha que para cada número natural n, $n \geq 1$, seja feita a assertiva A(n). Além disso, suponha que seja possível demonstrar as duas propriedades a seguir:

1. *Base da indução*. A assertiva A(1) é verdadeira.
2. *Passo da indução*. Para cada número natural n ≥ 1, se A(n) for verdadeira, então A(n+1) também é verdadeira.

Conclui-se que a assertiva A(n) é verdadeira para todo número natural n.

As propriedades 1 e 2 deste princípio de indução finita correspondem, respectivamente, às condições básica e indutiva vistas na seção anterior, sendo denominadas por *base da indução* e *passo da indução*.

Voltando ao problema dos dominós, visto anteriormente, pode-se analisar que:

1. O dominó número 1 é derrubado, ou seja, A(1) é verdadeira.
2. Para cada natural n ≥ 1, se o dominó de número n for derrubado, ou seja, se A(n) for verdadeira, então A(n+1) também será, ou seja, o dominó de número n+1 também será derrubado.

Se os fatos acima forem verdadeiros, então A(n) é verdadeira para todo n natural, ou seja todos os dominós serão derrubados.

Exemplo 2.4. Demonstrar que a igualdade (1+2+. .+ n) = n(n + 1)/2 é válida para todo número natural n.

Para isto, teremos que verificar se a propriedade é verdadeira para o caso base, A(1), e para o caso indutivo, A(n+1), utilizando o fato de que A(n) é verdadeira, ou seja, utilizando A(n) como hipótese de indução.

Seja $A(n) = \{(1+2+...+n) = n(n+1)/2\}$. Vamos verificar se ela verdadeira para todo número natural n.

1. Caso base A(1): $1 = 1(1+1)/2$. Logo, A(1) é verdadeira.
2. Passo indutivo A(n+1): $(1 + 2 + ... + n + (n + 1)) = (1 + 2 + ... + n) + (n + 1)$. Utilizando a hipótese de indução e substituindo na igualdade acima, temos: $(1+2+...+n) + (n+1) = n(n+1)/2 + (n+1) = n(n+1)/2 + 2(n+1)/2 = (n+1)(n+2)/2$.

`

Assim, a assertiva é verdadeira para o caso base e para o passo indutivo. Logo, ela é verdadeira para todo número natural n.

Princípio de indução forte

A segunda forma do princípio da indução finita, também conhecido como princípio de indução forte, é equivalente à primeira forma. No entanto, ela é mais aceita por alguns pesquisadores da Lógica que insistem em negar a primeira forma. Vejamos a sua definição.

Definição 2.5. (segunda forma do princípio da indução finita). Suponha que para cada número natural n, $n \geq 1$, seja feita a assertiva A(n). Além disso, suponha que seja possível demonstrar as duas propriedades a seguir:

1. *Base da indução*. A assertiva A(1) é verdadeira.
2. *Passo da indução*. Para cada número natural $n \geq 1$, se A(k) for verdadeira para todo k, $1 \leq k \leq n$, então A(n+1) também é verdadeira.

Conclui-se que a assertiva A(n) é verdadeira para todo número natural n.

De forma análoga ao que foi feito na primeira forma, as propriedades 1 e 2 são denominadas, respectivamente, por base da indução e passo da indução.

Exemplo 2.5. Seja a proposição: todo número natural $n \geq 2$ ou é primo ou é um produto de números primos.

1. *Caso base*: P(2) é verdade, já que 2 é primo.
2. *Passo indutivo*: A hipótese de indução, dada por P(x), é válida para $2 \leq x < n$. A tese a ser verificada é P(n).

Vejamos dois casos:

a. Se n for primo, então a tese é válida.
b. Se n não for primo, $n = n_1 * n_2$, com $n_1 < n$ e $n_2 < n$, já que se n não for primo ele é divisível por algum número natural diferente de 1. Pela hipótese de indução, $P(n_1)$ e $P(n_2)$, ou seja a propriedade é válida para n_1 e para n_2, já que esta é a hipótese de indução

Conclusão: como o caso base e o passo indutivo são verdadeiros, então a propriedade é verdadeira para todo número natural $n \geq 2$..

Por que o princípio de indução finita funciona?

O princípio da indução, na primeira e segunda formas, é expresso como a implicação, ou seja:

[Base + Passo da indução] → [A(n) é verdadeira para todo n]

Admitir o princípio da indução finita, significa aceitar como válida a implicação acima. Considerando tal implicação como válida, para demonstrar que [A(n) é verdadeira para todo n] basta demonstrar que [Base + Passo da indução] também é verdadeira. Isto ocorre porque se a implicação é válida e a base e o passo da indução são verdadeiros, então necessariamente a afirmação “A(n) é verdadeira para todo n” também é verdadeira. Observe que é impossível se ter:

[Base + Passo da indução] → [A(n) é verdade para todo n].
T T F

onde o antecedente seja verdadeiro, a implicação seja também verdadeira e o consequente seja falso. Aceitar o princípio de indução finita corresponde à aceitação da validade desta implicação.

Deve, no entanto, ser observado que tal validade pode ser questionada, porque a base e o passo indutivo consideram apenas valores finitos para n, como A(1) e A(n) → A(n+1) e o consequente considera qualquer valor de n. Neste sentido, o princípio corresponde a concluir algo infinito a partir de premissas finitas.

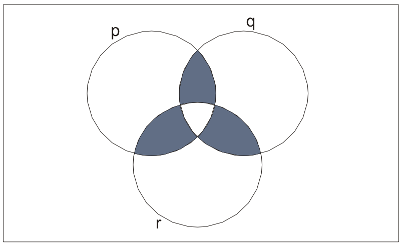
1.14 RESUMO

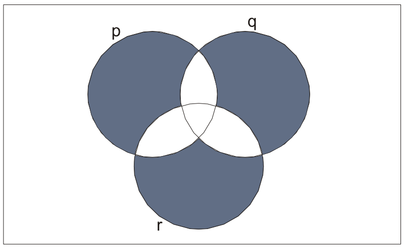
Esta unidade consistiu de um estudo da Lógica como fundamento de algumas teorias utilizadas em várias áreas do conhecimento humano, como a Matemática, a Eletrônica Digital e a Teoria da Computação.

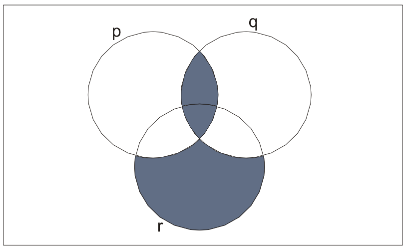
O objetivo foi justificar algumas propriedades destas teorias, tendo a Lógica como fundamentação, para que o leitor tenha certeza de que os resultados obtidos nestas teorias são válidos. Outras áreas de aturação também são campos de atuação da Lógica, por exemplo, a Filosofia. Este estudo está fora do escopo deste estudo, mas o leitor fica convidado a estudar e pesquisar este tema na bibliografia indicada.

Com o domínio deste conhecimento, o leitor está capacitado a entender outros conceitos da Lógica que complementam a Lógica Proposicional, conhecido como Lógica de Predicados, o tema objeto de estudo da próxima unidade.

7. Dados os diagramas de Venn abaixo, encontre:
   a. A expressão booleana que os representa.
   b. A expressão da Teoria dos Conjuntos que os representa.
   c. A expressão proposicional que os represente.







8. Dadas as expressões da Lógica Proposicional a seguir, encontre:
   a. As expressões correspondentes na Lógica de Boole.
   b. As expressões correspondentes na Teoria dos Conjuntos.
   c. As representações em diagramas de Vem.

   i. $(p \rightarrow q \land r) \lor (p \land q \leftrightarrow r)$
   ii. $(p \land q \leftrightarrow r) \land (p \rightarrow q \lor r)$

9. Apesar da técnica de verificação da validade de fbfs ser bastante intuitiva e prática, ela padece de uma limitação que a torna utilizável em apenas alguns casos. Analise que limitação é esta e discuta possíveis soluções.

10. Usando o Princípio de Indução Finita, mostre que as seguintes proposições são verdadeiras para todo número inteiro positivo n:
    a. $2^0 + 2^1 + 2^2 + . . . + 2^n = 2^{n+1} – 1$
    b. 3 divide $n^3 – n$
    c. Para $n \geq 4$, $n! > 2n$
    d. $1 + 3 + 5 + . . . + (2n – 1) = n^2$
    e. 3 divide $2^{2n} – 1$
    f. $2 + 6 + 10 + . . . + (4n – 2) = 2n^2$
    g. $1 + 5 + 9 + . . . + (4n – 3) = n(2n – 1)$
    h. $4 + 10 + 16 + . . . + (6n – 2) = n(3n + 1)$
    i. $1^3 + 2^3 + 3^3 + . . . + n^3 = n^2(n + 1)^2/4$
    j. $1^2 + 3^2 + . . . + (2n – 1)^2 = n(2n – 1)(2n + 1)/3$
    k. Se o quadrado de um número inteiro n for ímpar, então n é ímpar.
    l. Mostre que todo número natural $n \geq 1$ é um número primo ou um múltiplo de primos

1.16 SAIBA MAIS

Existem muitos bons textos e alguns deles estão listados na Bibliografia colocada ao final da Unidade 2. Outros estão na Internet à disposição . Estes estão listados a seguir.

1.17 WEB-BIBLIOGRAFIA

www.ufpi.br/uapi (A página da UAPI)
www.uab.gov.br (O Site da Universidade Aberta do Brasil- UAB)
www.seed.mec.gov.br (A Homepage da Secretaria de Educação a Distância do MEC - SEED)
www.abed.org.br (O Sitio da Associação Brasileira de Educação a Distância - ABED)
http://pt.wikipedia.org/ O site da Wikipedia.
www.pucsp.br/~logica/
www.inf.ufsc.br/ine5365/introlog.html
www.gregosetroianos.mat.br/logica.asp

Unidade 3

Lógica de Predicados

RESUMO

O objetivo principal desta unidade é apresentar os principais conceitos e estruturas da Lógica de Predicados bem como ela pode ser utilizada no ordenamento do raciocínio humano, na busca de soluções para os problemas que ocorrem na natureza e que não podem ser simbolizados utilizando apenas a Lógica Proposicional.

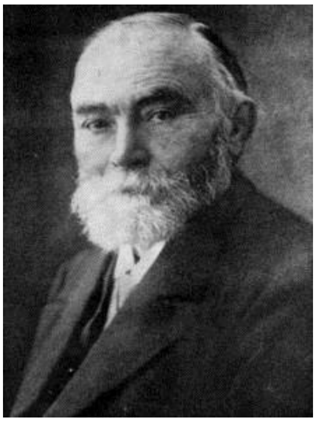
Na unidade é mostrada a formulação correta de argumentos utilizando os quantificadores universal e existencial, bem como as metodologias utilizadas na verificação da validade de argumentos, notadamente o uso de tableaux semânticos

A unidade contém vários exemplos e exercícios resolvidos, tentando proporcionar ao leitor o entendimento pleno dos conceitos envolvidos, além de serem propostos vários exercícios visando sedimentar a teoria apresentada.

A forma de apresentação utilizada é de acordo com o exigido para o ensino à distância, ou seja, tendo em vista sempre esta nova modalidade de ensino.

SUMARIO

LÓGICA DE PREDICADOS



Gottlob Frege

Gottlob Frege em sua Conceitografia (Begriffsschrift), descobriu uma maneira de reordenar várias sentenças para tornar sua forma lógica clara, com a intenção de mostrar como as sentenças se relacionam em certos aspectos. Antes de Frege, a Lógica formal não obteve sucesso além do nível da Lógica Proposicional: ela podia representar a estrutura de sentenças compostas de outras sentenças, usando palavras como "e", "ou" e "não", mas não podia quebrar sentenças em partes menores. Não era possível mostrar como "Cavalos são animais" leva a concluir que "Partes de cavalos são partes de animais".

A Lógica Proposicional explica como funcionam palavras como "e", "mas", "ou", "não", "se-então", "se e somente se", e "nem-ou". Frege expandiu a Lógica para incluir palavras como "todos", "alguns", e "nenhum". Ele mostrou como introduzir variáveis e quantificadores para reorganizar sentenças.

