Lógica e Matemática Computacional

Conceitos

Para melhorar o entendimento da lógica, é necessário conhecer as definições de alguns termos importantes e muito utilizados na lógica. Mundim (2002) destaca:

- **Proposição:** Consiste em um enunciado, uma frase declarativa.
- **Premissas:** Consistem em proposições que são utilizadas como base para um raciocínio. Pode-se dizer que são as proposições do silogismo.
- **Argumento:** Conjunto de enunciados que se relacionam uns com os outros.
- **Silogismo:** Consiste em um raciocínio dedutivo (premissas) e possibilita a dedução de uma conclusão a partir das premissas.
- **Falácia:** Consiste em argumentos que logicamente estão incorretos.

A partir dos vocabulários, podemos definir os tipos de lógica existentes, entre os quais estão a lógica forma e a lógica transcendental.

Lógica Formal

Começa nos estudos de Aristóteles, na Grécia Antiga. A lógica é dita forma quando analisa e representa a forma de qualquer argumento para que possa ser considerado válido para alguma conclusão.

Para se entender a lógica forma e como formamos nosso raciocínio é importante ter em mente alguns conceitos. Uma proposição é um pensamento em forma de frase declarativa. Essa proposição pode ser verdadeira ou falsa, ela apenas garante que, com base em premissas verdadeiras, seja possível chegar a conclusões verdadeiras.

Portanto, temos que as premissas podem ser verdadeiras ou falsas. Se afirmo que o céu está claro e sem nuvens, você pode olhar pela janela e concluir se a afirmação é verdadeira ou falsa. Se estiver chovendo, você dirá que minha afirmação é falsa. Por outro lado, um argumento pode ser válido ou inválido.

Inferência é o processo que permite chegar a conclusões a partir de premissas, constituindo a argumentação lógica perfeita. A inferência, como veremos a seguir, pode ser de dois tipos: Indutiva e dedutiva. Uma inferência inválida é chamada falácia.

Em especial no século XIX, alguns matemáticos e filósofos concluíram que a lógica formal não era suficiente para que se pudesse alcançar o rigor necessário na análise dos argumentos. Por isso que foi desenvolvida a lógica simbólica, relacionada à matemática, a partir do século XIX. Ela permite a expressão das premissas e de suas relações por meio de símbolos matemáticos, construindo equações para expressar argumentos. Tal linguagem é absolutamente precisa e não dá margem a duplas interpretações.

Lógica Transcendental

É desenvolvida por toda a obra do filósofo Immanuel Kant, em especial em seu célebre livro Crítica da Razão Pura (2015). Nesse livro, Kant discute que nosso conhecimento, o conhecimento humano, parte de duas fontes principais. A primeira trata da receptividade das impressões por meio de nossos sentidos, a segunda fonte é relativa à faculdade de conhecer um objeto por representações mentais, a partir do pensamento.

Desse modo, a lógica transcendental opera a partir das representações, dos conceitos e não das coisas em si. Trata-se de uma investigação sobre as representações a priori, as categorias, os conceitos puros em relação aos objetos, enquanto a lógica geral se volta para a forma lógica do pensamento.

Kant distingue dois tipos de conhecimento:

- **Conhecimento Empírico:** Também chamado conhecimento ao posteriori está relacionado ao que é obtido por meio de nossos sentidos, à observação, à experimentação, com base na presença real de determinado objeto.

Lógica Dedutiva

É aquela que parte de premissas afirmativas ou leis mais gerais permitindo a obtenção de verdades menos gerais ou particulares. Vamos a um exemplo de inferência dedutiva ou dedução?

Todo o analista de sistemas sabe programar.

Mariana é analista de sistemas.

Portanto, Mariana sabe programar.

Aqui, partimos de uma informação geral sobre as habilidades dos analistas de sistemas para concluir sobre as habilidades de Mariana. Com base na afirmativa geral, tomada como verdadeira, a conclusão é inevitável. Chamamos silogismo esse tipo de argumentação lógica.

Lógica Indutiva

Se preocupa com argumentos que permitem conclusões gerais a partir de casos particulares. Vamos a um exemplo de inferência indutiva ou indução?

Mariana é analista de sistemas e sabe programar.

Enzo é analista de sistemas e sabe programar.

Sabrina é analista de sistemas e sabe programar.

Portanto, todos os analistas de sistemas sabem programar.

Observe que, ao consultar dezenas ou centenas de analistas de sistemas, chegamos a uma conclusão geral com relação a eles. Um cuidado a ser tomado com a lógica indutiva é que um único contraexemplo é capaz de invalidar todo um raciocínio.

Desenvolvimento Histórico

Antes de Aristóteles, filósofos e pensadores já aplicavam argumentos lógicos, porém de maneira intuitiva, sem que houvesse necessariamente uma reflexão sobre tais argumentos. Aristóteles, porém, foi o primeiro a reconhecer que a lógica poderia ser examinada e desenvolvida, constituindo-se assim como uma ferramenta do pensamento que nos ajudaria a compreender melhor o mundo.

Quando discorremos sobre o Período Aristotélico, estamos nos referindo à chamada Lógica Clássica, que é regida, basicamente, por três princípios: O da identidade, o da não contradição e o do terceiro excluído. Todos esses três princípios são facilmente compreendidos. O mais importante é que esses princípios funcionam como leis que permitirão a formulação de conclusões lógicas sobre proposições, mesmo que não estejamos familiarizados com a natureza daqui que está sendo discutido (Zegarelli, 2013).

- **Princípio da Identidade:** Estabelece que todo objeto é idêntico a si mesmo. O princípio da identidade mostra que qualquer proposição no formato “A é A” tem que ser verdadeira.
- **Princípio da Não-Contradição:** Busca a especifidade de cada coisa, ou seja, é impossível que ela seja e não seja ao mesmo tempo. Isso significa que uma proposição não pode ser verdadeira e falsa ao mesmo tempo.
- **Princípio do Terceiro Excluído:** Afirma que toda proposição é verdadeira ou falsa, não havendo uma terceira possibilidade para valoração da proposição.

Zegarelli (2013) aponta que após o declínio da civilização grega, o estudo da lógica passou por um longo período de ostracismo, com alguns renascimentos esporádicos durante o Império Romano e a Europa Medieval. Segundo Santos (2014), nos séculos XII e XIII, por exemplo, há um ressurgimento do interesse motivado, ao que parece, pelo desenvolvimento de um gênero específico da lógica medieval que se ocupava com o estudo de paradoxos semânticos.

