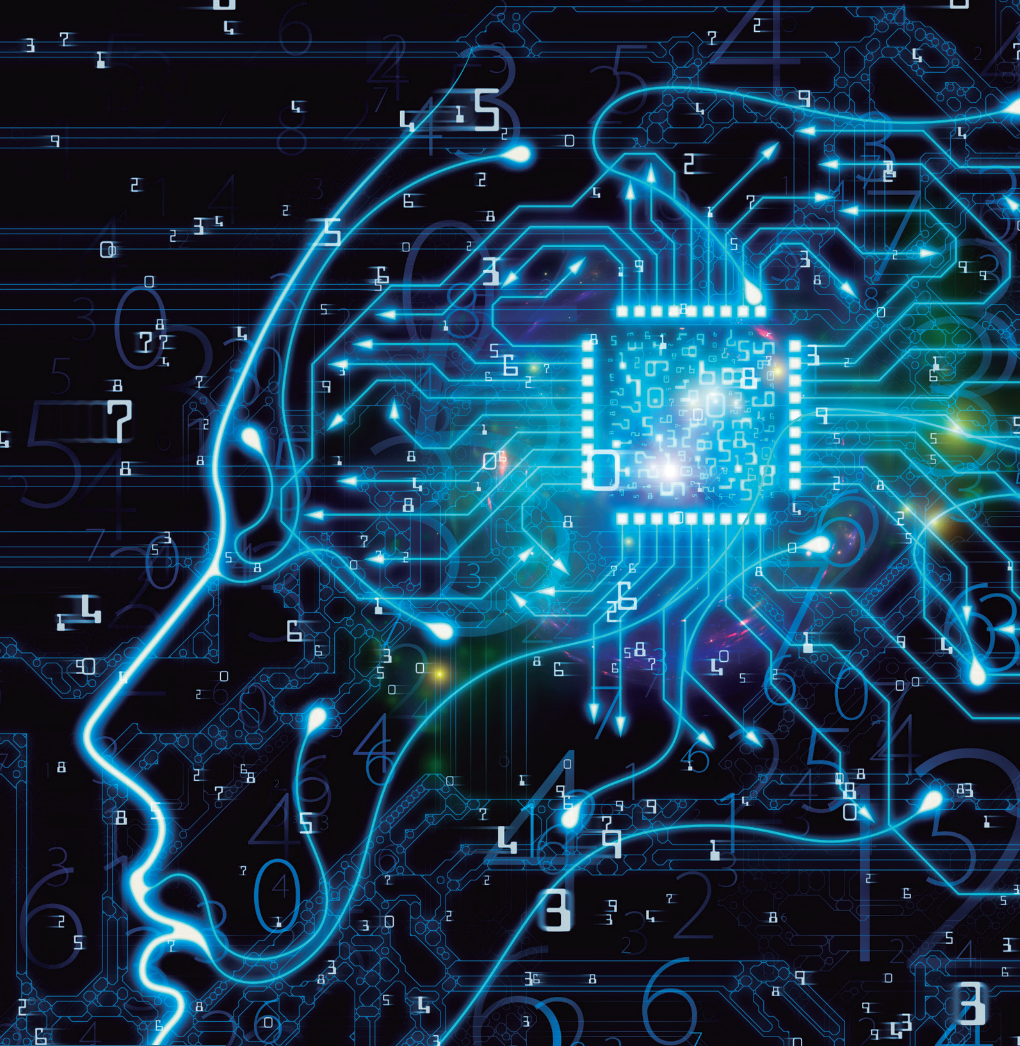


KLS

Lógica Computacional

Lógica Computacional

Thiago Pinheiro Felix da Silva e Lima

Presidência
Rodrigo Galindo

Vice-Presidência de Produto, Gestão e Expansão
Julia Gonçalves

Vice-Presidência Acadêmica
Marcos Lemos

Diretoria de Produção e Responsabilidade Social
Camilla Veiga

Gerência Editorial
Fernanda Migliorança

Editoração Gráfica e Eletrônica
Renata Galdino
Luana Mercurio

Supervisão da Disciplina
Marcio Aparecido Artero

Revisão Técnica
Marcilyanne Moreira Gois
Marcio Aparecido Artero

Imagens
Adaptadas de Shutterstock.
Todos os esforços foram empregados para localizar os detentores dos direitos autorais das imagens reproduzidas neste livro; qualquer eventual omissão será corrigida em futuras edições.