Neste novo tipo de Lógica, a Lógica de Predicados, a sentença "Todos os humanos são mortais" se torna "Para todo x, se x é humano, então x é mortal.", o que pode ser escrito simbolicamente como:

$$(\forall \mathbf{x})\ (\mathbf{H(x)} \rightarrow \mathbf{M(x)})$$

A sentença "Alguns humanos são vegetarianos" se torna "Existe algum (ao menos um) x tal que x é humano e x é vegetariano", sendo escrita simbolicamente como:

$$(\exists \mathbf{x})\ (\mathbf{H(x)} \wedge \mathbf{V(x)})$$

Frege trata sentenças simples sem substantivos como predicados. A estrutura lógica na discussão sobre objetos pode ser operada de acordo com as regras da Lógica Proposicional, com alguns detalhes adicionais para adicionar e remover quantificadores. O trabalho de Frege foi um dos que deu início à Lógica formal contemporânea.

Frege adicionou à Lógica Proposicional:

- o vocabulário de quantificadores (o A de ponta-cabeça, e o E invertido) e variáveis;
- uma semântica que explica como as variáveis denotam objetos individuais e como os quantificadores têm algo como a força de "todos" ou "alguns" em relação a esses objetos;
- métodos para usá-los numa linguagem.

Para introduzir um quantificador "todos", assume-se uma variável arbitrária, prova-se algo que deve ser verdadeiro, e então prova-se que não importa qual variável foi escolhida, que aquilo deve ser sempre verdade. Um quantificador "todos" pode ser removido aplicando-se a sentença para um objeto em particular. Um quantificador "algum" (existe) pode ser adicionado a uma sentença verdadeira de qualquer objeto; pode ser removido em favor de um termo sobre o qual você ainda não esteja pressupondo qualquer informação.

Charles Babbage

1.18 Primeiros passos

O principal objetivo do estudo da Lógica na computação é encontrar formas de se verificar se uma sentença, que depende de seus conectivos, que podem ser muitos, é verdadeira ou falsa. No estudo da Lógica Proposicional, foram analisadas as sentenças atômicas e as sentenças moleculares, construídas a partir dos conectivos **¬,** V**,** **^,** → e ↔. Nosso foco agora se volta para o estudo da Lógica de Predicados como uma extensão da Lógica Proposicional com maior poder de representação.

A necessidade deste estudo surge a partir de sentenças que não podem ser representadas de forma adequada na Lógica Proposicional. Vejamos, por exemplo, as seguintes sentenças:

- *p: Emiliano é pai de Chagas* e
- *q: Chagas é pai de Bruno*

pode ser verificado que foram usadas duas letras sentenciais diferentes, *p* e *q*, para expressar idéias semelhantes e, mesmo assim, com esta representação não foi captado o fato de que as duas sentenças se referem à mesma relação de parentesco entre Emiliano e Chagas e entre Chagas e Bruno.

Outro exemplo de limitação do poder de expressividade da linguagem proposicional diz respeito a sua incapacidade de representar instâncias de uma propriedade geral. Por exemplo, se quisermos representar as sentenças

- *r: todo objeto é igual a si mesmo* e
- *s: 5 é igual a 5,*

também tivemos que usar letras sentenciais distintas para representar cada uma das sentenças, sem captar que a segunda sentença é uma instância particular da primeira.

Estas observações permitem concluir que se, por algum processo de dedução, chegarmos à conclusão de que um indivíduo arbitrário de um universo tem uma certa propriedade, é razoável imaginar que esta propriedade também seja válida para qualquer indivíduo do universo em questão.

Usando uma linguagem proposicional para expressar

- *m: um indivíduo arbitrário de um universo tem uma certa propriedade* e

- *n: esta propriedade vale para qualquer indivíduo do universo*

também teríamos de usar dois símbolos proposicionais distintos e não teríamos como concluir a segunda sentença a partir da primeira.

A linguagem de primeira ordem pode captar relações entre indivíduos de um mesmo universo de discurso e a lógica de primeira ordem permite concluir particularizações de uma propriedade geral dos indivíduos de um mesmo universo, bem como derivar generalizações a partir de fatos que valem para um indivíduo arbitrário do universo em questão. Para ter este poder de expressividade, a linguagem de primeira ordem usa um conjunto de símbolos mais sofisticado do que o da linguagem proposicional.

Consideremos novamente a sentença *r: todo objeto é igual a si mesmo*. Esta sentença fala de uma propriedade (a de ser igual a si mesmo) que vale para todos os elementos de um universo, sem identificar os objetos deste universo.

Considere agora a sentença *u: existem números naturais que são pares.* Esta sentença descreve uma propriedade (a de ser par) que é válida para alguns (pelo menos para um) dos indivíduos do universo dos números naturais, sem, no entanto, referenciar o número "0" ou o número "2" ou o número "4", etc em particular.

Para expressar propriedades gerais (que valem para todos os indivíduos) ou existenciais (que valem para alguns indivíduos) de um universo são utilizados os *quantificadores* $\forall$ (universal) e $\exists$ (existencial), respectivamente. Estes quantificadores se apresentam sempre seguidos de um símbolo de *variável*, captando, desta forma, a idéia de estarem simbolizando as palavras "*para todos*" e "*para algum*".

Considere as sentenças:

- *p: Sócrates é homem* e
- *q: todo aluno do Departamento de Informática e Estatística estuda Lógica.*

A primeira sentença se refere a uma propriedade (ser homem) de um indivíduo em particular (*Sócrates*) em um domínio de discurso. Já a segunda sentença faz referência a elementos distingüidos (*Departamento de Informática e Estatística* e *Lógica*). Tais objetos podem ser representados usando os símbolos soc para *Sócrates*, inf para *Departamento de Informática e Estatística* e lg para *Lógica*. Tais símbolos são chamados de *constantes*.

As propriedades "*ser aluno de*" e "*estuda*" relacionam objetos do universo de discurso considerado, isto é, *ser aluno de* relaciona os indivíduos de uma Universidade com os seus Departamentos e *estuda* relaciona os indivíduos de uma Universidade com as matérias. Para representar tais relações serão usados símbolos de *predicados* (ou *relações*). Nos exemplos mostrados, podemos usar *Estuda* e *Aluno* como símbolos de relação binária. As relações unárias expressam propriedades dos indivíduos do universo (por exemplo *ser par* e *ser homem*). A relação *ser igual a* é tratada de forma especial e é representada pelo símbolo de igualdade $\approx$.

Desta forma, podemos simbolizar as sentenças conside-radas da seguinte forma:

- *Todo mundo é igual a si mesmo* por **$(\forall x)\ (x=x)$**;
- *Existem números naturais que são pares* por

  $(\exists x)\ (Par(x))$;
- *Sócrates é homem* por **Homem(soc)**;
- *Todo aluno do Departamento de Informática e Estatística estuda Lógica* por **$(\forall x)\ (Aluno(x,inf) \rightarrow Estuda\ (x,lg))$.**

Já vimos como representar objetos do domínio através de constantes. Uma outra maneira de representá-los é através do uso de símbolos de função. Por exemplo podemos representar os números naturais 1, 2, 3, etc, através do uso de símbolo de função, digamos, suc, que vai gerar nomes para os números naturais 1, 2, 3, etc. a partir da constante 0. Por exemplo, o número 1 vai ser denotado por suc(0) e o número 3 vai ser denotado por suc(suc(suc(0))). Seqüências de símbolos tais como suc(0) e suc(suc(suc(0))) são chamadas *termos*.

Assim, a sentença *todo número natural diferente de zero é sucessor de algum número natural* pode ser simbolizada por $(\forall x)\ \neg(x = 0) \rightarrow (\exists y)\ (suc(y) = x))$.

A ordem em que os quantificadores aparecem em uma expressão é muito importante. Por exemplo, a expressão $(\forall x)$ $(\exists y)$ Q(x,y) deve ser lida como “para todo x, existe um y tal que Q(x,y. Em uma interpretação onde o conjunto universo é o conjunto dos números inteiros e Q(x,y) é a propriedade $x < y$, a expressão anterior informa que para qualquer número inteiro x, existe um outro número inteiro y maior que x. Esta expressão tem um valor lógico verdadeiro. Agora vamos trocar a ordem em que os quantificadores aparecem na expressão, ou seja, $(\exists y)$ $(\forall x)$ Q(x,y). Neste caso, a expressão afirma que existe um número inteiro y que é maior do que qualquer outro número inteiro x. Neste caso, o valor lógico da expressão é falso.

1.19 O cálculo de predicados de 1ª ordem

O ***Cálculo de Predicados***, dotado de uma linguagem mais rica que o Cálculo Proposicional, tem várias aplicações importantes, não só para matemáticos e filósofos, mas também para estudantes de Ciência da Computação.

Nas linguagens de programação, conhecidas como *procedurais* (C e outras), os programas explicam de forma tácita como o computador deve proceder para realizar determinada tarefa. No entanto, existem outras linguagens de programação, conhecidas como declarativas (Prolog e outras), nas quais os programas são compostos por uma série de dados e um conjunto de regras que são usadas para gerar conclusões. Estes programas são conhecidos como Sistemas Especialistas ou Sistemas Baseados no Conhecimento, uma vez que eles simulam, em muitos casos, a ação de um ser humano. As linguagens declarativas incluem predicados, quantificadores, conectivos lógicos e regras de inferência, que constituem o Cálculo de Predicados.

1.20 Símbolos da linguagem

Para que possamos tornar a estrutura de sentenças complexas mais entendível é necessária a introdução de novos símbolos na linguagem do Cálculo Proposicional, obtendo-se a linguagem do Cálculo de Predicados de 1ª Ordem.

Para esta nova linguagem será considerado um novo alfabeto, como foi considerado para a Lógica Proposicional.

Definição1 (Alfabeto). O alfabeto da linguagem da Lógica de Predicados é definido pelo conjunto dos símbolos descritos a seguir:

- Símbolos de pontuação: (e).
- Símbolos de verdade: *false*.
- Um conjunto enumerável de símbolos para variáveis: x, y, z, w, x_1, y_1, z_1, w_1, x_2, ...

- Um conjunto enumerável de símbolos para funções: f, g, h, f_1, g_1, h_1, f_2, g_2, ...
- Um conjunto enumerável de símbolos para predicados: P, Q, R, P_1, Q_1, R_1. P_2, Q_2, ...
- Conectivos: ¬, V, ∀**,** ∃**.**

Associado a cada símbolo para função ou predicado, tem-se um número inteiro não negativo k. Este número indica a aridade ou número de argumentos da função ou predicado.

Como já foi dito, o alfabeto da linguagem da Lógica de Predicados é um extensão do alfabeto da Lógica Proposicional. Além dos infinitos símbolos contidos no alfabeto da linguagem da Lógica Proposicional, ele ainda contém infinitos símbolos para funções, predicados, variáveis, etc.

As variáveis. Os símbolos para variáveis formam um novo conjunto que não ocorre na Lógica Proposicional. Como será visto mais adiante, as variáveis têm um papel importante na Lógica de Predicados e na Ciência da Computação. Em programação Lógica, por exemplo, as variáveis são utilizadas na determinação das respostas dos programas.

As funções e os predicados. Os símbolos para funções e para predicados não ocorrem na Lógica Proposicional. A presença de tais símbolos na Lógica de Predicados permite um maior poder de representação. Na Lógica, na Matemática e na Ciência da Computação os conceitos de função e predicado são fundamentais. Na Ciência da Computação existem as linguagens funcionais que se baseiam no conceito de função, por exemplo, Haskell, SML, Miranda, KRC, Erlang e outras.

As constantes e os símbolos proposicionais. Cada símbolo para função ou predicado possui um número k, não negativo, a ele associado Quando k = 0, tem-se uma função ou predicado com zero argumentos. As funções com zero argumentos, ou aridade nula, representam constantes. De forma similar, os predicados com aridade zero representam símbolos proposicionais, que ocorrem no alfabeto da Linguagem Proposicional.

Notação. Os símbolos para funções zero-árias são denominados constantes. Elas são representadas por letras minúsculas a, b, c, a_1, b_1, c_1, a_2, b_2, etc.