Paradoxo é um tipo de pensamento ou argumento que, apesar de aparentemente correto, apresenta uma conclusão ou consequência contraditória, ou em oposição a determinadas verdades aceitas (Japiassú, Marcondes, 2006).

É apenas por volta do século XVIII, com o advento do Iluminismo na Europa, quando a era da fé vai gradualmente dando lugar à era da razão, que a lógica volta a figurar como objeto de maior interesse de cientistas e filósofos. De acordo com Zegarelli (2013), cientistas como Isaac Newton e filósofos como René Descartes passaram a procurar respostas sobre o funcionamento do universo, indo além dos ensinamentos da igreja. Com isso, a lógica ressurgiu no pensamento científico para se estabelecer como uma ferramenta essencial da razão.

É a partir desse contexto que temos o desenvolvimento do chamado Período Booleano (1840-1910) de desenvolvimento da lógica. No final do século XIX, matemáticos desenvolveram a Lógica Formal, também chamada de Lógica Simbólica, na qual símbolos computáveis substituem palavras e proposições (Zegarelli, 2013). Os três maiores expoentes desse período foram:

- **George Boole:** Inventor da chamada Álgebra Booleana, que foi o primeiro sistema totalmente detalhado que lida com a lógica como cálculo. Colocamos de uma maneira bem simples, podemos dizer que a Álgebra Booleana se caracteriza por utilizar apenas dois números (dígitos), 0 e 1, que significam, respectivamente, falso e verdadeiro, e que por meio de propriedades essenciais dos operadores lógicos e de conjuntos oferece uma estrutura para se lidar com proposições.
- **George Cantor:** Idealizador da Teoria de Conjuntos. A Álgebra dos Conjuntos, advinda da Teoria de Conjuntos, com operadores particulares como União ($\cup$) e Intersecção ($\cap$) não serviu apenas como uma estrutura de linguagem para a lógica forma, mas também como alicerce de toda a Matemática Moderna.
- **Gottlob Frege:** Criador da chamada Lógica Matemática. Inspirado nas ideias e notações de Leibniz, Frege reformulou toda a lógica tradicional, construindo um sistema para apresentá-

la em linguagem matemática. É com base em suas obras que se desenvolveram o cálculo proposicional e o cálculo de predicados. Uma de suas maiores contribuições foi a invenção do quantificador e a utilização de variáveis para formalizar a generalidade da linguagem natural (Japiassú, Marcondes, 2006).

Conectivos

Segundo Alencar Filho (2002), quando pensamos, efetuamos muitas vezes certas operações sobre proposições, chamadas operações lógicas. Estas operações obedecem à regras de um cálculo, denominado cálculo proposicional, semelhante ao da aritmética sobre números. Os conectivos sentenciais correspondem a várias palavras nas linguagens naturais que servem para conectar proposições declarativas. Os principais conectivos (operações lógicas fundamentais) são representados atualmente pelos seguintes símbolos:

- **~:** O til corresponde à operação lógica negação. Alguns autores também utilizam o símbolo ¬ para designar negação.
- **∧:** A cunha corresponde à operação lógica conjunção. Em programação, a conjunção é representada pela palavra AND, ou pelo símbolo &, que corresponde ao conectivo E.
- **∨:** A letra v corresponde à operação lógica disjunção. Equivale à palavra OU em seu sentido inclusivo. Em programação, a conjunção também é representada pela palavra OR.
- **→:** A seta corresponde à operação condicional. Em português, corresponde à relação “se…, senão…”.
- **↔:** A dupla seta corresponde à operação bicondicional. Em português, corresponde à relação “se, e se somente se, …”.

Listas

Ao se deparar com um problema de contagem, normalmente será preciso determinar quantos elementos existem em um conjunto finito. Determinar essas quantidades de recursos finitos podem gerar questões difíceis de serem respondidas (Gersting, 2017). Por isso, vamos inicialmente nos familiarizar com o conceito de lista.

Uma lista é uma sequência ordenada de objetos (Scheinerman, 2015). Costumamos representar uma lista abrindo parênteses e apresentando cada elemento da lista, separando-os por vírgula. Por exemplo, a lista (2, 4, 8, 16) é uma lista cujo primeiro elemento é o número 2, o segundo elemento é o número 4, o terceiro elemento é o número 8 e o quarto elemento é o número 16. A ordem com a qual os elementos figuram na lista é significativa. Assim ,a lista (2, 4, 8, 16) na é a mesma que a lista (4, 2, 16, 8). Embora os elementos que compõem a lista sejam os mesmos, a forma pela qual foram arranjados (ordem) é diferente. Também é importante destacar que uma lista pode conter elementos repetidos, como (3, 4, 5, 5, 6), em que o número 5 aparece duas vezes.

Em uma lista, chamamos de comprimento ao número de elementos que a compõe. Quando a lista tem apenas dois elementos ela recebe o nome de par ordenado. E uma lista vazia é uma lista cujo comprimento é igual a zero.

Outro tipo de lista que podemos ter, são listas de dois elementos em que há n escolhas para o primeiro elemento e, para cada uma dessas escolhas, há m escolhas do segundo elemento. Então o número de tais listas é n x m.

Um caso especial envolvendo a contagem de listas consiste no problema de se determinar quantas listas de comprimento n extraídas de um universo de n objetos, em que não se permitem repetições, podem ser formadas. Em outras palavras, de quantas maneiras diferentes podemos dispor n objetos em uma lista, usando cada objeto exatamente uma única vez?

Podemos estender o princípio da multiplicação para listas com dois elementos para esta situação, determinando o total de listas como:

$n \times (n - 1) \times (n - 2) \ldots (n - n + 1) = n \times (n - 1) \times (n - 2) \ldots (2) \times (1)$

Essa expressão ocorre com frequência em matemática e recebe um nome e símbolo especiais: Chama-se fatorial de n e representamos por n!. Por exemplo, $6! = 6 \times 5 \times 4 \times 3 \times 2 \times 1 = 720$. Por definição, considera-se que $1! = 1$ e $0! = 1$ (Scheinerman, 2015).