Conteúdo em websites
Os endereços de websites listados neste livro podem ser alterados ou desativados a qualquer momento pelos seus mantenedores. Sendo assim, a Editora não se responsabiliza pelo conteúdo de terceiros.

Dados Internacionais de Catalogação na Publicação (CIP)

Scheffer, Vanessa Cadan
S316l Lógica computacional / Vanessa Cadan Scheffer, Gilberto Vieira, Thiago Pinheiro Félix da Silva e Lima. - Londrina : Editora e Distribuidora Educacional S.A., 2020.
184 p.

ISBN 978-85-522-1688-9

1. Lógica. 2. Proposições. 3. Tabela-Verdade. I. Vieira, Gilberto. II. Lima, Thiago Pinheiro Félix da Silva. III. Título.

CDD 511.3

Jorge Eduardo de Almeida CRB-8/8753

2020
Editora e Distribuidora Educacional S.A.
Avenida Paris, 675 – Parque Residencial João Piza
CEP: 86041-100 — Londrina — PR
e-mail: editora.educacional@kroton.com.br
Homepage: http://www.kroton.com.br/

Sumário

Palavras do autor

Prezado aluno, seja bem-vindo! Vamos dar início à disciplina de Lógica Computacional, que trará conteúdos fundamentais para a construção de algoritmos e a compreensão de linguagens de programação, premissas importantes para disciplinas ligadas à computação e ao desenvolvimento de sistemas.

Na primeira unidade você terá a oportunidade de conhecer e explorar os princípios da lógica, estudar sua evolução e tomar contato com os princípios matemáticos que a regem. Na segunda unidade, você aprenderá a teoria de conjuntos, suas operações, os conceitos em diversos casos, além de conhecer aplicações. Na terceira unidade do material, você terá contato com práticas e técnicas de lógica proposicional, incluindo o uso de conectivos, métodos e inferências lógicas. Finalmente, na quarta e última unidade, você conhecerá a tabela verdade e aplicará seus conceitos em problemas relacionados à lógica computacional; vai aprender como construir a tabela verdade, analisar seus resultados e suas aplicações.

Caro aluno, a lógica computacional contém assuntos multidisciplinares e contribuirá para formação do seu senso crítico, pensamento lógico e racional. Espero que você se sinta motivado a dedicar seu tempo e seus esforços a um estudo que lhe proporcionará chances reais de assimilar os conceitos e as práticas que você utilizará na sua vida profissional.

Vamos lá!

Unidade 1

Thiago Pinheiro Felix da Silva e Lima

Princípios fundamentais da matemática e da lógica

Convite ao estudo

Olá, prezado aluno, seja bem-vindo! Nesta unidade você terá contato com os princípios fundamentais da matemática e da lógica. A partir desses conhecimentos, você estará pronto para se aprofundar no estudo da lógica computacional, que vai ajudá-lo a entender como trabalhar com programação e a construir algoritmos.

Hoje em dia, com sistemas informacionais espalhados por todos os setores da economia e em nosso dia a dia, as escolhas profissionais mais populares quanto à profissão a seguir são, com certeza, voltadas à computação, programação ou análise de sistemas. O surgimento das *startups* é uma tendência, e a demanda por profissionais qualificados em programação tende a subir. Frente a isso, para que o profissional tenha condições de progredir na profissão que demanda construção de sistemas que envolvem software, será necessário um conhecimento sólido de lógica computacional.

Para resolver esses desafios, nesta unidade você aprenderá o que são argumentos, inferências, silogismos e falácias, aprenderá a diferenciar o raciocínio indutivo do raciocínio dedutivo e poderá comunicar-se de maneira eficiente. Aprenderá também a classificar a lógica empírica e pura, diferenciando-as conforme o estudo de lógica transcendental. Você também compreenderá como se deu a evolução da lógica até os dias atuais explorando problemas ligados à área computacional.

Na primeira seção apresentaremos os fundamentos de lógica, conhecimento essencial para se compreender quais são as origens da lógica e conseguir classificá-la. Na segunda seção desta unidade, você será apresentado à evolução histórica da lógica, começando pelos conhecimentos de Aristóteles até à lógica utilizada nos sistemas computacionais atuais. Na terceira e última seção desta unidade, você vai explorar alguns princípios da matemática relacionados à lógica computacional, conhecer seus fundamentos e conceitos e ter contato direto com sua aplicação, abordando especialmente alguns tópicos de análise combinatória e teoria das probabilidades.