Os símbolos para predicados zero-ários são denominados símbolos proposicionais. Eles são representados por P, Q, R, S, P_1, Q_1, R_1, S_1, P_2, Q_2 etc.

Como existem infinitos símbolos para funções com aridade zero, existem também infinitas constantes. De forma similar, existem infinitos símbolos proposicionais.

Os conectivos. O conjunto dos conectivos contém ¬ e V, que correspondem à versão simplificada do alfabeto da Lógica Proposicional. Além destes conectivos, existem também ∀ e ∃ que representam os quantificadores universal (para todo) e existencial (existe), respectivamente. Tais quantificadores ampliam o poder de representação da Lógica de Predicados, mas também aumentam a complexidade das demonstrações. Na linguagem da Lógica de Predicados, os outros conectivos Λ, → e ↔ são definidos a partir de ¬ e V, conforme é indicado na Lógica Proposicional. Além disso, o símbolo de verdade true também é definido a partir de *false*, uma vez que *true* = ¬ *false.*

Exemplos:

1. *Marta é inteligente*: **I(m)**; onde **m** está identificando *Marta* e **I** a propriedade de *ser inteligente*.
2. Alguém gosta de Marta: **G(x,m)**; onde **G** representa a relação *gostar de*, **x** representa *alguém* e **m** representa Marta.

De forma reduzida, pode-se afirmar que:

- **P(x)** significa que **x** tem a propriedade P;
- (∀**x)P(x)** significa que a propriedade **P** vale para todo **x,** ou ainda, que *todos os objetos* do conjunto Universo considerado tem a propriedade **P**.
- **(∃x)P(x)** significa que algum **x** tem a propriedade **P**, ou ainda, que *existe pelo menos um objeto* do conjunto Universo considerado que tem a propriedade **P.**

Os símbolos de predicados podem ser unários, binários ou n-ários, conforme a propriedade que representam, ou seja da quantidade de objetos que ela envolve, podendo ser um, dois ou mais. Neste caso, diz-se que o símbolo de predicado tem peso ou aridade 1, 2 ... ou n.

Observações:

- Um símbolo de predicado 0-ário (peso 0) identifica-se com um dos símbolos de predicado; por exemplo*: chove* podemos simbolizar *C.*
- As fórmulas mais simples do Cálculo de Predicados de 1ª Ordem são chamadas de ***fórmulas atômicas*** e podem ser definidas da seguinte forma:
    - *Se P for um símbolo de predicado de peso n e se t_1 , t_2 , ...,t_n forem termos então P(t_1 , t_2 , ...,t_n) é uma fórmula atômica.*

Fórmulas bem formadas na Lógica de Predicados

As fórmulas bem formadas na Lógica de Predicados serão chamadas de ***fbfs predicadas*** para diferenciá-las das **fbfs proposicionais**. Elas são definidas da seguinte maneira:

1. toda fórmula atômica é uma **fbf** predicada;
2. se α e β forem **fbfs** predicadas, então $(\neg\alpha)$, $(\alpha\wedge\beta)$, $(\alpha \vee \beta)$, $(\alpha\rightarrow \beta)$ e $(\alpha\leftrightarrow\beta)$ são **fbfs** predicadas;
3. se α for uma **fbf** predicada e **x** uma variável então $(\forall x)\alpha$ e $(\exists x)\alpha$ são **fbfs** predicadas;
4. as únicas **fbfs** predicadas são dadas por 1. 2. e 3..

Exemplos

1. A expressão **P(x) (∀ V) ∃y** não é uma fbf predicada**.**
2. As expressões, a seguir, são fbfs predicadas:

   a) $P(x,a)$;

   b) $(\forall z)(P(x,a) \rightarrow R(y,b,z))$;

   c) $\neg (\exists x)(\neg P(x,a) \wedge R(y,b,t))$;

   d) $(\exists y)(\forall x)R(y,b,t)$.

A seguir serão mostrados alguns exemplos da representação simbólica de algumas sentenças.

Todo amigo de Paulo é amigo de Francisco.
Gaspar não é amigo de Paulo
Logo, Gaspar não é amigo de Francisco.

Pode ser representado por

$$(\forall x) (A(x,p) \rightarrow A(x,f))$$

$$\neg A(g,p)$$

$$\neg A(g,f)$$

onde **A(x,y)** significa que **x** é amigo de **y**. As letras **p, f** e **g** são constantes que representam Paulo, Francisco e Gaspar, respectivamente.

Todos os humanos são racionais.
Alguns animais são humanos.
Portanto, alguns animais são racionais.

Pode ser representado por

$$(\forall x)\ (H(x) \rightarrow R(x))$$

$$(\exists x)\ (A(x) \wedge H(x))$$

$$(\exists x)\ (A(x) \wedge R(x))$$

onde **H, R** e **A** simbolizam as propriedades de: ser humano, ser racional e ser animal, respectivamente.

Escopo de um quantificador

Se α for uma fórmula e **x** for uma variável, então em $(\forall x)\alpha$ ou em $(\exists x)\alpha$ dizemos que α é o escopo do quantificador $(\forall x)$ ou $(\exists x)$.

Por exemplo, na fórmula $(\exists y)(\forall x)(R(y,b,t) \rightarrow (\forall z)P(z,a))$ temos os seguintes quantificadores e seus respectivos escopos:

$(\exists y)$: $(\forall x)(R(y,b,t) \rightarrow (\forall z)\ P(z,a))$

$(\forall x)$: $(R(y,b,t) \rightarrow (\forall z)\ P(z,a))$

$(\forall z)$: $P(z,a)$

Ocorrências livres e ligadas de uma variável

Uma ocorrência de uma variável **x** numa fórmula é *ligada* se **x** estiver no escopo de um quantificador $(\forall x)$ ou $(\exists x)$ na fórmula. Caso contrário a ocorrência de **x** é livre.

Se uma ocorrência de uma variável **x** for ligada em uma fórmula, dizemos que **x** é variável ligada nesta mesma fórmula, o mesmo valendo para as ocorrências livres. Isto significa que uma mesma variável pode ocorrer ligada em um ponto de uma fbf e livre em outro ponto da mesma fbf, ou seja, uma mesma variável pode ser livre e ligada ao mesmo tempo, em uma mesma fórmula. Este estudo tem importância fundamental na transformação de fbfs predicadas em fbfs equivalentes, um processo conhecido como forma *prenex* e *skolemização*.

Exemplo 3.1. Na fórmula $(\exists y)((\forall x)R(y,x,c) \rightarrow (\forall z)\ P(x,z))$ temos quatro ocorrências das variáveis que são **y, x, x** e **z.** O escopo de y é toda a fbf, logo a ocorrência de y em R(y,b,c) é ligada a ele. O escopo de x é R(y,x,c), logo a ocorrência de x é ligada a ele. Já o x de P(x,z) ocorre livre porque ele não está no escopo de x. A ocorrência de z é ligada.

Definição 3.2. Uma fórmula em que não há ocorrências livres de variáveis é chamada de *sentença*. Em um outro contexto, conhecido como *λ-cálculo* ou *cálculo lambda* (uma teoria matemática desenvolvida por Alonzo Church), ela é conhecida como *combinador*.

Termo livre para uma variável

Um termo **t** é livre para a variável **y** na fórmula α se, quando se substituem as ocorrências livres de **y** por **t**, as ocorrências de **t** em α assim obtidas ocorrem livres.

Exemplos:

1. **x** é livre para **y** em **P(y).**
2. **x** não é livre para **y** em $(\forall x)P(y)$**.**
3. **x** é livre para **x** em qualquer fórmula.
4. qualquer termo é livre para **x** numa fórmula α se em α não houver ocorrência livre de **x**.

Negação de fórmulas quantificadas

A partir da definição de fórmula dada anteriormente, verifica-se que os quantificadores universal e existencial podem ser precedidos de uma negação. Vejamos agora como podemos proceder, se for necessária, a eliminação dessa negação.

Consideremos, por exemplo, a fórmula$(\forall x)P(x)$ e o conjunto universo **U={a,b,c}**. Neste esse caso, temos:

$(\forall x)P(x) \Leftrightarrow P(a) \wedge P(b) \wedge P(c)$. Desta forma, pode-se considerar que:

$\neg(\forall x)P(x) \Leftrightarrow \neg(P(a) \wedge P(b) \wedge P(c)) \Leftrightarrow \neg P(a) \vee \neg P(b) \vee \neg P(c)$ o que significa que existe no mínimo um objeto em **U** tal que **¬P(x)**, ou seja, $\neg(\forall x)P(x) \Leftrightarrow (\exists x) \neg P(x)$ ou ainda de modo geral para uma fórmula α qualquer temos

(1) $\neg(\forall x)\, \alpha \Leftrightarrow (\exists x) \neg\alpha$

Da equivalência acima segue imediatamente que :

(2). $\neg(\forall x)\neg P(x) \Leftrightarrow (\exists x)P(x)$

(3). $\neg(\exists x)P(x) \Leftrightarrow (\forall x)\neg P(x)$

(4). $\neg(\exists x)\neg P(x) \Leftrightarrow (\forall x)P(x)$

1.21 Proposições categóricos

O estudo clássico ou aristotélico da dedução fundamenta-se em argumentos que contém proposições de um tipo especial, chamadas de “proposições categóricas”. Por exemplo, seja o argumento:

Nenhum atleta é vegetariano.
Todos os jogadores de futebol são atletas.

Logo, nenhum jogador de futebol é vegetariano.

Tanto as premissas quanto a conclusão deste argumento são proposições conhecidas como “categóricas” por serem habitualmente feitas sobre classes, afirmando ou negando que uma classe esteja incluída em uma outra, seja no todo ou em parte. Uma proposição categórica contém, apenas, um termo sujeito, um termo predicado e um operador silogístico que os une. As premissas e a conclusão desse argumento referenciam a classe dos “*atletas*”, a classe dos “*vegetarianos*” e a classe dos “*jogadores de futebol*”. Uma classe é uma coleção de todos os objetos que têm alguma característica específica em comum. As classes podem estar relacionadas entre si de várias formas. Se todo membro de uma classe for também membro de outra classe, diz-se que a primeira classe está incluída ou contida na segunda. Se apenas alguns membros de uma classe forem também membros de outra classe, diz-se que a primeira classe está parcialmente contida na segunda. Existem ainda algumas classes que não contém qualquer membro em comum, por exemplo, a classe dos triângulos e dos círculos. Essas várias relações distintas entre as classes são afirmadas ou negadas pelas proposições categóricas.

Há quatro formas típicas de proposições categóricas, ilustradas pelas quatro seguintes proposições:

1. Todos os políticos são ladrões.
2. Nenhum político é ladrão.
3. Alguns políticos são ladrões.
4. Alguns políticos não são ladrões.

A proposição 1 é universal e afirmativa onde afirma-se que a classe dos políticos está contida ou incluída na classe dos ladrões.. Isto implica que todo membro da primeira classe é também membro da segunda. Neste exemplo, o termo o termo sujeito “políticos” designa a classe de todos os políticos e o termo predicado “ladrões” designa a classe de todos os ladrões. Esta sentença pode ser escrita como

Todo P é L.

A proposição 2 é também universal mas negativa. Nega universalmente que os políticos sejam ladrões. Neste caso, verifica-se que a primeira classe, a classe dos políticos, está totalmente excluída da segunda classe, a classe dos ladrões, ou seja, não há qualquer membro da primeira classe que também esteja na segunda classe. Qualquer proposição universal negativa pode ser esquematizada da seguinte forma:

Nenhum P é L.

Onde, mais uma vez, as letras P e L representam os termos sujeito e predicado, respectivamente.

A proposição 3 é uma proposição particular afirmativa. Aqui se estabelece que alguns membros da primeira classe, a dos políticos, são também elementos da segunda classe, a classe dos ladrões. No entanto, a assertiva informa que algum ou alguns políticos, mas não todos, são ladrões. Isto significa que a sentença informa que as duas classes têm alguns membros em comum, mas não todos. Esta proposição é esquematizada da seguinte forma:

Algum P é L.

Onde as letras P e L representam os termos sujeito e predicado, respectivamente.