Árvore de Decisão

Outra forma de representarmos os possíveis resultados de uma ordenação (listas) é a utilização de um diagrama chamado Árvore de Decisão. Uma árvore de decisão é uma estrutura hierárquica que representa um mapeamento de possíveis resultados de uma série de escolhas relacionadas. A árvore de decisão é uma importante ferramenta para auxiliar na tomada de decisões, bem como visualizar as ramificações e as consequências. Elas podem ser usadas tanto para conduzir diálogos informais quanto para mapear um algoritmo que prevê a melhor decisão (escolha), matematicamente, além de auxiliar na criação de planos de ação.

Em geral, uma árvore de decisão inicia a partir de um único nó de origem (chamado de nó raiz), que se divide em possíveis resultados. O nó raiz é representado por um elemento no topo da árvore e representa o todo ou uma amostra de alguma categoria e expressa nós de decisão, em que, a partir do nó raiz, temos ramificações que levam a escolhas de um resultado (decisão).

Combinatória

Em Combinatória, existem diferentes tipos de agrupamentos (ordenados ou não) que recebem os nomes específicos de Arranjos, Permutações e Combinações. Apesar de ser possível determinar esses agrupamentos de maneira intuitiva, existem fórmulas matemáticas que nos auxiliam a realizar essa tarefa. Vamos, então, aprender a utilizá-las, considerando os agrupamentos (Arranjos, Permutações e Combinações) simples, isso é, formados por elementos distintos.

Arranjos

De acordo com Iezzi et. all. (2004), dado um conjunto com n elementos distintos, chama-se arranjo dos n elementos, tomados p a p, a qualquer sequência ordenada de p elementos distintos escolhidos entre os n existentes. Para determinar o número de arranjos podemos utilizar a fórmula.

Exemplo: Considere o conjunto $A = \{1, 2, 3, 4\}$. Vamos determinar o número de arranjos desses quatro elementos $(n = 4)$ tomados dois a dois $(p = 2)$. Utilizando a fórmula para determinação do número de arranjos.

Permutação

Um caso especial de arranjo, denominado permutação, é obtido quando dado um conjunto com n elementos distintos, selecionamos exatamente n elementos para formar a sequência ordenada.

Exemplo: Considere o problema de se determinar de quantas maneiras seis pessoas A, B, C, D, E e F podem ser dispostas em uma fila indiana. Cada maneira de compor a fila é uma permutação das seis pessoas, pois qualquer fila obtida é uma sequência ordenada na qual comparecem sempre as seis pessoas. Ao utilizarmos a fórmula do número de arranjos, percebemos que neste caso $n = p$.

Ou seja, existem 720 maneiras distintas para organização dessa fila.

Combinação

Há ainda outra forma de agrupamento, denominada combinação, que considera cada sequência obtida como um conjunto não ordenado. Dado um conjunto com n elementos distintos, chama-se combinação dos n elementos, tomados p a p, a qualquer subconjunto formado por p elementos distintos escolhidos entre os n existentes. Para determinar o número de combinações, podemos utilizar a fórmula.

Exemplo: Considere que dos cinco funcionários A, B, C, D e E de uma empresa do setor de Tecnologia de Informação, três serão promovidos. Queremos determinar todas as combinações desses cinco funcionários, tomados dois a dois. Podemos calcular o número de combinações.

Álgebra de Conjuntos

A álgebra de conjuntos é um importante ramo da matemática e com aplicações em diferentes áreas de conhecimento, entre elas a computação. A linguagem de conjuntos se caracteriza por ser uma linguagem clara, concisa, rigorosa e que não dá margens a interpretações equivocadas. Por apresentar essas características, ela é utilizada na organização de informações e resolução de problemas ligado a várias áreas, como a computação.

O livro de Benzecry e Rangel (2008), inteiramente dedicado à álgebra de conjuntos, traz em sua primeira parte, nas páginas 1 a 6, um resumo com os principais tipos de conjuntos e as diferentes formas de representá-los. É uma ótima oportunidade para aprofundamento dos estudos.

Conjuntos

Conjuntos podem ser definidos como coleções não ordenadas de objetos que podem ser, de alguma forma, relacionados (Ferreira, 2001). Em geral, objetos de um mesmo conjunto gozam de uma propriedade em comum. Costuma-se utilizar letras maiúsculas do nosso alfabeto para representar os conjuntos.

Para descrever determinado conjunto, é necessário identificar seus elementos. Para tanto, pode-se proceder de três maneiras distintas:

1. Listar todos os elementos do conjunto.
2. Indicar os primeiros elementos do conjunto (presumindo que os elementos do conjunto possam ser ordenados) que denotem um padrão para uma listagem indefinida.
3. Escrever uma propriedade que caracterize os elementos que constituem o conjunto. Por exemplo: $C = \{x|x$ é um número inteiro e $4 < x <= 9\}$. Escrever a propriedade característica dos elementos de um conjunto por meio de palavras é a maneira mais usual de descrever um conjunto. Isso porque, muitas vezes, ao se trabalhar com conjuntos que poissuem um número muito grande de elementos (ou até mesmo conjuntos infinitos), a listagem de todos os elementos do conjunto não se torna viável.

Há ainda uma maneira alternativa de representar conjuntos com forte apelo visual. Trata-se dos Diagramas de Venn.

Diagrama de Venn

John Venn (1834-1923) foi um matemático inglês, tendo-se licenciado na Universidade de Cambridge onde, depois, ensinou Lógica e Teoria das Probabilidades. Venn introduziu os diagramas em seus trabalhos baseados nos círculos eulerianos, por isso, alguns autores referem-se aos diagramas de Venn como diagramsa de Eulen-Venn (Novaes, 2014).

John Venn introduziu os diagramas em um trabalho de lógica formal publicado em 1880, na Philosophical Magazine and Journal of Science, intitulado On the Diagrammatic and Mechanical Representation of Propositions and Reasonings (Da representação mecânica e diagramática de proposições e raciocínios).

Na introdução desse trabalho, Venn afirmou:

"Esquemas de representação diagramática têm sido tão familiarmente introduzidos nos tratados de lógica durante o último século, que se podem supor que muitos leitores, mesmo aqueles que não fizeram nenhum estudo profissional de lógica, possam ter familiaridade com a noção geral de tais objetos."