Aproveite bem a oportunidade de praticar o estudo dos fundamentos da lógica; tais conceitos são de suma importância para que o profissional que desenvolve sistemas de software utilizando diferentes linguagens de programação e algoritmos execute, de forma eficaz, suas funções técnicas e tenha resultados profissionais excepcionais.

Bons estudos!

Fundamentos de lógica

Diálogo aberto

Caro aluno, nesta seção será apresentada a definição de lógica e você aprenderá a classificá-la como indutiva e dedutiva a fim de construir argumentos válidos e tirar conclusões lógicas a partir de afirmações. Este estudo é importante para o trabalho na medida em que, ao conversar com pessoas, por escrito ou por meio da fala, não haja falha de comunicação ou má interpretação. É fundamental considerar também que tais conceitos permitem a construção de algoritmos para a resolução de problemas do cotidiano e, posteriormente, o desenvolvimento de programas de computador (softwares) de forma a minimizar a existência de *bugs*, falhas e demais comportamentos não esperados.

Para motivar sua aprendizagem, imagine que você trabalha em uma empresa de tecnologia que ajudou a fundar, uma *startup* que desenvolve aplicativos para o setor industrial. Para negociar com seu cliente, papel que é de sua responsabilidade na empresa, você tem que usar o raciocínio lógico para interpretar as intenções das pessoas que se relacionam comercialmente com você e conseguir demonstrar seu produto e fechar vendas.

Seu desafio é interpretar se um negócio pode ou não ser fechado a partir de frases contidas nos e-mails de seus clientes. Você deverá utilizar as classificações de lógica indutiva e dedutiva para esse exercício e pensar sobre possíveis diálogos de negociação. O ponto principal a ser entendido é: a partir de uma determinada frase do cliente, como podemos inferir algumas outras conclusões com grande probabilidade de acerto, ou seja, visando à venda de um produto?

Por exemplo, a partir de um e-mail do cliente, você extraiu as seguintes frases:

"Realizamos dezenas de testes com o seu software e em todos ele foi capaz de chegar à melhor solução para nosso problema."

"Nos últimos anos, os softwares que resolveram nossos problemas foram adquiridos para a melhoria de nossos processos."

Você deverá concluir, com base na lógica clássica, se o cliente está motivado a adquirir o produto compondo argumentos e esclarecendo se a lógica utilizada é a indutiva ou a dedutiva.

Os conteúdos teóricos necessários para vencer o desafio estão apresentados nesta seção, na qual serão caracterizadas as definições de lógica, a lógica transcendental e formal, classificação da lógica: indutiva, dedutiva, clássica, não clássica.

Sua aprendizagem será importante para um futuro profissional de sistemas computacionais, em especial os relacionados à análise de sistemas e linguagem de programação. Então, bom trabalho!

Não pode faltar

Prezado aluno, começaremos nossa exploração sobre lógica com algumas discussões filosóficas e históricas acerca do tema. Apresentaremos brevemente os conceitos de lógica formal de Aristóteles, lógica transcendental e como classificá-las em indutiva, dedutiva, clássica e não clássica para que consigamos diferenciar facilmente esses conceitos e aprender todas as nuances do estudo da lógica.

Segundo Forbellone (2005, p. 1), lógica é a "arte de bem pensar", que é a "ciência das formas do pensamento". Visto que a forma mais complexa do pensamento é o raciocínio, a lógica estuda a correção do raciocínio; podemos então enxergá-la simplesmente como uma maneira dos seres humanos raciocinarem ou mesmo como um encadeamento de fatos ou uma organização de técnica ou ciência pura, como, por exemplo, a matemática.

A lógica vem sendo matéria estudada desde os filósofos da Grécia Antiga, muitos anos antes de Cristo, começando em Aristóteles e sua exploração da lógica formal, e vem evoluindo até os tempos atuais, tornando-se assunto importante na Matemática, conforme discute Fajardo (2017). O estudo da lógica nos permite, portanto, de forma prática, entender como nosso raciocínio lógico é formado, fundamentar nossos argumentos, escrever e registrar de forma organizada, nos comunicar melhor, além de fazer conexões entre diversos assuntos e entender melhor o mundo que está a nossa volta. Atualmente, é fortemente estudada em matérias relacionadas à ciência da computação, tecnologia da informação e programação, pois é a base para a construção de algoritmos. É importante ter um forte entendimento dessa ciência para que possa compreender como construir algoritmos e desenvolver e analisar sistemas computacionais.