A proposição 4 é uma proposição particular e negativa. Também como o exemplo anterior, é particular porque não se refere a todos os políticos, mas apenas a alguns. Mas, ao invés da proposição anterior, não afirma que os membros particulares da primeira classe estejam incluídos na segunda classe. Ao contrário, ela nega. Esta proposição é esquematizada da seguinte forma:

Algum P não é L.

Onde as letras P e L representam os termos sujeito e predicado, respectivamente.

As quatro proposições vistas anteriormente, representam enunciados que são representados genericamente pelas letras **A, E, I, O** e onde as letras S e P significam sujeito e predicado, respectivamente.

> **A** - da forma "*Todo S é P*" (universal afirmativa);
> **E** - da forma "*Nenhum S é P*" ou "*Todo S não é P*" (universal negativa);
> **I** - da forma "*Algum S é P*" (particular afirmativa);
> **O** - da forma "*Algum S não é P*" (particular negativa).

Estes enunciados categóricos podem ser simbolizados respectivamente por:

$$\mathbf{A} - (\forall x)(S(x) \rightarrow P(x))$$

$$\mathbf{E} - (\forall x)(S(x) \rightarrow \neg P(x))$$

I - $(\exists x)(S(x) \wedge P(x))$

O - $(\exists x)(S(x) \wedge \neg P(x))$

Desde que a interpretação booleana das proposições categóricas depende substancialmente da noção de classe nula, é conveniente ter um símbolo especial para representá-la.

O símbolo de zero, 0, é utilizado para este fim. Para afirmar que a classe designada pelo termo S não tem membros, escreve-se o sinal de igualdade entre S e 0, Assim, a equação S = 0 afirma que a coleção S não tem qualquer membro.

Por outro lado, afirmar que a classe S tem membros equivale a negar que ela seja vazia. A classe dos elementos que não pertencem à coleção S é simbolizada por S' (complemento de S).

A sentença "Todo S é P" informa que todo elemento de S é também de P. Isto significa afirmar que não existe qualquer elemento de S que não seja também de S, ou seja, "nenhum S é não P".que pode ser representado pela equação SP' = 0. Assim as proposições categóricas A, E, I e O analisadas anteriormente podem ser representadas da seguinte forma:

A – $SP' = 0$

E – $SP = 0$

I – $SP \neq 0$

O – $SP' \neq 0$

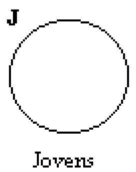
Pode-se observar que as proposições **A** e **O** são contraditórias, o mesmo acontecendo com as proposições **E** e **I.**

Diagramas de Euler-Venn para proposições categóricas

Se considerarmos S e P, representados anteriormente, como dois conjuntos quaisquer, as proposições referidas anteriormente podem ser interpretados a luz dos diagramas de Euler-Venn da Teoria dos Conjuntos. Isto pode ser útil na verificação da validade de argumentos onde as premissas e a conclusão sejam enunciados categóricos do tipo **A, E, I** ou **O.**

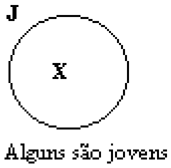
Apesar de representarem uma verdade, não devem ser considerados instrumentos rigorosos de prova. Deve ser lembrado que, no Cálculo Proposicional, os diagramas de Euler-Venn foram utilizados para estabelecer correlações entre as linhas da tabela verdade de uma fórmula e as regiões correspondentes do diagrama.

***Exemplo 3.2**:* Suponhamos que **J** represente o predicado "**ser jovem**". Desta forma, os predicados a seguir são representados da seguinte forma:



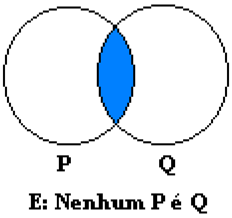
- cada círculo representa uma classe de objeto que quando em branco indica ausência de informação a respeito do conjunto, ou seja, não se sabe se tem, ou não, elementos neste conjunto.

- círculo hachurado ou **região de um círculo hachurada**, representa região **VAZIA** de elementos.

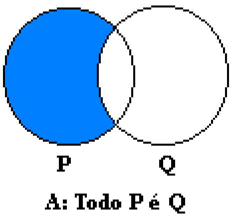


- círculo ou região de um círculo com **X** representa uma região não vazia de elementos.

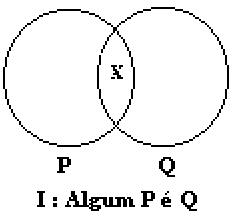
Os enunciados categóricos podem ser representados como pode ser verificado nas figuras ao lado, onde as letras **S** e **P** foram substituídas pelas letras **P** e **Q,** respectivamente.



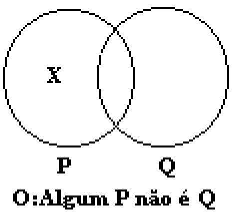
- **A**: **Todo P é Q** afirma que todos os elementos de P são também elementos de **Q,** ou seja, **P** é um subconjunto de **Q**, ou seja, os elementos de **P** que não fazem parte de **Q** formam um conjunto vazio, ou ainda, **PQ' = 0.**



- **E**: **Nenhum P é Q** afirma que os conjuntos **P** e **Q** não têm elementos em comum, isto é, que **P ∩ Q =∅** ou ainda **PQ = 0**.



- **I** : **Algum P é Q** afirma que os conjuntos **P** e **Q** têm pelo menos um elemento em comum, isto é, **P ∩ Q ≠∅,** ou ainda, **PQ ≠ 0.**



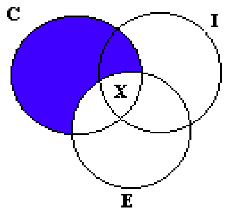
- **O**: **Algum P não é Q** afirma que **P** tem pelo menos um elemento que não está em **Q**, ou seja, que **P ∩ Q' ≠ ∅,** ou ainda, **PQ' ≠ 0.**

1.22 Validade de argumentos categóricos

Para verificar a validade de um argumento categórico deve-se proceder da seguinte forma:

- Transfere-se para o diagrama, formado por três círculos, as informações das premissas, iniciando pelos enunciados universais;
- Verifica-se se a informação dada na conclusão está representada ***sem nenhuma condição e de modo único***.
- se isto ocorrer, então o argumento é válido.

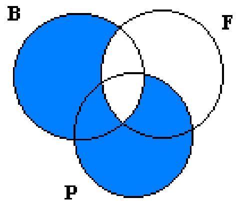
Vejamos os seguintes exemplos:



Exemplo 3.3.

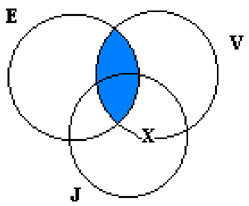
(1) Todos os cientistas são estudiosos.
(2) Alguns cientistas são inventores.
(3) Alguns estudiosos são inventores.

A parte hachurada corresponde ao enunciado (1), vazia de elementos; a parte assinalada com X corresponde ao enunciado (2). Dessa forma, as informações das premissas foram transferidas para o diagrama e a conclusão (3) está também representada. Portanto o argumento é válido.



Exemplo 3.4.

Todos os brasileiros são felizes.
Todos os paulistas são brasileiros.
Todos os paulistas são felizes.
O diagrama mostra que o argumento é válido



Exemplo 3.5.

(1) Nenhum estudante é velho

(2) Alguns jovens não são estudantes.

(3)Alguns velhos não são jovens.

A premissa (1) está representada na região hachurada e a premissa (2) está marcada com **X** sobre a linha pois a informação correspondente pode estar presente em duas regiões e não temos informação para saber especificamente em qual delas. Desse modo o argumento não é válido pois a conclusão não está representada com absoluta certeza.

A validade de um argumento não depende do conteúdo dos enunciados e sim da sua forma e da relação entre as premissas e a conclusão.

1.23 Árvores de refutação ou tableaux semânticos

SAIBA MAIS
Regras
http://www.pucsp.br/%7Elogica/Arvore.htm#Regras

No Cálculo Proposicional mostramos como as tabelas verdade, as demonstrações e as árvores de refutação, ou tableaux semânticos, podem ser usadas para a verificação da validade de argumentos e de tautologias. Será verificado agora como as árvores de refutação podem ser generalizadas para o Cálculo de Predicados de 1ª Ordem.

Já sabemos que as árvores de refutação permitem verificar a validade de argumentos em um número finito de passos. No entanto, esta técnica no Cálculo de Predicados pode não fornecer qualquer resposta em alguns casos como será verificado.

Para o Cálculo de Predicados de 1ª Ordem, é feita uma generalização das árvores de refutação mantendo todas as regras apresentadas para o Cálculo Proposicional. Isto é feito acrescentando-se novas regras para tratar com os quantificadores Universal ($\forall$) e Existencial ($\exists$). Assim, as seguintes novas regras são adicionadas:

Regra da Negação do Quantificador Universal **($\neg\forall$)**: Uma fórmula do tipo $\neg(\forall x)\beta$ gera uma linha na qual escrevemos a fórmula $(\exists x)\neg\beta$. Procedemos assim em todos os ramos abertos aos quais a fórmula $\neg(\forall x)\beta$ pertence.

Regra da Negação do Quantificador Existencial ($\neg\exists$) : Uma fórmula do tipo $\neg(\exists x)\beta$ gera uma linha na qual escrevemos a fórmula $(\forall x)\neg\beta$. Procedemos assim em todos os ramos abertos aos quais a fórmula $\neg(\exists x)\beta$ pertence.

Regra do Quantificador Existencial **($\exists$)**: Uma fórmula do tipo $(\exists x)\beta(x)$ gera uma linha na qual escrevemos a fórmula β**(c)** onde **c** é uma **nova constante** que não ocorre em qualquer ramo da árvore e substituirá as ocorrências da variável **x**, do quantificador, na fórmula β. Procedemos assim em todos os ramos abertos aos quais a fórmula $(\exists x)\beta(x)$ pertence. Esta regra também é conhecida como *particularização existencial*.

Justificativa*:* A fórmula $(\exists x)\beta(x)$ significa que existe pelo menos um objeto do Universo que tem a propriedade β e este será identificado, sempre, por uma "nova" constante ou seja, uma constante que não ocorre na árvore.

Regra do Quantificador Universal **($\forall$)**: Uma fórmula do tipo $(\forall x)\beta(x)$ gera uma linha na qual escrevemos a fórmula β**(c)** onde **c** é **qualquer constante que já ocorre** em qualquer ramo da árvore e substituirá as ocorrências da variável **x**, do quantificador, na fórmula **β**. Procedemos assim em todos os ramos abertos aos quais a fórmula **(∀x)β(x)** pertence. Esta regra é também conhecida como *generalização universal*.

Justificativa: A fórmula **(∀x)β(x)** significa que todos os objetos do universo tem a propriedade **β.** Sendo assim, a regra deve ser aplicada a todas as constantes presentes na árvore e eventualmente para aquelas que surgirem durante a "construção" da árvore como observamos abaixo.

OBSERVAÇÕES IMPORTANTES:

1. Como sabemos, as fórmulas para as quais são aplicadas as regras, sempre serão "marcadas" **(∠).** No entanto, para a regra **(∀)** do quantificador universal isto não será obedecido pois, se surgir uma nova constante na árvore por aplicação da regra **(∀)**, para esta constante deverá ser aplicada a regra **(∀)** em todas as fórmulas do tipo **(∀x)β(x)** da árvore.
2. Apenas no caso de nenhuma constante ocorrer em algum ramo é que podemos introduzir uma nova constante para ser usada em possíveis aplicações da regra **(∀)** ao longo do referido ramo.

Exemplo 3.6. Vamos verificar que a fórmula (∀**x)P(x)**→(∃**x)P(x)** é válida usando a árvore de refutação.

1. **¬((∀x)P(x) → (∃x)P(x))** ∠	Premissa
2. **(∀x)P(x)**	1. (¬→)
3. **¬ (∃x)P(x)**∠	1. (¬→)
4. **(∀x) ¬P(x)**	3. (¬∃)
5. **P(a)**	2. (∀) **(obs.2 acima)**

6. $\neg P(a)$	4. $(\forall)$
7. **X**	5. e 6.