Porém, a primeira referência escrita conhecida do termo Diagrama de Venn surgiu apenas em 1918, no livro A Survey of Symbolic Logic, do lógico Clarence Inving Lewis (Novaes, 2014).

Os diagramas de Venn consistem em círculos (que podem estar intersectados), os quais representam os conjuntos. No interior dos círculos são listados os elementos do conjunto. Por exemplo, o conjunto $C = \{x|x$ é um número inteiro e $4 < x <= 9\}$ pode ser representado pelo diagrama.

Um conjunto é chamado de finito quando sua cardinalidade é um número inteiro, caso contrário, é chamado de infinito. Um conjunto é chamado de conjunto vazio quando sua cardinalidade é igual a zero, ou seja, é um conjunto desprovido de elementos.

Os conceitos estudados nessa aula também são apresentados de forma bastante didática por Souza (2016) na unidade 1 de seu livro, nas páginas 1 a 12. São abordadas as diferentes formas de notação e representação de conjuntos. Vale a pena conferir.

Álgebra de Conjuntos

Em matemática, quando nos referimos a operações, automaticamente nos recordamos das operações numéricas fundamentais (adição, subtração, multiplicação e divisão), porém, em teoria de conjuntos, várias operações podem ser realizadas. Podemos, por exemplo, somar ou multiplicar os elementos de conjuntos, reuni-los, considerar apenas os elementos comuns, enfim, há uma série de operações que podem ser feitas. Entre essas operações, as mais fundamentais são denominadas união ($\cup$) e interseção ($\cap$). Nesta aula, focaremos esses dois tipos de operações.

Diagrama de Venn

Os diagramas de Venn podem ser utilizados para ilustrar as operações binárias de união e interseção de conjuntos.

Sejam os conjuntos $A = \{10, 11, 12, 13, 14, 15\}$ e $B = \{13, 14, 15, 16, 17, 18, 19\}$, o conjunto $A \cup B$ consiste no conjunto formado por elementos de A e de B.

$A \cup B = \{10, 11, 12, 13, 14, 15, 16, 17, 18, 19\}$

Repare que há elementos pertencentes a ambos os conjuntos, porém, ao efetuarmos a operação união, esses elementos são contabilizados uma única vez.

Já o conjunto $A \cap B$ consiste no conjunto formado pelos elementos comuns aos conjuntos A e B.

$A \cap B = \{13, 14, 15\}$

Você sabia que é impossível criar um diagrama de Venn para quatro conjuntos A, B, C e D utilizando apenas círculos? Um diagrama de Venn para n conjuntos consiste basicamente de n curvas simples e fechadas no plano que determinam uma região conexa para cada uma das interseções que os conjuntos formam (Cerioli, 2004).

Por meio de círculos, é possível fazer o diagrama para 1, 2 ou 3 conjuntos. Entretanto, é possível desenhar diagramas de Venn para mais de 3 conjuntos, desde que sejam utilizadas outras formas geométricas, diferentes do círculo. Para quatro conjuntos, podemos utilizar o diagrama representado.

O livro de Barbosa, em seu segundo capítulo, páginas 48 a 57, trata da álgebra de conjuntos e destaca o tratamento das relações entre conjuntos ilustradas com diagramas de Venn. Lá, você encontrará exemplos e exercícios já resolvidos, bem como poderá colocar o conhecimento em prática resolvendo os problemas propostos.

Aplicações de Teoria dos Conjuntos

Nesta aula, vamos estudar como o diagrama de Venn pode ser utilizado para demonstrar relações arbitrárias entre conjuntos.

Já sabemos que os diagramas de Venn também podem ser usados para resolver problemas sobre cardinalidade de conjuntos, isto é, problemas que envolvem a contagem do número de elementos de conjuntos finitos. Mas, segundo Novaes (2014), o diagrama de Venn não é apenas um esquema para ajudar o raciocínio. Ele foi concebido como uma representação diagramática capaz de atender a todas as possíveis relações lógicas entre as classes em estudo, sendo úteis, inclusive, para demonstrar relações arbitrárias entre conjuntos.

Relação Arbitrária entre Conjuntos

Considere os conjuntos A e B, em que os compartimentos do diagrama representam as partes disjuntas do universo: Utilizaremos uma numeração binária, composta apenas pelos algarismos 0 e 1, em que o primeiro algarismo é 0 ou 1, conforme um objeto desse compartimenta pertença ou não ao conjunto A, enquanto o segundo algarismo é 0 ou 1, conforme um objeto desse compartimento pertença ou não ao conjunto B.

A numeração binária mostra que $2^2 = 4$ compartimentos esgotam todas as possibilidades lógicas para um objeto do universo. O número 10 (lê-se: um, zero) representa objetos que pertençam exclusivamente ao conjunto A. O número 01 (zero, um) representa objetos que pertençam exclusivamente ao conjuntos B. Já o número 11 (um, um) representa objetos que pertençam simultaneamente aos conjuntos A e B (interseção de A e B). Por fim, o número 00 (zero, zero) representa objetos que não pertencem a nenhum dos conjuntos A e B.

Tabelas-Verdade

Uma maneira equivalente de representar essa relação arbitrária é a utilização de tabelas-verdade. Confira a tabela-verdade para a realização arbitrária entre os dois conjuntos:

	x ∈ A	**x ∈ B**	**x ∈ A ∩ B**
00	F	F	F
01	F	V	F
10	V	F	F
11	V	V	V

Na primeira coluna da tabela, indicamos os $2^2 = 4$ compartimentos que compõe o diagrama de Venn para dois conjuntos, representados por números binários. Na segunda coluna anotamos V (verdadeiro) ou F (falso) conforme o objeto representado pelo seu respectivo número binário pertença, ou não, ao conjunto A. Na terceira coluna anotamos V (verdadeiro) ou F (falso) conforme o objeto representado pelo seu respectivo número binário pertença, ou não, ao conjunto B. Finalmente, na última coluna, anotamos V (verdadeiro) ou F (falso) conforme o objeto representado pelo seu respectivo número binário pertença, ou não, ao conjunto A ∩ B.

Esperamos que você tenha compreendido a utilização de diagrama de Venn tanto na representação de operações quanto na representação de relações arbitrárias entre conjuntos. Compreender a natureza dessas operações é componente importante no desenvolvimento do raciocínio computacional.