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

- 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 formal e a lógica transcendental.

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

A lógica formal lida com as relações entre premissas e as conclusões que se chegam a partir das premissas, independentemente se a premissa é verdadeira ou falsa (MUNDIM, 2002).

A lógica busca uma harmonia de raciocínio utilizando-se de argumentos para se desenvolver um raciocínio, bem como traz regras a fim de que um raciocínio encadeado corretamente possibilite conclusões verdadeiras, conforme expôs Fajardo (2017).

Para se entender a lógica formal 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. A lógica não permite concluir se uma proposição ou afirmação é 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.

Exemplificando

Se eu afirmar (premissas):

Em dias sem nuvens, chove.
Se há chuva, João sai com seu guarda-chuva.

Então (conclusão):

Em um dia sem nuvens, João sairá com seu guarda-chuva.

A primeira premissa parece absurda ou falsa, mas caso você considere ambas verdadeiras, terá um argumento válido: a conclusão é inevitável.

Por isso, a análise da veracidade das premissas e a análise da validade dos argumentos são distintas. A lógica se ocupa da validade do raciocínio, a verdade das premissas deve ser decidida por outros meios.

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**. A seguir, mostraremos alguns exemplos de inferências da lógica formal com argumento válido e conclusão lógica. Veja que, a partir de duas frases, que são as premissas, chegamos a uma conclusão.

1. Todos os homens são mortais.

   Elias é homem.

   Logo, Elias é mortal.

2. Toda novela conta histórias sobre o dia a dia das pessoas.

   Roque Santeiro é uma novela.

   Logo, Roque Santeiro conta histórias sobre a vida das pessoas.

Em especial no século XIX, como apresenta Fajardo (2017), alguns matemáticos e filósofos concluíram que a lógica formal não era suficiente para que se pudesse alcançar o rigor necessário na análise dos argumentos. Em determinados idiomas, como na língua portuguesa, por exemplo, a linguagem falada e escrita apresenta um conjunto de sinais (sonoros, no caso da fala, e visuais, no caso da escrita). Para a linguagem escrita, existe uma série de símbolos, incluindo letras, acentos gráficos e sinais de pontuação, que, quando reunidos, formam palavras, sentenças e frases. Para a correta comunicação, é necessário o domínio da língua e de suas regras gramaticais, caso contrário, podemos recair em frases ambíguas que permitem múltiplos entendimentos, conforme discussão em Fajardo (2017).

Um usuário proficiente da língua também pode manipulá-la para construir os chamados paradoxos, como, por exemplo, os paradoxos de Zenão de Eléia (490-430 a.C.), que desenvolveram uma discussão sobre a existência ou não do movimento (um fato empírico) a partir de conclusões lógicas de premissas difíceis de serem contestadas. A seguir, serão expostos alguns exemplos de situações exploradas por Zenão que trazem conclusões lógicas, embora cientificamente inverídicas.

- A flecha que voa jamais sai do lugar, porque, em cada instante de tempo, ocupa uma só posição no espaço, logo, ela está imóvel todo o tempo.
- Entre dois pontos existem infinitos pontos. Ninguém pode atravessar infinitos pontos. Então, não há movimento.

Quando a partir de argumentos válidos e verdadeiros obtemos conclusões falsas, temos erros de raciocínio. Os paradoxos apresentados mostram que conceituar o infinito com a linguagem tradicional escrita é muito complicado quando lidamos com conceitos abstratos.

Um famoso silogismo presente nessa interpretação de Zenão de Eléia apresenta que o herói Aquiles nunca seria capaz de alcançar uma tartaruga, pois quando Aquiles alcançasse a posição da tartaruga, ela já teria avançado mais um pouco.

É 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.

Vamos estudar mais à frente que a lógica matemática originou a lógica utilizada em computadores, que são sistemas digitais construídos para executar tarefas programáveis e que fornecem respostas e saídas exatas, conforme determinado conjunto de instruções e dados fornecidos a esses sistemas.