Exemplo 3.7. Verifique a validade do argumento categórico:

Todos os cientistas são estudiosos. - $(\forall x)(C(x) \rightarrow E(x))$

<u>Alguns cientistas são inventores.</u> - $(\exists x)(C(x) \wedge I(x))$

Alguns estudiosos são inventores. - $(\exists x)(E(x) \wedge I(x))$

1.	$(\forall x)(C(x) \rightarrow E(x))$	Premissa
2.	$(\exists x)(C(x) \wedge I(x))$ ∠	Premissa
3.	$\neg(\exists x)(E(x) \wedge I(x))$ ∠	Premissa Adicional
4.	$(\forall x)\ \neg(E(x) \wedge I(x))$	3.$(\neg\exists)$
5.	$(C(a) \wedge I(a))$ ∠	2. $(\exists)$: a é nova constante
6.	$(C(a) \rightarrow E(a))$ ∠	1.$(\forall)$: a é constante que já ocorre
7.	$\neg(E(a) \wedge I(a))$ ∠	4. $(\forall)$: a é constante que já ocorre
8.	$C(a)$	5. $(\wedge)$
9.	$I(a)$	5. $(\wedge)$

```
                /          \
10.   ¬C(a)             E(a)              6.(→)
                          /      \
11. X (10,8)    ¬E(a)    ¬I(a)         7.( ¬∧)
12.                 X (10,11)   X(9,11)
```

O **argumento é válido,** pois todos os ramos foram fechados.

Exemplo 3.8. Verifique a validade do argumento categórico

Nenhum estudante é velho . $(\forall x)(E(x) \rightarrow \neg V(x))$

<u>Alguns jovens não são estudantes</u> $(\exists x)(J(x) \wedge \neg E(x))$

Alguns velhos não são jovens $(\exists x)(V(x) \wedge \neg J(x))$

1.	$(\forall x)(E(x) \rightarrow \neg V(x))$	Premissa
2.	$(\exists x)(J(x) \wedge \neg E(x))$∠	Premissa
3.	$\neg\ (\exists x)(V(x) \wedge \neg J(x))$∠	Premissa Adicional

4. $(\forall x)\neg(V(x)\wedge\neg J(x))$ 3. $(\neg\exists)$

5. $(J(a)\wedge\neg E(a))\angle$ 2. $(\exists)$: a é nova constante

6. $(E(a)\rightarrow\neg V(a))\angle$ 1. $(\forall)$: a é a constante que já existe.

7. $\neg(V(a)\wedge\neg J(a))\angle$ 4. $(\forall)$: a é constante que já existe

8. $J(a)$ 5. $(\wedge)$

9. $\neg E(a)$ 5.$(\wedge)$

/ \

10. $\neg E(a)$ $\neg V(a)$ 6.$(\rightarrow)$

/ \ / \

11. $\neg V(a)$ $\neg\neg J(a)$ $\neg V(a)$ $\neg\neg J(a)$ 7.$(\neg\wedge)$

12. / \ / \

O **argumento não é válido,** pois a árvore terminou e temos ramos abertos.

Exemplo 3.9: $(\forall x)(\exists y)P(x,y)$, $P(a,a)$

1. $(\forall x)(\exists y)P(x,y)$ $\angle$ Premissa
2. $\neg P(a,a)$ $\angle$ Premissa adicional.
3. $(\exists y)P(a,y)$ $\angle$ 1. $(\forall)$: a é constante que já existe.
4. $P(a,b)$ 3. $(\exists)$: b é nova constante.
5. $(\exists y)P(b,y)$ $\angle$ 1. $(\forall)$: b é constante que já existe.
6. $P(b,c)$ 5. $(\exists)$: c é nova constante.

Como se pode observar, a árvore nunca terminará; é infinita. Assim, pode-se assumir que o **argumento não é válido**.

Na verdade não existe um método efetivo que permita decidir sempre, e para qualquer argumento do Cálculo de Predicados, se um determinado argumento é válido ou não. Isto mostra que o Cálculo de Predicados é indecidível. A indecidibilidade do Cálculo de Predicados pode ser provada e é conhecida como ***Tese de Church***. Há muitos livros de Lógica e de Teoria da Computação que abordam este assunto com profundidade.

Quando verificamos a validade de um argumento estamos verificando se, no caso das premissas serem verdadeiras elas inferem uma determinada conclusão. Isto é possível ser feito por vários métodos no Cálculo Proposicional os quais nem todos se generalizam para o Cálculo de Predicados como verificamos acima.

1.24 Consequência lógica em Tableaux semânticos

Pode-se verificar que a aplicação dos tableaux semânticos na Lógica de Predicados é uma extensão da aplicação destes tableaux na Lógica Proposicional. Na Lógica de Predicados, eles definem uma estrutura para a representação e dedução de conhecimento. Esta estrutura é definida por conceitos análogos aos apresentados na Lógica Proposicional, onde, foi verificado que eles podem ser utilizados para provar teoremas e conseqüências lógicas. A seguir, serão mostrados exemplos mostrando como estes tableaux podem ser utilizados nestas demonstrações.

Exemplo.3.10. Seja a fbf h = $(\exists x)(\exists y)(p(x,y) \rightarrow p(a,a))$. Vamos provar a sua validade utilizando o método do tableau semântico. Para isso, vamos considerar ¬h e vamos verificar que vamos chegar a um tableau cujos ramos são todos fechados.

1. $\neg(\exists x)(\exists y)(p(x,y) \rightarrow p(a,a))$ $\angle$ --¬h
2. $(\exists x)(\exists y)(p(x,y) \wedge \neg p(a,a))$ $\angle$ --def de → em 1
3. $(\exists x)(\exists y)\ p(x,y)$ $\angle$ --R_8 em 2

4. $\neg p(a,a)$ --R_8 em 3
5. $(\exists y)\ p(t_1,y)$ $\angle$ --R_{12} em 3 e $t_1 \neq a$
6. $p(t_1,t_2)$ --R_{12} em 5, $t_2 \neq a$, $t_1 \neq t_2$

Verifica-se, neste ponto, que o desenvolvimento da negação de h atingiu a fbf $p(t_1,t_2)$, sendo $t_2 \neq a$, $t_1 \neq a$ e $t_1 \neq t_2$. Para que se chegasse a uma contradição seria necessário que se tivesse chegado à fbf $p(a,a)$ para se contradizer com $\neg p(a,a)$ da linha 4 da sequência de demonstração. Desta forma, o tableau não é fechado e isto significa que h não é válida.

É necessário, no entanto, saber analisar os resultados dos tableaux semânticos atingidos em uma sequência de demonstrações. Para nos guiar nesta direção, utilizaremos os teoremas a seguir, que não serão demonstrados, dado o escopo deste estudo. No entanto, o leitor mais exigente é convidado a consultar a bibliografia indicada.

Teorema da correção. Seja h uma fbf predicada. Se existir uma prova de h utilizando tableau semântico na Lógica de Predicados, então h é uma tautologia.

Teorema da completude. Seja h uma fbf predicada. Se h for uma tautologia, então existe uma prova de h utilizando tableaux semânticos.

Exemplo 3.11. Seja a fbf $h = (\forall x)(p(x) \land q(x)) \rightarrow (\forall x)p(x)$. Verifiquemos se pode-se provar sua validade, ou não, utilizando o método dos tableaux semânticos.

1. $\neg(\forall x)(p(x) \land q(x)) \rightarrow (\forall x)p(x)$ $\angle$ - $\neg h$
2. $(\forall x)(p(x) \land q(x)) \land \neg(\forall x)p(x)$ $\angle$ -def de $\rightarrow$ em 1

3. $(\forall x)(p(x) \land q(x))$ -R_8 em 2
4. $\neg(\forall x)p(x)$ $\angle$ -R_8 em 2
5. $(\exists \mathbf{x})\ \neg p(x)$ $\angle$ - R_{10} em 4
6. $\neg p(a)$ -R_{12} em 5
7. $p(a) \land q(a)$ R_{13} em 3, $x = a$
8. $p(a)$ - R_1 em 7

fechado (6,8)

Neste caso, o tableau é fechado e h é válida. Deve ser observado pelo leitor que na linha 6 a regra R12 foi utilizada antes da regra R_{13} na linha 7, onde faz-se x = a. Esta decisão tem uma conseqüência importante no tableau final e, como conseqüência, na prova, porque se for utilizada uma outra sequência de construção do tableau em que estes dois passos sejam invertidos (as linhas 6 e 7) poderemos chegar a um tableau diferente. Vamos considerar a mesma fbf h do exemplo anterior.

1. $\neg(\forall x)(p(x) \land q(x)) \rightarrow (\forall x)p(x)$ $\angle$ - $\neg h$
2. $(\forall x)(p(x) \land q(x)) \land \neg(\forall x)p(x)$ $\angle$ -def de $\rightarrow$ em 1
3. $(\forall x)(p(x) \land q(x))$ $\angle$ -R_8 em 2
4. $\neg(\forall x)p(x)$ $\angle$ -R_8 em 2
5. $(\exists x\mathbf{)}\ \neg p(x)$ $\angle$ - R_{10} em 4
6. $p(a) \land q(a)$ $\angle$ R_{13} em 3, a qualquer
7. $p(a)$ -R_1 em 6
8. $q(a)$ - R_1 em 6
9. $\neg p(t)$ -R_{12} em 5, $t \neq a$

Aberto

Neste caso, o termo t é qualquer na aplicação da regra R_{12} na linha 9, onde t deve ser diferente de a. Neste caso, o tableau obtido é aberto. Considerando estas duas sequências de demonstração por tableau, pode-se verificar que para uma mesma tautologia h é possível se determinar tableaux abertos ou fechados associados a uma fbf. Por outro lado, se h não for uma tautologia, então, pelo teorema da correção, não existe um tableau fechado associado a h.

Os teoremas da correção e da completude enunciados anteriormente, permitem afirmar que:

a) Se h for uma tautologia, então existe um tableau fechado associado a h.

b) Se h for uma tautologia, então pode existir um tableau aberto associado a h.

c) Se h não for uma tautologia, então não existe tableau fechado associado a h.

d) Se h não for uma tautologia, então todo tableau associado a h é aberto.

e) Se um tableau associado a h for fechado, então h é uma tautologia.

f) Se um tableau associado a h for aberto, então não se pode concluir se h é ou não uma tautologia.

g) Se todo tableau associado a h for aberto, então h não é uma tautologia.

Deve ser observado que, na Lógica Proposicional, se existir um tableau semântico fechado associado a uma fbf h, então todos os tableaux semânticos associados a h serão fechados.

Exemplo 3.12 (conseqüência lógica em tableaux semânticos).

a) Sejam $h_1 = (\exists x)(p(x) \rightarrow q(x))$ e $h_2 = (\forall x)p(x) \rightarrow (\exists x)q(x)$. Então h_1 é equivalente a h_2.

b) Sejam $h_1 = (\forall x)(p(x) \rightarrow q(x))$ e $h_2 = (\exists x)p(x) \rightarrow (\forall x)q(x)$. Então h_2 implica em h_1 mas h_1 não implica em h_2..