Introdução à Lógica Proposicional

Segundo Machado e Cunha (2008), o objetivo fundamental de um curso de lógica é desenvolver a competência na argumentação, compreender as razões próprias e dos outros nas tomadas de posição diante dos acontecimentos e nas decisões. Mas afinal, o que esse desenvolvimento, que parece ser linguístico, tem a ver com nosso universo computacional? A verdade é que tem tudo a ver. Construiremos algoritmos capazes de tomar decisões, e para isso precisaremos implementar regras baseadas na lógica formal. Isso mesmo, aquela lógica forma desenvolvida por Aristóteles entre 300 e 400 anos antes de Cristo (Machado, Cunha, 2008). Assim, nessa aula vamos estudar algumas características para desenvolveras tomadas de decisões.

Premissas e Conclusões

Para relembrarmos rapidamente, Aristóteles desenvolveu um método no qual ele separa a forma (podemos entender como regras) do conteúdo nas argumentações. Ou seja, no método formal, “não são considerados os conteúdos das sentenças componentes de um argumento, mas apenas a forma de articulá-las ou o modo como umas são deduzidas das outras” (Machado, Cunha, 2008).

Na lógica computacional, vamos utilizar as mesmas regras da lógica formal, porém iremos valorar os conteúdos, como verdadeiro ou falso, a fim de extrair nossas conclusões.

Em nosso cotidiano, usamos a linguagem natural para nos expressar por meio de frases, que em alguns casos podem ser argumentativas sendo assim compostas por premissas e conclusões.

Vejamos um exemplo extraído de Machado e Cunha (2008). Observe o argumento de uma professora sobre o desempenho de um certo aluno: “É lógico que Sérgio será aprovado nos exames, pois ele é inteligente e estuda muito e todos os alunos inteligentes e estudiosos são aprovados”. Esse argumento foi construído embasado por premissas (razões) e que levam a uma única conclusão.

Premissas (Razões)	Conclusão
Sérgio é inteligente.	Sérgio será aprovado.
Sérgio estuda muito.	
Todos os alunos inteligentes e estudiosos são aprovados.	

Veja que separamos a frase premissas e conclusão. Nesse caso, três premissas permitiram chegar a uma conclusão coerente. Extrair essa conclusão do argumento só foi possível devido às regras da lógica proposicional, que por meio de premissas e conectores extraem-se resultados lógicos. Fazer essa separação (premissa/conclusão) é muito importante, pois nem toda frase é um argumento.

Sentenças na Lógica Computacional: Argumento e Classificação

Para uma sentença ser argumento é preciso existir uma conclusão, logo, nem toda frase é um argumento. Por exemplo, a frase “segure firme”, não possui premissas e conclusões, pois trata-se de uma sentença imperativa (ordem) ou então a frase “você pode abrir a porta” também não é um argumento, pois estamos diante de uma sentença interrogativa. As sentenças exclamativas, como por exemplo, “que lindo”, “parabéns” também não são consideradas argumentos. No estudo da lógica, além de distinguir se uma frase é ou não um argumento, também é importante distinguirmos se uma sentença pode ou não ser classificada como verdadeira ou falsa (não ambas ao mesmo tempo). Por exemplo, considere as frases:

1. O Brasil é um país da América Latina.
2. Minas Gerais é um estado do Nordeste.
3. São Paulo é a capital do Paraná.
4. Três mais um é igual a quatro.
5. Que horas são?

As quatro primeiras frases podem ser classificadas (valoradas) em verdadeira (V) ou falso (F). Veja:

1. Verdadeira.
2. Falso.
3. Falso.
4. Verdadeira.

Mas a quinta frase não pode ser valorada em V ou F, pois a resposta é um certo horário.

Essa distinção entre os tipos de sentenças é crucial para o estudo da lógica, pois uma frase que pode ser classificada como verdadeira ou falsa (não ambas ao mesmo tempo) é chamada proposição (Machado, Cunha, 2008, Souza, 2016, Gersting, 2017).

Proposição é uma sentença declarativa que pode ser classificada como verdadeira ou falsa, jamais ambas ao mesmo tempo. Ou seja, não pode haver dúvida quanto à classificação da sentença. Também podemos dizer que trata-se de uma classificação binária, pois só existem dois resultados possíveis: V ou F, ou ainda 1 ou 0.

Quando extraímos as proposições simples, podemos fazer adequações nos verbos, o mesmo acontece quando usamos proposições simples para fazer as compostas. Reescrevendo a frase utilizando a notação simbólica teremos: Se A então R. Nesse caso, temos duas palavras fazendo a ligação entre as proposições: Se… Então.

Essas "palavras" usadas para unir as proposições simples são os conectivos (ou conectores) lógicos e influenciam a valoração de uma proposição composta. Os conectivos disponíveis para fazer a conexão são: (I) e, (II) ou, (III) não, (IV) se… então, (V) se, e somente se (Bispo, Castanheira, 2011).

Conectivo Lógico de Conjunção – E

Vamos começar estudando o conectivo lógico "e", que também pode ser visto na literatura escrito em inglês *AND* ou ainda por meio do seu símbolo $\wedge$. Essa operação lógica é chamada conjunção e sua valoração será verdadeira somente quando ambas as proposições simples forem verdadeiras. Resumindo, se A e B forem proposições simples verdadeiras, a proposição composta $A \wedge B$ (lê-se "A e B") será verdadeira.

Conectivo Lógico de Disjunção – Ou (Inclusivo)

Outro conector muito utilizado tanto na lógica formal, quanto na computacional é "ou", que também pode ser visto na literatura escrito em inglês *OR* ou ainda por meio do seu símbolo $\vee$. Essa operação lógica é chamada de disjunção e seu operador lógico pode ser utilizado de duas formas distintas: Inclusivo ou exclusivo.

O operador lógico de disjunção usado na forma inclusive terá sua valoração falsa somente quando ambas as proposições simples forem falsas. Resumindo, se A e B forem proposições simples falsas, a proposição composta $A \vee B$ (lê-se "A ou B") será falsa, nos demais casos a valoração é verdadeira.