A **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 Conhecimento Empírico e Conhecimento Puro.

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

Já o **Conhecimento Puro**, também chamado de conhecimento a priori, é relativo à representação que não se mescla com a sensação, é puramente racional, não depende de nenhuma informação vinda de nossos sentidos.

Como vimos anteriormente ao discutirmos as inferências, a lógica pode ser classificada em indutiva ou dedutiva.

A **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.

Exemplos de Lógica Dedutiva:

1. Todos os brasileiros gostam de praia. Antônio é brasileiro. Portanto, Antônio gosta de praia.
2. Todo os jogadores de futebol treinam em academias. Paulo é um jogador de futebol. Então, Paulo treina em uma academia.

Já a **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.

Um exemplo famoso é que por séculos a afirmação “todo cisne é branco” foi considerada verdadeira pelos europeus, pois a cada encontro com um cisne, eles verificavam que sua cor era branca. Pela lógica indutiva, chegou-se à conclusão que todos eram brancos. Com as navegações, entretanto, verificou-se que na Austrália há cisnes negros. O contraexemplo, portanto, invalidou imediatamente a conclusão anterior.

Observando alguns exemplos, fica mais fácil perceber as diferenças entre as lógicas indutiva e dedutiva. Veja a seguir exemplos de lógica indutiva.

Exemplos de Lógica Indutiva:

1. Uma maçã solta no ar cai em direção ao solo. Uma caneta solta no ar cai em direção ao solo. Um livro solto no ar cai em direção ao solo. Todos os objetos soltos no ar caem em direção ao solo.
2. Um imã atrai um prego de ferro. Um imã atrai limalha de ferro. Um imã atrai argolas de ferro. Um imã atrai o elemento ferro.

Note a importância do raciocínio indutivo para o desenvolvimento da ciência por meio da experimentação e da observação da natureza, dos fatos naturais isolados, permitindo a construção das leis naturais em um processo de generalização.

Assimile

Para a lógica dedutiva, partiremos de premissas gerais para concluirmos verdades específicas e particulares. Por outro lado, para a lógica indutiva, partiremos da experiência com as verdades e fatos particulares na busca de uma conclusão geral.

A lógica também pode ser dividida em **lógica clássica** e **lógica não clássica**. Estudamos até esse ponto a lógica clássica cujo expoente foi Aristóteles e que está centrada na verdade das conclusões e na validade dos argumentos. A conclusão pode ter dois valores lógicos: verdadeiro (1) ou falso (0).

A lógica não clássica permite variações, como nos casos em que mais de dois valores de verdade podem ser aplicados, por exemplo. Um exemplo é a conhecida lógica fuzzy, para a qual o valor verdade pode ser qualquer número real entre 0 e 1. Outras variações são possíveis com o abandono de alguns princípios da lógica clássica e análise de suas consequências.

Para exemplificar, podemos citar a lógica modal, desenvolvida por Lewis no início do século passado, na qual a proposição pode ser, além de verdadeira ou falsa, necessária ou impossível (necessariamente verdadeira ou necessariamente falsa).

Conforme explica Oliveira (2010), as lógicas não clássicas seguem geralmente uma ou mais das três características a seguir:

- São baseadas em linguagens mais ricas em poder de expressão.
  - Por exemplo: as lógicas modais, que apresentam operadores de necessidade e possibilidade.
- São baseadas em princípios distintos.
  - Por exemplo: as lógicas não reflexivas, que negam axiomas da lógica clássica, que não admitem o princípio da identidade.
- Admitem semânticas distintas.
  - Por exemplo: as lógicas do tempo, que apresentam operadores temporais específicos para esse tipo de lógica.

Caro aluno, chegamos ao fim desta seção e esperamos que tudo que lhe apresentado nas páginas anteriores o tenha inspirado a refletir sobre como a lógica faz parte de nossa vida e pode nos ajudar nas decisões cotidianas, em um mundo a cada dia mais informatizado e conectado.

Sem medo de errar

Você se lembra que na situação problema você trabalha em uma *startup* que produz softwares para uso industrial e está analisando a troca de e-mails com o cliente para concluir se a venda deverá ser fechada ou não? Para isso, você decidiu recorrer à lógica clássica e aos raciocínios dedutivo e indutivo.

Vamos retomar as frases extraídas do e-mail do cliente.