Solução:

a) Sendo $h_1 = (\exists x)(p(x) \rightarrow q(x))$ e $h_2 = (\forall x)p(x) \rightarrow (\exists x)q(x)$, então h_1 equivale a h_2 se, e somente se, $(h_1 \leftrightarrow h_2)$ for uma tautologia. $(h_1 \leftrightarrow h_2)$ será uma tautologia se, e somente se, $\models (h_1 \leftrightarrow h_2)$. Utilizando tableau semântico, temos:

1.	$\neg(\exists x)(p(x) \rightarrow q(x)) \leftrightarrow (\forall x)p(x) \rightarrow (\exists x)q(x)$. $\angle$				-- $\neg(h_1 \leftrightarrow h_2)$
2.	$(\exists x)(p(x) \rightarrow q(x)) \land \neg((\forall x)p(x) \rightarrow (\exists x)q(x))$		$\neg(\exists x)(p(x) \rightarrow q(x)) \land ((\forall x)p(x) \rightarrow (\exists x)q(x))$		--R_9 em 1
3.	$(\exists x)(p(x) \rightarrow q(x))$		$\neg(\exists x)(p(x) \rightarrow q(x))$		-R_1 em 2
4.	$\neg((\forall x)p(x) \rightarrow (\exists x)\, q(x))$		$(\forall x)p(x) \rightarrow (\exists x)\, q(x)$		-R_1 em 2
5.	$(\forall x)p(x)$	--R_8 em 4	$(\forall x)\neg(p(x) \rightarrow q(x))$		-R_{11} em 3
6.	$\neg(\exists x)\, q(x)$	--R_8 em 4			
7.	$p(a) \rightarrow q(a)$	--R_{12} em 3	$\neg(\forall x)p(x)$	$(\exists x)q(x)$	--R_3 em 4
8.	$(\forall x)\neg q(x)$	--R_{11} em 6	$(\exists x)\neg p(x)$		--R_3 em 4
9.	$\neg q(a)$	--R_{13} em 8	$\neg p(a)$	$q(a)$	--R_{12} em 7,8
10.	$p(a)$	--R_{13} em 5	$\neg p(a) \rightarrow q(a)$	$\neg(p(a) \rightarrow q(a))$	--R_{13} em 5
11.			$p(a)$	$p(a)$	--R_8 em 10
12.			$\neg q(a)$	$\neg q(a)$	--R_8 em 10
13.	$\neg p(a)$ $q(a)$	R_3 em 7	fechado	fechado	
	fechado fechado				

Como o tableau é fechado, então $(h_1 \leftrightarrow h_2)$ é uma tautologia e assim, h_1 equivale a h_2.

b) Neste caso, as fbfs são análogas às do item anterior, com a diferença de que os quantificadores universais estão trocados. Assim, tem-se: $h_1 = (\forall x)(p(x) \rightarrow q(x))$ e $h_2 = (\exists x)p(x) \rightarrow (\forall x)q(x)$, onde h_2 implica em h_1, mas h_1 não implica em h_2. Sabemos que h_2 implica em h_1 se, e somente se, $h_2 \rightarrow h_1$ for uma tautologia, ou seja, se e somente se $\models (h_2 \rightarrow h_1)$. A sequência de prova pode ser:

1 $\neg(((\exists x)p(x) \rightarrow (\forall x)(q(x)) \rightarrow (\forall x)(p(x) \rightarrow q(x)))$ -- $\neg(h_2 \leftrightarrow h_1)$

2 $(\exists x)p(x) \rightarrow (\forall x)(q(x)$ --R_8 em 1

3 $\neg(\forall x)(p(x) \rightarrow q(x))$ --R_8 em 1

4 $(\exists x)\neg(p(x) \rightarrow q(x))$ --R_{10} em 3

5 $\neg(p(a) \rightarrow q(a))$ --R_{12} em 4

6 $p(a)$ --R_8 em 5

7 $\neg q(a)$ --R_8 em 5

8 $\neg(\exists x)p(x)$ | $(\forall x)q(x)$ --R3 em 2

9 $(\forall x)\neg p(x)$ --R_{10} em 8 | $q(a)$ -- R_{12} em 8

10 $\neg p(a)$ --R_{12} em 9 | fechado(7,9)

11 fechado(6,10)

Como o tableau é fechado, então $h_2 \rightarrow h_1$ é uma tautologia. Logo, h_2 implica em h_1.

Para mostrar a segunda parte do item b), ou seja, que h_1 não implica em h_2, vamos considerar o tableau associado a $\neg(h_1 \rightarrow h_2)$ e vamos verificar que ele é aberto e que não é possível obter um tableau fechado a ele associado.

1 $\neg(((\forall x)(p(x) \rightarrow q(x)) \rightarrow ((\exists x)p(x) \rightarrow \neg(\forall x)\, q(x)))$ -- $\neg(h_1 \rightarrow h_2)$

2 $(\forall x)(p(x) \rightarrow q(x))$ -- R_8 em 1

3 $\neg((\exists x)p(x) \rightarrow (\forall x)\, q(x)$ -- R_8 em 1

4 $(\exists x)p(x)$ --R_8 em 3

5 $\neg(\forall x)\, q(x)$ --R_8 em 3

6 $(\exists x)\neg q(x)$ --R_{10} em 5

7 $p(a)$ –R_{12} em 4

8 $\neg q(b)$ –R_{12} em 3

9	$p(a) \rightarrow q(a)$		$p(a) \rightarrow q(a)$		$--R_{13}$ em 1
10	$\neg p(a)$	$q(a)$	$\neg p(a)$	$q(a)$	$--R_{\#}$ em 9
	fechado	aberto	fechado	aberto	

Como pode ser observado, este tableau é aberto e não é possível fechá-lo. Na linha 7, a regra R_{12} é aplicada substituindo-se a variável x pela constante “a”, obtendo-se p(a), porque “a” é a constante nova que não parece nas linhas anteriores do tableau (1 até 6). Na linha 8, a regra R_{12} é novamente aplicada, mas agora a variável x não pode mais ser substituída pela constante “a” porque ela já apareceu na linha 7, anterior. Por este motivo, foi escolhida a constante “b”, para se obter ¬q(b) na linha 8. Já na linha 9, a variável x poderia ser substituída tanto por “a” quanto por “b” que o resultado seria o mesmo. No caso, foi escolhida aleatoriamente a constante “a”. Isto significa que não é possível obter um tableau fechado neste caso e isto é verdade, mesmo que se inverta a ordem de aplicação das regras. Desta forma, não existe um tableau fechado associado a $\neg(h_1 \rightarrow h_2)$, ou seja, h_1 não implica em h_2.

Exemplo 3.13. Considere o argumento: “Todo aluno de Ciência da Computação é mais inteligente que algum aluno de Medicina. Logo, não existe aluno de Medicina que seja mais inteligente que todos os alunos de Ciência da Computação”. Este argumento é válido ou não?

Para responder a esta questão, vamos exibir uma prova utilizando a metodologia dos tableaux semânticos. Para isto, teremos:

p(x): x é aluno de Ciência da Computação.

q(x): x é aluno de Medicina.

r(x,y): x é mais inteligente que y.

Este argumento pode ser representado por

$$h = (\forall x)(p(x) \rightarrow (\exists y)(q(y) \land r(x,y))) \rightarrow \neg(\exists y)(q(y) \land (\forall x)r(y,x))$$

1	$\neg((\forall x)(p(x) \rightarrow (\exists y)(q(y) \wedge r(x,y))) \rightarrow \neg(\exists y)(q(y) \wedge (\forall x)r(y,x)))$	-- ¬h
2	$(\forall x)(p(x) \rightarrow (\exists y)(q(y) \wedge r(x,y)))$	--R_8 em 1
3	$\neg\neg(\exists y)(q(y) \wedge (\forall x)r(y,x)))$	-- R_8 em 1
4	$(\exists y)(q(y) \wedge (\forall x)r(y,x)))$	-- R_5 em 3
5	$(q(a) \wedge (\forall x)r(a,x))$	-- R_{12} em 4
6	$q(a)$	-- R_1 em 5
7	$(\forall x)(r(a,x))$	-- R_1 em 5
8	$p(a) \rightarrow (\exists y)\ (q(y) \wedge r(a,y))$	--R_{12} em 8
9	$\neg p(a)$ $(\exists y)\ (q(y) \wedge r(a,y))$	-- R_3 em 8
	aberto	

Este tableau contém um ramo aberto. Este fato não é suficiente para se concluir que h não seja uma tautologia. É necessário provar que todos os tableaux semânticos associados a ¬h são, obrigatoriamente, abertos para se concluir que h não seja uma tautologia e como conseqüência o argumento não é válido.

1.25 Forma prenex

Pelo que foi observado até este ponto deste estudo, as demonstrações da validade de argumentos não é uma tarefa fácil de ser realizada, necessitando de muita experiência no manuseio e construção dos mecanismos adotados para esta finalidade.

Neste particular, muita pesquisa tem despertado a atenção dos cientistas da Lógica e muitas técnicas tem se desenvolvido, notadamente na utilização e manuseio dos tableaux semânticos. Isto se deve ao fato desta técnica ser, até o momento, a que tem se mostrado mais adequada para ser implementada como programas de computadores.

Uma metodologia que tem sido estudada e utilizada com bastante sucesso se refere a transformação de fbfs predicadas em fbfs equivalentes na forma prenex que, de forma generalizada, é uma fbf onde os quantificadores se encontram todos em seu início.

Esta metodologia se justifica porque toda fbf predicada tem uma fbf equivalente na forma prenex e também pelo fato de que as formas prenexes são mais fáceis de serem provadas e implementadas mecanicamente.

Para dar início a este estudo, é necessário enunciar algumas definições.

Definição 3.3 (fórmula aberta). Uma fórmula da Lógica de Predicados é dita aberta se ela não possui qualquer quantificador.

Definição 3.4 (forma prenex). Uma fórmula h da Lógica de Predicados está na forma prenex se h for do tipo h = $(Qx_1)...(Qx_n)g$, onde g é uma fórmula aberta e os Qx_i são quantificadores universal ou existencial.

Exemplo 3.14. As fórmulas $(\exists x)(\forall x)(r(x,y)\rightarrow p(y))$ e $(\exists x)(\forall x)(p(x)\land q(x,y))$ estão na forma prenex porque todos os seus quantificadores estão no início da fórmula, seguidos por fórmulas abertas. Já a fórmula $(\exists y)((\forall x)p(x)\rightarrow q(x,y))$ não está na forma prenex porque o escopo do quantificador universal é apenas p(x) e não o restante da fórmula. Para estar na forma prenex, todos os quantificadores devem estar no início da fórmula e os seus escopos devem ser extendidos até o final da fórmula.

Como afirmado anteriormente, toda fbf predicada tem uma fbf predicada equivalente na forma prenex. O algoritmo prenex, que transforma uma fbf predicada h em uma fbf g na forma prenex considera a definição e as regras a seguir.

Definição 3.5 (regras prenex). Sejam h e g duas fbfs e Qx_1 e Qx_2 dois quantificadores que podem ser existencial ou universal. As regras prenexes são as seguintes:

$$R_1: \frac{(\forall x)p \wedge q}{(\forall x)(p \wedge q)} \qquad R_2: \frac{(\forall x)p \vee q}{(\forall x)(p \vee q)} \qquad R_3: \frac{(\exists x)p \wedge q}{(\exists x)(p \wedge q)}$$

$$R_4: \frac{(\exists x)p \vee q}{(\exists x)(p \vee q)} \qquad R_5: \frac{(\forall x)p \wedge (\forall x)q}{(\forall x)(p \wedge q)} \qquad R_6: \frac{(\exists x)p \vee (\exists x)q}{(\exists x)(p \vee q)}$$

$$R_7: \frac{(Q_1x)p \wedge (Q_2y)q}{(Q_1x)(Q_2y)(p \wedge q)} \qquad R_8: \frac{(Q_1x)p \vee (Q_2y)q}{(Q_1x)(Q_2y)(p \vee q)}$$

- Nas regras R_1, R_2, R_3 e R_4, a variável x não ocorre livre em q.
- Nas regras R_7 e R_8, a variável x não ocorre livre em q e y não ocorre livre em p.

Deve ser observado que as regras prenexes deduzem fórmulas equivalentes, por exemplo, $(\forall x)p$^q equivale a $(\forall x)(p \wedge q)$. Assim, não é possível deduzir $(\forall x)(p \vee q)$.a partir de $(\forall x)p \vee (\forall x)q$, nem deduzir .$(\exists x)(p \wedge q)$ a partir de $(\exists x)p \wedge (\exists x)q$ porque não são equivalentes. Além disso, todas as fórmulas deduzidas tem os seus quantificadores no início, mesmo que não estejam na forma prenex, porque não se pode aplicar as regras prenexes às fórmulas anteriores.