Operador Lógico de Negação – Não

Os operadores lógicos de conjunção e disjunção são binários, ou seja, juntam duas expressões para formar uma nova proposição. O operador lógico de negação é unário, ou seja, ele não junta duas proposições, ele age sobre uma única proposição (que pode ser resultado de uma operação binário). A palavra usada para fazer a negação é o "não" que também pode ser visto na literatura em inglês *NOT*, ou ainda de forma simbólica como ~, ¬, '. Os dois primeiros símbolos são usados antes da letra que representa a proposição, já o terceiro é usado depois da letra. Por exemplo, as expressões: $\sim A$, $\neg B$, C' representam as negações das proposições A, B, C.

A operação lógica de negação troca o valor-verdade da proposição. Ou seja, se a proposição é verdadeira, quando acompanhada do operador de negação passará a ser falsa, por outro lado, se ela for falsa passará a ser verdadeira.

Nessa aula estudamos algumas características da lógica formal. Continue estudando este conceito, pois sem essa lógica não conseguiríamos construir os algoritmos computacionais.

Conectivos Lógicos

Uma proposição comporta pode ser criada fazendo a conjunção de duas proposições simples, neste caso são utilizadas as palavras “e”, “mas”, “no entanto”, dentre outras para fazer a conexão. Também podemos criar uma proposição composta fazendo a disjunção de duas proposições simples, nesse caso usamos a palavra “ou” para a conexão. A disjunção possui uma particularidade, ela pode ser inclusive ou exclusiva.

Conectivo Lógico de Disjunção – Ou (Exclusivo)

Considere as seguintes proposições simples:

1. João é estudante.
2. João é trabalhador.
3. João é paulista.
4. João é carioca.

Agora vamos usar as proposições simples A, B, C, D, criar as compostas usando a disjunção. Observe:

1. João é estudando ou é trabalhador.
2. João é paulista ou é carioca.

A proposição 1 representa uma disjunção inclusiva, pois João pode ser estudante e também trabalhador. Já a proposição 2 é uma disjunção exclusiva, pois João não pode ser paulista e carioca, ele só pode ser um dos dois.

Segundo Souza (2016), a disjunção inclusiva é representada pelo símbolo $\vee$, ou seja, a proposição 1, pode ser escrita como $A \vee B$. Já a disjunção exclusiva é representada pelo símbolo $\underline{\vee}$. Ou seja, a proposição 2 pode ser escrita como $C \underline{\vee} D$.

Na valoração de uma disjunção exclusiva, o resultado será verdadeiro se, e somente se, apenas uma das proposições simples forem verdadeiras.

Conectivo Condicional (Implicação Lógica) – Se… Então

Dadas as proposições simples A, B, elas formam uma condicional (ou implicação lógica) se for possível construir a estrutura: Se A, então B (Bispo, Castanheira, 2011). A primeira proposição é chamada antecedente, e a segunda consequente. A condicional significa que a verdade da primeira proposição implica a verdade da segunda proposição (Gersting, 2017). O símbolo usado para representar a implicação lógica é o $\rightarrow$, logo a regra se A então B, pode ser escrita como $A \rightarrow B$.

Na valoração do condicional, se o antecedente e o consequente forem verdadeiros então o resultado será verdadeiro. Ou seja, ($V \rightarrow V = V$). Porém, se o antecedente for verdadeiro e o consequente falso, o resultado será falso ($V \rightarrow F = F$). Vejamos um exemplo:

1. O interruptor da sala foi desligado.
2. A luz da sala apagou.
3. $A \rightarrow B$.

A proposição 3 deve ser traduzida como “Se o interruptor da sala for desligado, então a luz se apagará”.

Se as duas proposições realmente acontecerem, então temos o caso $V \rightarrow V = V$, ou seja, 3 é verdade. Porém, se o interruptor for desligado, mas por algum motivo a luz não se apagar, então teremos o caso $V \rightarrow F = F$, ou seja, 3 é falso.

Conectivo Bicondicional – Se, e Somente Se

Dadas as proposições simples A, B, elas formam uma bicondicional se for possível construir a estrutura: A se, e somente se, B (Bispo, Castanheira, 2011). O símbolo usado para representar esse conectivo é o $\leftrightarrow$, então a expressão A se, e somente se B, pode ser expressa simbolicamente por $A \leftrightarrow B$. Vejamos um exemplo. Considere as proposições a seguir:

1. Carlos receberá o dinheiro.
2. Carlos completará o trabalho.
3. $A \leftrightarrow B$.

A proposição 23, deve ser traduzida como “Carlos receberá o dinheiro se, e somente se, completar o trabalho”.

A valoração do conectivo bicondicional será verdadeira quando o valor-lógico das duas proposições forem iguais, tanto para verdadeiro como para falso. Ou seja, $V \leftrightarrow V = V$ e $F \leftrightarrow F = V$.

Equivalência Lógica

No conectivo bicondicional apresentamos uma definição trazida por Gersting (2017) que diz que a fórmula (I) $A \leftrightarrow B$ é um atalho para (II) $(A \rightarrow B) \wedge (B \rightarrow A)$. A autora usa o termo “atalho” porque o resultado da fórmula (I) é igual ao da (II) para todas as combinações possíveis de entradas, isso acontece porque estamos diante de uma equivalência lógica.

Além do conectivo bicondicional, existem outras importantes equivalências lógicas que serão estudadas em outra seção. Nesse momento, vamos apresentar outras duas equivalências lógicas, chamadas de Leis de De Morgan que foram obtidas e demonstradas pelo matemático inglês Augustus De Morgan. As duas equivalências são:

1. $\neg(A \vee B) \Leftrightarrow \neg A \wedge \neg B$.
2. $\neg(A \wedge B) \Leftrightarrow \neg A \vee \neg B$.

Observe na fórmula I que, do lado esquerdo da equivalência, a negação está sendo aplicada ao resultado de uma disjunção, enquanto do lado direito a negação afeta cada uma das proposições. Para treinarmos, vamos verificar a veracidade da fórmula I. Para ser uma equivalência, o resultado precisa ser igual para todas as combinações possíveis para as proposições A e B: (1) $A = V$ *e* $B = V$, (2) A $= V$ *e* $B = F$, (3) $A = F$ *e* $B = V$, (4) $A = F$ *e* $B = F$. Vamos testar essas quatro combinações de entrada, primeiro para a fórmula $\neg(A \vee B)$. Será ilustrado cada passo para a valorização da fórmula, para cada combinação possível de entrada. Como resultado lógico temos que para as entradas (1), (2) e (3) o resultado é F, já para a entrada (4) o resultado é V.