"Realizamos dezenas de testes com seu software e em todos ele foi capaz de chegar à melhor solução para nosso problema."

"Nos últimos anos, os softwares que resolveram nossos problemas foram adquiridos para a melhoria de nossos processos."

Por meio de um raciocínio indutivo, partindo de diversos casos particulares para chegarmos a uma conclusão geral, poderíamos realizar a seguinte inferência com a primeira frase:

Realizamos dezenas de testes com seu software e em todos ele foi capaz de chegar à melhor solução para nosso problema.

Portanto, o software da sua empresa resolve o problema do cliente.

As dezenas de testes individuais em diferentes contextos nos permitem concluir que o software atendeu às expectativas do cliente e resolve o problema da indústria.

Com relação à segunda frase, podemos realizar a seguinte inferência:

"Nos últimos anos, os softwares que resolveram nossos problemas foram adquiridos para a melhoria de nossos processos."

Portanto, o cliente sempre adquire os softwares que resolvem seus problemas.

Em todas as situações anteriores, a empresa acatou à solução trazida pelo fornecedor para a solução dos seus problemas.

As duas conclusões apresentadas poderiam, por sua vez, ser combinadas por meio de um raciocínio dedutivo para se chegar a uma conclusão:

O cliente sempre adquire os softwares que resolvem seus problemas.
O software da sua empresa resolve o problema do cliente.
A conclusão natural da dedução, seria:
Portanto, o software de sua empresa será adquirido.

Isso mostra que as perspectivas são positivas para o fechamento do negócio, mas não se esqueça de que as conclusões do raciocínio indutivo são válidas até que ocorra um contraexemplo. Então, antes de comemorar, é melhor esperar que o cliente confirme formalmente a aquisição do software após as negociações finais.

Faça valer a pena

1. Segundo Forbellone (2005, p. 1), “lógica é a “arte de bem pensar”, que é a “ciência das formas do pensamento”. Visto que a forma mais complexa do pensamento é o raciocínio, a lógica estuda a “correção do raciocínio”. Tomando por base duas premissas, podemos chegar a uma conclusão, em um processo que é chamado inferência.

Analise a seguinte inferência:

I. Todos os softwares são desenvolvidos por programadores.

II. Microsoft Excel é um software.

III. Microsoft Excel foi desenvolvido por programadores.

O raciocínio lógico utilizado na inferência apresentada é chamado:

a. Indução.

b. Falácia.

c. Silogismo.

d. Dedução.

e. Método científico.

2. João é engenheiro e vai ao parque no domingo de manhã.

Elena é engenheira e vai ao parque no domingo de manhã.

Rodrigo é engenheiro e vai ao parque no domingo de manhã.

Camila é engenheira e vai ao parque no domingo de manhã.

(...)

Imagine que o raciocínio exposto para as quatro pessoas acima tem se confirmado em diversas consultas, de modo que você decide utilizar o raciocínio indutivo. Você conclui que:

a. Todos os engenheiros têm bons hábitos de saúde.

b. Todas as pessoas têm formação em engenharia.

c. O parque é agradável.

d. Todas as pessoas vão ao parque no domingo de manhã.

e. Todos os engenheiros vão ao parque no domingo de manhã.

3. Todos os alunos dos cursos de Análise e Desenvolvimento de Sistemas fazem uma disciplina de lógica computacional.

_________________.

Logo, José faz uma disciplina de lógica computacional.

A premissa que preenche corretamente a lacuna acima é:

a. José é aluno de um curso de Análise e Desenvolvimento de Sistemas.

b. José deseja trabalhar com Análise e Desenvolvimento de Sistemas.

c. No curso de José, não há disciplina de lógica computacional.

d. José gosta de lógica computacional.

e. José faz uma disciplina de lógica computacional, portanto, ele cursa Análise e Desenvolvimento de Sistemas.

Seção 2

Evolução da lógica

Diálogo aberto

Caro aluno, nesta seção você vai estudar como se deu a evolução da lógica, desde Aristóteles até os dias atuais. Parafraseando Isaac Newton, que disse: “se eu vi mais longe, foi porque me apoiei em ombros de gigantes”, é importante compreendermos os aspectos históricos que motivaram o estudo da lógica. Os gigantes, nesse caso, são matemáticos, filósofos e pensadores como Aristóteles, George Boole e Gottlob Frege, por exemplo. Suas ideias serviram de base para todas as lógicas de programação com as quais lidamos atualmente.