Para resolver este problema, é necessário que as variáveis sejam renomeadas e isto é feito baseando-se na seguinte definição:

Definição 3.6 (R_0 - renomeação de variáveis). Considere a fbf h = (Qx)g, sendo (Qx) um quantificador universal ou existencial. A renomeação da variável x por uma outra variável y se dá da seguinte forma:

f = (Qy)(g[x/y]) onde g[x/y] é uma substituição segura

Exemplo 3.15. Considere a fbf h = $(\forall x)(p(x)\rightarrow(\exists x)q(x,y))$ que contém dois quantificadores na mesma variável x. Neste caso, a renomeação de x em sua primeira ocorrência é feita deduzindo-se a $h_1 = (\forall z)(p(z)\rightarrow(\exists x)q(x,y))$, onde a segunda ocorrência de x em q(x,y) não pode ser renomeada porque esta ocorrência de x está no escopo do quantificador existencial $(\exists x)$ por ser mais interno que o quantificador universal. Se for necessária uma renomeação de x, em sua segunda ocorrência, ela poderá ser feita por outra variável, por exemplo, por w, onde h1 se transformará em h2 da seguinte forma: h_2 = . $(\forall z)(p(z)\rightarrow(\exists w)q(w,y))$,

Exemplo 3.16. Seja a fbf predicada $(\forall x)(\forall y)p(x,y,z)$. Neste caso, x não pode ser renomeada por y, porque se isso fosse feito a fórmula renomeada seria $(\forall y)p(y,y,z)$ e, neste caso, o contexto seria outro.

Este exemplo mostra que a renomeação deve ser feita por uma variável que não tenha ocorrido ainda no contexto. Se for feita por uma variável que já faz parte da fórmula, ocorre um fenômeno conhecido como “problema de captura”. No caso em voga, se a substituição de x por y fosse feita, o a variável x seria capturada por y. Este fato tem importância fundamental nas linguagens de programação onde as variáveis não locais a um determinado subprograma não podem ser renomeadas por uma variável local porque neste caso a variável não local se tornaria local por ter o mesmo nome desta e, neste caso, o ambiente de referência seria outro bem diferente do original.

Definição 3.7 (regra prenex de renomeação de variáveis). Seja h uma fbf predicada da seguinte forma: $(Q_1x_1)...(Q_nx_n)$ e as variáveis livres $z_{1,}...,z_k$. A regra prenex de renomeação de variáveis (R_0) corresponde à renomeação das variáveis x_1, ...,x_n pelas variáveis y_1,...,y_n, de forma que $y_i \neq y_j$ para $i \neq j$ e y_i não pertence ao conjunto $\{x_1, ...,x_n,z_1,...,z_k\}$.

As variáveis renomeadas y_1,...,y_n são deferentes entre si e diferentes das variáveis livres z_1,...,z_k e das variáveis x_1,...,x_n.

Exemplo 3.17. Seja a fbf predicada $h = (\forall x)(p(x)\rightarrow(\exists x)q(x,y))$.

Aplicando-se a regra R_o a h temos $h_1 = (\forall z)(p(z)\rightarrow(\exists w)q(w,y))$,

onde as variáveis z e w são diferentes da variável livre y.

Exemplo 3.18. Seja a fbf predicada $h = (\forall x)p(x) \vee (\forall x)q(x)$. Este

caso se inclui entre os que não é possível se aplicar qualquer regra prenex. No entanto, é possível se aplicar a regra R_0, ou

seja, $h_1 = (\forall y)p(y) \vee (\forall z)q(z)$. Nesta fbf podemos aplicar a regra

prenex R_8 transformando-a em $h_2 = (\forall y)(\forall z)(p(y) \vee q(z))$ e h_2 está

na forma prenex.

Neste ponto, estamos preparados para conhecer o algoritmo prenex, que é utilizado para transformar uma fbf h não prenex em uma fbf g na forma prenex.

Definição 3.8 (algoritmo prenex). Sejam h, g, p e q fbfs predicadas. O algoritmo a seguir transforma h em uma fbf equivalente g, na forma prenex.

1. Substitua as fbfs $(p\rightarrow q)$ por $(\neg p\vee q)$.
2. Substitua as fbfs $\neg(p\wedge q)$ por $(\neg p\vee\neg q)$.
3. Substitua as fbfs $\neg(p\vee q)$ por $(\neg p\wedge\neg q)$.
4. Substitua as fbfs $\neg\neg p$ por p.
5. Substitua as fbfs $\neg(\forall x)p$ por $(\exists x)\neg p$.
6. Substitua as fbfs $\neg(\exists x)p$ por $(\forall x)\neg p$.
7. Substitua as fbfs $(p\leftrightarrow q)$ por $(p\rightarrow q)\wedge(q\rightarrow p)$. Se for necessário, volte ao passo 1, até obter uma fbf com apenas os conectivos $\neg$, $\vee$ e $\wedge$.
8. Aplique R_0 para renomear variáveis.
9. Utiliza as regras R_1 a R_8 para substituir fbfs

A fbf g obtida pela aplicação dos passos 1 até 9 é equivalente a h e está na forma prenex.

Exemplo 3.19. Seja $h = (\forall x)p(x)\wedge((\forall x)q(x)\rightarrow(\exists y)r(x,y,z))$. Vamos utilizar o algoritmo prenex para transformar h em uma outra fbf equivalente na forma prenex. Isto será feito a seguir, onde cada passo corresponde ao passo de mesmo número no algoritmo, mesmo que o passo não seja aplicável.

1. $h_1 = (\forall x)p(x) \wedge (\neg(\forall x)q(x) \vee (\exists y)r(x,y,z))$
2. Não aplicável.
3. Não aplicável.
4. Não aplicável.

5. $h_5 = (\forall x)p(x) \wedge ((\exists x)\neg q(x) \vee (\exists y)r(x,y,z))$.

6. Não aplicável.

7. Não aplicável.

8. R_0. $h_8 = (\forall y_1)p(y_1) \wedge ((\exists y_2)\neg q(y_2) \vee (\exists y_3)r(x,y_3,z))$.

9. $h_9 = (\forall y_1)p(y_1) \wedge ((\exists y_2)(\exists y_3)(\neg q(y_2) \vee r(x,y_3,z))$

$g = (\forall y_1)(\exists y_2)(\exists y_3)\ (p(y_1) \wedge (\neg q(y_2) \vee r(x,y_3,z))$.

1.26 Skolemização

Deve ser observado que na aplicação do algoritmo prenex da seção anterior, em cada passo, a fórmula resultante é equivalente à fórmula do passo anterior. Logo, por transitividade, a saída do algoritmo é uma fórmula equivalente à fórmula de entrada. Agora será visto um algoritmo para obtenção de fórmula da forma $\underline{(Qx)}\beta$ onde só aparecem quantificadores universais e β é uma fórmula aberta na forma normal conjuntiva. O processo de obtenção de tais fórmulas é chamado de *Skolemização*.

Durante a Skolemização são introduzidos novos símbolos de função, isto é, que não ocorrem na assinatura da linguagem da fórmula de entrada. A saída do algoritmo é uma fórmula que não é logicamente equivalente à fórmula de entrada, mas que tem a propriedade de ser satisfatível se e só se a fórmula de entrada o for. Antes de darmos o algoritmo daremos alguns exemplos para ilustrar o papel dos símbolos novos introduzidos durante a Skolemização.

Exemplo 3.20. Considere a fórmula P(x), cujo significado pretendido é: o valor atribuído a x tenha a propriedade expressa por P. Para isto ser verdade, é necessário que exista um objeto no universo que tenha a propriedade expressa por P. Assim, a fórmula $(\exists x)P(x)$ é verdade neste contexto. Por outro lado, se $(\exists x)P(x)$ é verdade em um contexto, então existe um objeto no universo que tem a propriedade expressa por P. Logo, a fórmula P(x) é verdade neste contexto quando atribuímos este objeto a x. Neste exemplo mostrarmos que P(x) é satisfatível se e somente se $(\exists x)P(x)$ for satisfatível.

Exemplo 3.21. Considere a fórmula $(\exists x)P(x)$ cujo significado pretendido é que exista algum objeto no universo que tenha a propriedade expressa por P**.** Assim, se adicionarmos uma constante c à assinatura de P e interpretarmos c como este objeto que tem a propriedade P, a fórmula P(c) passa a ser verdade neste contexto. Por outro lado, se P(c) é verdade em um contexto, então a interpretação de c é um objeto que tem a propriedade que P expressa neste contexto. Logo, existe um objeto no contexto que tem a propriedade expressa por P, isto é, a fórmula $(\exists x)P(x)$ é verdade neste contexto. Neste exemplos mostramos que $(\exists x)P(x)$ é satisfatível se e somente se P(c) for satisfatível.

Exemplo 3.22. Agora considere a fórmula $(\forall y)(\exists x)P(x,y)$ cujo significado pretendido é que para qualquer elemento do universo de discurso exista um objeto que esteja relacionado por P com aquele elemento. É claro que para cada elemento *e* que estivermos considerando, o objeto que existe relacionado com *e* não precisa ser único, nem ser o mesmo relacionado com todos os elementos do universo. Isto significa que objetos diferentes podem estar relacionados com elementos diferentes ou alguns objetos diferentes podem estar relacionados com o mesmo objeto. Além disso, pode haver mais de um objeto relacionado com o mesmo elemento. De qualquer modo, podemos definir uma função tal que, para cada elemento *e* do universo, escolhe-se um dos objetos dentre os que estejam relacionados com *e*. Observe que esta escolha não precisa ser feita de forma efetiva, (por exemplo, um sorteio é uma escolha não efetiva) mas que pode sempre ser feita pelo fato de sempre haver pelo menos um objeto relacionado com cada elemento do universo. Isto é, a fórmula $(\forall y)P(f(y),y)$ é verdade neste contexto, quando f é um símbolo novo de função que é interpretado como a função acima descrita.

Por outro lado, se $(\forall y)P(f(y),y)$ for verdade em um contexto, então, para cada elemento *e* do contexto, o objeto nomeado por *f*(*e)* está relacionado com *e* (onde *f* é a interpretação de *f* no contexto), ou seja,. a fórmula $(\forall y)(\exists x)P(x,y)$ é verdade neste contexto. Este exemplo mostra que $(\forall y)(\exists x)P(x,y)$ é satisfatível se e somente se $(\forall y)P(f(y),y)$ for satisfatível, onde *f* é um símbolo novo de função.

Na discussão acima os símbolos novos, c de constante e f de função, são introduzidos com significados pretendidos específicos. Tais símbolos são chamados de funções de Skolem. Estas idéias serão agora formalizadas.

Proposição PS3. Para cada fórmula α existe um procedimento efetivo para se obter uma fórmula na forma $(\underline{Qx})\beta$ onde só aparecem quantificadores universais no prefixo Qx e β é uma fórmula aberta na forma normal conjuntiva tal β é satisfatível se e somente se $(\underline{Qx})\beta$ for satisfatível.

Prova: A fórmula de saída poderia ser obtida a partir da fórmula de saída do algoritmo de obtenção de forma norma conjuntiva, mas por questão de eficácia, daremos um algoritmo alternativo.

Dada uma fórmula α:

1. Tome o fecho existencial de α, ou seja, se α contiver uma variável livre x**,** substitua α por $(\exists x)\alpha$**.** Repita este processo até que a fórmula corrente não tenha mais variáveis livres.
2. Elimine quantificadores redundantes, ou seja, elimine todo quantificador $(\forall x)$ ou $(\exists x)$ que não contenha qualquer ocorrência livre de x no seu escopo.
3. Renomeie as variáveis ligadas de forma que as variáveis governadas por quantificadores sejam todas distintas.
4. Elimine as ocorrências dos conectivos $\rightarrow$ e $\leftrightarrow$.
5. Mova o conectivo $\neg$ para o interior da fórmula até que preceda imediatamente fórmulas atômicas.
6. Mova os quantificadores para o interior da fórmula
7. Elimine os quantificadores existenciais (Skolemização). Seja ϕ a fórmula corrente. Obtenha a nova fórmula corrente, ϕ', substituindo a subfórmula da forma $(\exists y)\beta$, que se situa mais à esquerda em ϕ por $\beta[y/f(x_1, ..., x_n]$**,** onde:$x_1, ..., x_n$ é uma lista de todas as variáveis livres de $(\exists y)\beta$ e *f* é um símbolo de função n-ário (função de Skolem) que não ocorre em ϕ. Caso não haja variáveis livres em $(\exists y)\beta$ substitua $(\exists y)\beta$ por $\beta[y/c]$**,** onde c é uma constante (de Skolem) que não ocorre em ϕ. Repita o processo (de Skolemização) até que todos os quantificadores existenciais tenham sido eliminados.
8. Obtenha a forma normal *prenex* da fórmula obtida no passo 6, ou seja, mova os quantificadores para a esquerda.