Métodos Dedutivos e Inferência Lógica

Nessa aula será apresentado duas regras de equivalência para a lógica proposicional: A regra de equivalência e a regra de inferência.

Regras de Equivalência de Dedução para a Lógica Proposital

As regras de dedução são divididas em dois tipos: Regras de equivalência e regras de inferência. Lembrando que duas fbfs são equivalentes, quando todas as combinações possíveis de entradas geram o mesmo resultado de saída para ambas a fbfs, as regras de equivalência serão usadas quando uma fbf (que pode ser uma hipótese ou resultado de uma regra) pode ser substituída por outra fbf, mantendo o resultado lógico. Por exemplo, se considerarmos a fbf que traduz uma das leis de De Morgan: $\neg(A \vee B) \Leftrightarrow \neg A \wedge \neg B$, em uma situação adequada podemos substituir a fbf $\neg(A \vee B)$ por $\neg A \vee \neg B$, pois ambas são equivalentes. No quadro estão elencadas as regras de equivalência que iremos utilizar.

	Expressão (fbf)	Equivalente (fbf)	Nome/Abreviação
1	$P \vee Q$	$Q \vee P$	Comutatividade/con
2	$P \vee Q$	$Q \wedge P$	
3	$(P \vee Q) \vee R$	$P \vee (Q \vee R)$	Associatividade/ass
4	$(P \wedge Q) \wedge R$	$P \wedge (Q \wedge R)$	
5	$\neg(P \vee Q)$	$\neg P \wedge \neg Q$	Leis de De Morgan/De Morgan
6	$\neg(P \wedge Q)$	$\neg P \vee \neg Q$	
7	$P \rightarrow Q$	$\neg P \vee Q$	Condicional/cond
8	P	$\neg(\neg P)$	Dupla Negação/dn
9	$P \leftrightarrow Q$	$(P \rightarrow Q) \wedge (Q \rightarrow P)$	Definição de Equivalência/que

No quadro você pode verificar que existem seis conjuntos de regras de dedução, sua utilização será da seguinte forma: Se tivermos uma expressão como da linha 1, $P \vee Q$, quando necessário, podemos substituí-la por $Q \vee P$, pois essas fbfs são equivalentes e trata-se da propriedade da comutatividade. O contrário também é válido, quando aparecer $Q \vee P$, podemos substituir por $P \vee Q$. Esse processo de substituir uma fbf por outra, é o mesmo para todas as demais regras apresentadas.

Para finalizar essa aula é importante ter em mente que nas regras de equivalência, as colunas podem ser usadas nos dois sentidos. Já nas regras de inferência, só existe um sentido: A fbf da coluna "De" pode ser substituída pela coluna de "Podemos Deduzir", mas o contrário não é verdade.

Construção da Tabela Verdade

Verificação de Satisfazibilidade e Validade de Fórmulas

Tanto o hardware como software computacional são baseados na lógica computacional, que, por sua vez, é baseada na lógica forma. Tudo é possível porque a lógica computacional produz somente resultados binários para suas fórmulas, ou seja, verdadeiro ou falso.

As fórmulas podem ora ser valoradas em 0, em cujo caso a valoração falsifica a fórmula, ora ser valoradas em 1, em cujo caso a valoração satisfaz a fórmula. Estes fatos motivam a classificação das fórmulas de acordo com o seu comportamento diante de todas as valorações possíveis de seus átomos. Um dos grandes desafios da computação é encontrar métodos eficientes para decidir se uma fórmula é satisfazível/insatisfazível, ou se é válida/falsificável. Um dos primeiros métodos propostos na literatura para a verificação de satisfazibilidade e validade de fórmulas é o método da tabela da verdade.

Tabela Verdade

A tabela verdade é um mecanismo que permite valorar fórmulas de forma genérica a partir de entradas binárias e conectores lógicos.

A figura mostra o esquema geral para uma tabela verdade. Nas primeiras colunas coloca-se as proposições (quantas forem necessárias testar), em seguida, coloca-se as operações lógicas que se deseja valorar. Nas linhas são colocadas os valores lógicos (V – F) tanto para as proposições quanto para os resultados das fórmulas.

A quantidade de linhas (combinações) aumenta exponencialmente com a quantidade de proposições seguindo a regra 2^n, em que n é o número de proposições. Portanto, para duas proposições tem-se 2^2 = 4 linhas, para três proposições tem-se 2^3 = 8 linhas, para quatro proposições tem-se 2^4 = 16 linhas, e assim por diante.

Tabela Verdade da Conjunção (AND – E)

O conectivo lógico de conjunção (AND – E) é utilizado para realizar uma operação binária entre duas proposições quando se deseja obter um resultado verdadeiro se, e somente se, as duas proposições forem verdadeiras. Confira a seguir:

A	B	A ∧ B
V	V	V
V	F	F
F	V	F
F	F	F

Tabela Verdade da Disjunção (OR – OU)

O conectivo lógico de disjunção (OR – OU) é utilizado para realizar uma operação binária entre duas proposições quando se deseja obter um resultado falso se, e somente se, as duas proposições forem falsas. Confira a seguir:

A	B	A ∨ B
V	V	V
V	F	V
F	V	V
F	F	F

Tabela Verdade para Negação (NOT – NÃO)

O operador lógico de negação tem a função de inverter seja uma entrada ou o resultado de uma operação. Confira a seguir:

A	¬A
V	F
F	V

A figura abaixo mostra a negação do resultado de uma fórmula que possui a conjunção. Os parênteses na fórmula deve ser usados com consciência, pois influenciam o resultado. Na figura, os parênteses indicam que a negação fora deles deve inverter o resultado de tudo que está dentro.

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

Resultados na Tabela Verdade

Construímos uma tabela verdade para testarmos todos os resultados possíveis para uma combinação de entradas em determinada fórmula. Considerando seus conhecimentos em tabela verdade, nesta aula estudaremos a implicação lógica.

Implicação Lógica

Uma fórmula é composta por proposições e operadores lógicos, por exemplo a negação (NOT), a conjunção (AND) e a disjunção (OR). Além desses conectores, as proposições podem ser combinadas na forma “se proposição 1, então proposição 2”. O conectivo lógico dessa combinação é o condicional, representado por →, e significa que a verdade da proposição 1 implica a verdade da proposição 2 (Gersting, 2017). Em outras palavras, podemos dizer que, dada uma sequência de proposições, a partir da operação condicional é possível chegar a uma conclusão (um resultado), que é uma nova proposição.