9. Obtenha a forma normal conjuntiva da matriz da fórmula obtida no passo 7, substituindo a matriz desta pela forma normal conjuntiva obtida.
10. Simplifique a fórmula do passo 9 eliminando repetições de literais no mesmo conjunto e disjunções que são tautologias.

Observe que:

- o fecho existencial da fórmula obtido no passo 1 é satisfatível se e somente se a fórmula original for satisfatível. O argumento é análogo ao usado no exemplo Sklm1.
- os passos 2 a 6 produzem fórmulas equivalentes à formula obtida no passo 1. Assim, a fórmula obtida no passo 6 (equivalente à do passo 1) é satisfatível se e somente se a fórmula de entrada, α, for satisfatível.
- a cada introdução de símbolo de função ou constante de Skolem, ocorrida no passo 7, a fórmula obtida é satisfatível se e só se a fórmula antes da substituição for satisfatível. O argumento é análogo ao usado nos exemplos Sklm2 e Sklm3.
- os passos 8 a 10 produzem fórmulas equivalentes à fórmula obtida no passo 7. Assim, a fórmula obtida pelo passo 10 (equivalente à do passo 7) é satisfatível se e somente se a fórmula de entrada, α, for satisfatível.
- os passos 6 e 10 são opcionais. O passo 6 se justifica por evitar que sejam introduzidas no passo 7 funções com aridade maior do que a necessária. A aridade da função de Skolem introduzida da esquerda para direita vai depender do número de quantificadores universais que estejam à esquerda do quantificador existencial que está sendo eliminado.

Exemplo 3.23. Seja agora $h = (\exists y)(\forall x)p(x) \leftrightarrow q(x,y,z)) \wedge (\forall x)(\neg p(x) \rightarrow \neg (\forall y)r(x,y))$**.** Aplicando o passo 1 obtemos o fecho existencial de α: $\alpha_1 = (\exists z)(\exists x)(\exists y)((\forall x)p(x) \leftrightarrow q(x,y,z)) \wedge (\forall x)(\neg r(x) \rightarrow \neg(\forall y)p(x,y))$. O passo 2 não se aplica. Renomeando-se as variáveis quantificadas, temos:

$\alpha_2 = (\exists s)(\exists u)(\exists y)((\forall x)p(x) \leftrightarrow q(u,y,s) \wedge (\forall z)(\neg p(z) \rightarrow \neg(\forall v)r(z,v))$. Eliminando os conectivos $\leftrightarrow$ e $\rightarrow$ em α_2 temos:

$\alpha_3 = (\exists s)(\exists u)(\exists y)((\neg(\forall x)p(x) \vee q(u,y,s) \wedge (\neg q(u,y,s) \vee (\forall x)p(x) \wedge (\forall z)(p(z) \vee \neg(\forall v)r(z,v))$. Movendo $\neg$ para o interior da fórmula temos:

$\alpha_4 = (\exists s)(\exists u)(\exists y)(((\exists x)\neg p(x) \vee q(u,y,s) \wedge (\neg q(u,y,s) \vee (\forall x)p(x) \wedge (\forall z)(p(z) \vee (\exists v)\neg r(z,v)))$

O passo 6 não se aplica. Eliminando os quantificadores existenciais temos:

$\alpha_5 = ((\neg p(d) \vee q(b,c,a) \wedge (\neg q(b,c,a) \vee (\forall x)p(x) \wedge (\forall z)(p(z) \vee \neg r(z,f(z)))$

Obtendo a forma normal prenex temos:

$\alpha_6 = (\forall z)(\forall x)((\neg p(b) \vee q(c,a,d) \wedge (\neg q(c,a,d) \vee p(x) \wedge (p(z) \vee \neg r(z,f(z)))$

A matriz da forma já está na forma conjuntiva e o passo 10 não se aplica.

1.27 RESUMO

Esta unidade compreendeu o estudo da Lógica de Predicados, uma teoria complementar à Lógoca Proposicional vista nas unidades 1 e 2. A idéia de ser complementar, não significa que é menos importante. Na realidade, a Lógica de Predicados é mais completa que a Lógica Proposicional por ser capaz de resolver uma gama bem maior de tipos de problemas. Estes problemas envolvem situações não possíveis de serem resolvidas apenas com a teoria da Lógica Proposicional.

A Lógica de Predicados utiliza os quantificadores universal e existencial e funções para simbolizar e analisar sentenças que não eram possíveis de serem construídas apenas com as estruturas da Lógica Proposicional. Este foi o principal objetivo desta unidade.

A prova de fórmulas bem formadas na Lógica de Predicados é mais complexa que na Lógica Proposicional, sendo necessário o conhecimento de técnicas mais elaboradas para que as demonstrações possam ser levadas a efeito. Entre estas técnicas, está a substituição de fórmulas bem formadas predicadas por outras fórmulas equivalentes, mas com um formato distinto, porém mais fácil de ser construída uma demonstração para ela.

Apesar desta ser uma Unidade final, é importante mencionar que a Lógica compreende muitos outros temas e existem várias áreas de estudo e pesquisa da Lógica. Estas áreas têm representado um campo fértil de pesquisa há muito tempo, mas estão fora do escopo deste estudo.

Exercícios

1. Determine o valor lógico de cada uma das fbfs a seguir com a interpretação de que o conjunto universo é o conjunto dos ineiros, p(x) significa que “x é ímpar”, q(x) que “x < 0” e g(x) que “x > 9”.
   a) $(\exists x)p(x)$
   b) $(\forall x)[q(x) \rightarrow p(x)]$

c) $(\exists x)[q(x) \wedge g(x)]$

d) $(\forall x)[q(x) \vee g(x)]$

2. Qual o valor lógico de cada uma das fbfs a seguir com a interpretação em que o conjunto universo seja o conjunto dos inteiros?

a) $(\forall x)(\exists y)(x + y = x)$

b) $(\exists y)(\forall x)(x + y = x)$

c) $(\forall x)(\exists y)(x + y = 0)$

d) $(\exists y)\ (\forall x)\ (x + y = 0)$

e) $(\forall x)(\forall y)(x < y \vee y < x)$

f) $(\forall x)[x < 0 \rightarrow (\exists y)(y > 0 \wedge x + y = 0)]$

g) $(\exists x)(\exists y)(x^2 = y)$

h) $(\forall x)(x^2 > 0)$

3. Decida se é possível chegar a alguma conclusão a partir das hipóteses dadas a seguir e, caso positivo, qual é esta conclusão.,

"*Todas as flores são vermelhas ou roxas. Amores-perfeitos são flores. Amores-perfeitos não são roxos*".

4. Justifique cada passo na sequência de demonstração a seguir, para a fbf $(\exists x)[p(x) \rightarrow q(x)] \rightarrow [(\forall x)p(x) \rightarrow (\exists x)q(x)]$.

1 $(\exists x)[p(x) \rightarrow q(x)]$

2 $p(a) \rightarrow q(a)$

3 $(\forall x)p(x)$

4 $p(a)$

5 $q(a)$

6 $(\exists x)q(x)$

5. Justifique cada passo na sequência de demonstração a seguir, para a fbf $(\exists x)p(x)\wedge(\forall x)(p(x)\rightarrow q(x))\rightarrow(\exists x)q(x)$

1 $(\exists x)p(x)$

2 $(\forall x)(p(x) \rightarrow q(x))$

3 $p(a)$
4 $p(a) \to q(a)$
5 $q(a)$
6 $(\exists x)q(x)$

6. Considere a seguinte fbf $(\forall x)[(\exists y)p(x,y) \wedge (\exists y)q(x,y)] \to (\forall x)(\exists y)[p(x,y) \wedge q(x,y)$

   a. Encontre uma interpretação que mostre que essa fbf não é válida.

   b. Encontre o erro na seguinte sequência de demonstração para essa fbf :

1	$(\forall x)[(\exists y)p(x,y) \wedge (\exists y)q(x,y)]$	- hipótese
2	$(\forall x)[p(x,a) \wedge q(x,a)]$	-1, pe
3	$(\forall x)(\exists y)[p(x,y) \wedge q(x,y)]$	-2, ge

7. Prove que as fbfs a seguir são argumentos válidos :
   a) $(\forall x)p(x) \to (\forall x)[p(x) \vee q(x)]$
   b) $(\forall x)p(x) \wedge (\exists x)q(x) \to (\exists x)[p(x) \wedge q(x)]$
   c) $(\exists x)(\exists y)p(x,y) \to (\exists y)(\exists x)p(x,y)$
   d) $(\forall x)(\forall y)q(x,y) \to (\forall y)(\forall x)q(x,y)$
   e) $(\forall x)p(x) \wedge (\exists x)\neg[p(x) \to (\exists x)q(x)$

SAIBA MAIS

Existem muitos bons textos sobre este tema. Alguns deles estão listados na Bibliografia colocada ao final desta Unidade. Outros estão na Internet à disposição. Estes estão listados a seguir.

WEB-BIBLIOGRAFIA

www.ufpi.br/uapi (A Página da Universidade Aberta do Piauí - UAPI)

www.uab.gov.br (O Site da Universidade Aberta do Brasil- UAB)

www.seed.mec.gov.br (A Homepage da Secretaria de Educação a Distância do MEC - SEED)

www.abed.org.br (O site da Associação Brasileira de Educação a Distância - ABED)

http://pt.wikipedia.org/ O site da Wikipedia.

www.inf.ufsc.br/ine5365/introlog.html

www.gregosetroianos.mat.br/logica.asp

REFERÊNCIAS BIBLIOGRÁFICAS

ABRAMSKY, S. et all. *Admissibility of Logical Inference Rules*. North Holland, 1997.

ABRAMSKY, S, GABBAY, Dov and MALBAUM, T.S.E. editors. *Handbook of Logic in Computer Science, vol I*. Oxford University Press, 1992.

ALENCAR FILHO, E. *Iniciação à Lógica Matemática*. Editora Nobel, 1986.

BEN-ARI, Mordechai. *Mathematical Logic for Computer Science*. Springer, second edition, 2001.

BRODA, Krysia Broda et all. *Reasoned Programming*. Prentice Hall, 1994.

COSTA, N. C. A. and CERRION, R. *Introdução à Lógica Elementar*. Editora da UFRGS, 1988.

DIJKSTRA, E. W. *A Discipline of Programming*. Prentice Hall, 1976.

EBBINGHAUS, H.-D. FLUM, J. and THOMAS, W. *Mathematical Logic*. Springer, 1994.

ENDERTON, H. B. *A Mathematical Introduction to Logic*. Academic Press, 2nd edition, 2001.

FITTING, Melvin *First Order Logic and Automated Theorem-Proving*. Springer, second edition, 1996.

GABBAY, Dov. *Elementary Logics: A procedural perspective*. Prentice Hall, 1998.

GABBAY, Dov and GUNTHNER, F. *Handbook of Philosophical Logic*. Kluwer Academic Pb. 1994.

GOUBAULT-LARRECQ, Jean and MACKIE, Ian. *Proof Theory and Automated Deduction*. Kluwer, 1997.

GRIES, D. *The Science of Programming*. Spring Verlag, 1981.

HUTH, Michael R. A. and RYAN, M. D. *Logic in Computer Science Modeling and Reasoning About Systems*. Cambridge University Press, 2nd edition, 2004.

KLEENE, Stephen Cole. *Mathematical logic*. John Wiley, 1967.

MANNA, Z. *Mathematical Theory of Computation*. McGrow-Hill, 1974.

MANNA, Z. and WALDINGER, R. *The Logical Basis for Computer Programming*. Addison-Wesley, Vol I, 1985.

NERODE, Anil and SHORE, Richard A. *Logic for applications*. Springer, second edition, 2005.

NOIT, John and ROHATYN, Dennis. *Lógica*. McGRaw-Hill, Makron Books, 1991.

PRIEST, Graham. *An Introduction to Non-classical Logics*. Cambridge University Press, 2001.

SOUZA, João Nunes de. *Lógica para Ciência da Computação: Fundamentos de Linguagem, Semântica e Sistemas de Dedução*. Editora Campus. Rio de Janeiro, 2002.