Observe que a primeira parte, antes do conector (A), é chamada de antecedente, e a segunda parte (B) é chamada de consequente:

$A \rightarrow B$

A tabela verdade para o condiciona está apresentada na figura abaixo:

A	B	A → B
V	V	V
V	F	F
F	V	V
F	F	V

Construímos uma tabela verdade para testarmos todos os resultados possíveis para uma combinação de entradas em determinada fórmula. Considerando seus conhecimentos em tabela verdade, nesta aula estudaremos a implicação lógica.

Resultados da Implicação Lógica

Os resultados da implicação não são tão óbvios. Para entendermos, vamos utilizar o exemplo usado em uma nota de aula do professor Chibeni, da Unicamp (Chibeni, 2019). Considere as seguintes proposições:

- A: Soltar a pedra.
- B: A queda da pedra.

A fórmula A → B deve ser lida como “se a pedra for solta, então a pedra cairá”.

Agora, vamos avaliar todas as respostas possíveis com base na tabela verdade:

A	B	A → B
V	V	V
V	F	F
F	V	V
F	F	V

- **Linha 1:** Na primeira linha, temos como entrada a verdade das proposições A e B. Traduzindo para nosso exemplo, quer dizer que a pedra foi solta e caiu, portanto a condição era verdadeira e o resultado é V.
- **Linha 2:** Na linha dois, temos com entrada a verdade para A e a falsidade para B. No nosso exemplo, quer dizer que a pedra foi solta, mas não caiu. Nesse caso, a condição não é verdadeira e o resultado é F.
- **Linhas 3 e 4:** Nas terceira e quarta linhas, temos como entrada a falsidade para A (que é o antecedente), nesse caso, não há como avaliar a condicional, e o resultado é tomado como verdadeiro.

Tautologia, Contradição e Contingência

A tabela verdade deve ser usada como um método exaustivo de extração de resultados. Quando o resultado de uma fórmula obtém somente V como resposta, a fórmula é denominada tautologia, por outro lado, quando o resultado de uma fórmula obtém somente F como resposta, a fórmula é denominada contradição. Quando uma tabela verdade não é uma tautologia e não é uma contradição, ela é uma contingência.

Para saber se duas fórmulas são equivalentes, é necessário construir a tabela verdade e verificar se a equivalência é uma tautologia. Um importante resultado da equivalência são as leis De Morgan.

Aplicações da Tabela Verdade

Nesta aula, vamos estudar a ordem de precedência dos conectivos lógicos.

Ordem de Precedência dos Conectivos Lógicos

A matemática faz parte do cotidiano desde os primeiros anos escolares. As fórmulas começam simples, com as quatro operações básicas e depois vão ganhando complexidade. Assim como as fórmulas matemáticas, é possível construir expressões lógicas mais complexas a partir da combinação das proposições, dos conectivos e dos parênteses. As expressões lógicas podem ser simples, como $A \rightarrow B$, ou podemo combinar diversos operadores lógicos, em expressões mais complexas, por exemplo $(A \wedge B) \vee (B \rightarrow C)$.

Para resolver uma expressão lógica que combina várias preposições com conectores lógicos é preciso obedecer à seguinte regra de precedência:

1. Para expressões que possuem parênteses, primeiro são efetuadas as operações lógicas dentro dos parênteses mais internos.
2. $\neg$ (Negação).
3. $\wedge$, $\vee$ (Conjunção e disjunção).
4. $\rightarrow$ (Implicação).
5. $\leftrightarrow$ (Bicondicional).

Ao seguir rigorosamente a ordem de precedência dos operadores, o uso de parênteses pode ser omitido nos casos adequados. Por exemplo, a fórmula $A \vee (\neg B)$ pode simplesmente ser escrita como $A \vee \neg B$, pois, por causa da ordem de precedência, a negação será realizada primeiro, mesmo sem parênteses.

Dada uma fórmula com várias proposições, conectores e parênteses dentro de parênteses, a resolução deve começar pelos parênteses mais internos. Por exemplo, a fórmula $((A \vee B) \rightarrow C) \wedge A$ deve ter a seguinte ordem de resolução:

1. $A \vee B$ (Parênteses mais internos).
2. $((A \vee B) \rightarrow C)$ (Parênteses mais externos.
3. $((A \vee B) \rightarrow C) \wedge A$ (Operação fora dos parênteses).

A figura a seguir mostra o resultado para essa fórmula:

A	B	A → B	A ∨ B	P → C	Q ∧ A
V	V	V	V	V	V
V	V	F	V	F	F
V	F	V	V	V	V
V	F	F	V	F	F
F	V	V	V	V	F
F	V	F	V	F	F
F	F	V	F	V	F
F	F	F	F	F	F

Veja que foram usadas proposições intermediárias para nomear os resultados. Primeiro, obtém-se P que é o resultado 1, depois utiliza-o para obter Q, que é o resultado 2 e, por fim, o Q é usado para obter o resultado final da fórmula.

Operações e Regras Lógicas na Computação

Uma dúvida que pode surgir é como aplicar todas essas operações e regras lógicas no universo da programação. A resposta é simples: Elas são utilizadas para construir uma sequência de instruções, chamada de algoritmo, que soluciona algum problema. Mais precisamente, as operações lógicas são usadas em estruturas condicionais (ou estruturas de decisão) e têm o objetivo de realizar testes alterando o fluxo de execução de um programa, de acordo com a resposta obtida.

Por exemplo, em um site de aluguel de imóveis, quando o cliente seleciona opções como: Imóvel do tipo apartamento com 1 dormitório, 1 banheiro, sem vaga de garagem, essa seleção é transformada em uma expressão lógica do tipo: Apartamento E 1 quarto E 1 banheiro E sem garagem.

Na programação, a implicação $A \rightarrow B$ é traduzida para se A… então B… e significa que, se A for uma proposição verdadeira, então B acontecerá.

A estrutura condicional faz parte do arsenal de técnicas de programação que permitem alterar o fluxo de execução do programa em detrimento de decisões que são tomadas.