Lembre-se de que você está trabalhando como um colaborador de uma *startup* de tecnologia. Seu segundo desafio será o de mostrar aos colaboradores do setor de desenvolvimento de softwares como alguns conceitos de lógica podem ser relacionados à área da computação. Mais especificamente, você deverá elaborar uma apresentação destacando como conceitos como aberto e fechado, ou ligado e desligado podem ser representados e trabalhados com uma álgebra booleana. Para tanto, foi proposto a você o seguinte problema:

O comitê diretor de uma multinacional é formado por três membros: o diretor executivo, o vice-diretor financeiro e o vice-diretor de relações institucionais. Um projeto de criação de uma filial dessa empresa em um país emergente será votado pelo comitê, e o projeto só passará se o diretor executivo votar a favor e obtiver maioria. Você deverá projetar um circuito de modo que cada membro vote a favor apertando um botão e, ao final do processo, uma luz se acenderá caso o projeto seja aprovado. Esse circuito poderá ser utilizado em outras votações pelo mesmo comitê. Como você projetará esse circuito? Que relação esse projeto guarda com aspectos da evolução da lógica?

O estudo dos grandes períodos de evolução da lógica, em especial o Período Booleano, dará a você pistas para resolver esse problema. Bons estudos e mãos à obra!

Não pode faltar

Em um sentido amplo, a lógica é o estudo da estrutura e dos princípios relativos ao raciocínio, à estruturação do pensamento, com ênfase na argumentação, que pode ser considerada como válida ou inválida. Com base em premissas, ela permite a construção do raciocínio indutivo ou dedutivo, e também a realização de operações lógicas simbólicas e demonstrações matemáticas. Podemos classificar a estudo da lógica em três grandes períodos: o Período Aristotélico, o Período Booleano e o Período Atual.

Segundo Abar (2004), a história da lógica tem início com o filósofo grego Aristóteles (384 – 322 a.C.) de Estagira (hoje Estavo), na Macedônia, que desenvolveu a teoria do silogismo, um tipo de inferência dedutiva. Pode-se dizer que o Período Aristotélico se inicia aproximadamente em 390 a.C., perdurando até meados do século XIX (1840 d.C.). Aristóteles é tido como o homem mais erudito de todos os tempos, sendo sua morte considerada como o marco do fim do primeiro grande período, a Idade Helênica, na história da civilização grega (BOYER, 1996).

Assimile

Silogismo nada mais é do que um argumento constituído de proposições das quais se infere (extrai) uma conclusão. Assim, não se trata de conferir valor de verdade ou falsidade às proposições (frases ou premissas dadas) nem à conclusão, mas apenas de observar a forma como foi constituído. É um raciocínio mediado que fornece o conhecimento de uma coisa a partir de outras coisas (buscando, pois, sua causa) (CABRAL, 2020).

Em um silogismo, as premissas e conclusões se encaixam de tal forma que, uma vez que você aceita as premissas como verdadeiras, fica obrigado a aceitar que a conclusão também o é, independentemente do teor do real argumento que está sendo construído (ZEGARELLI, 2013).

Considere, por exemplo, o seguinte argumento lógico dedutivo:

Premissas:

- Todos os brasileiros torcem pelo Brasil.
- José é brasileiro.

Conclusão:

- José torce pelo Brasil.

Assumindo as premissas como **verdadeiras**, concluímos que José torce pelo Brasil. Observe que ao analisarmos as premissas, não podemos fazer juízo de valor sobre elas. O argumento lógico é deduzido a partir daquilo que é colocado como **verdade**, e a nossa opinião sobre a validade das premissas não pode interferir na elaboração da conclusão.

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 daquilo que está sendo discutido (ZEGARELLI, 2013).

O 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.

O princípio da não contradição busca a especificidade 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.

O princípio do terceiro excluído afirma que toda proposição é verdadeira ou falsa, não havendo uma terceira possibilidade para valoração da proposição.

Exemplificando

Compreender os três princípios que regem a Lógica Clássica é importante para chegarmos a conclusões lógicas sobre proposições. Considere as seguintes proposições:

I. A Estátua da Liberdade é a Estátua da Liberdade.

Na proposição (I) verificamos o princípio da identidade, ou seja, cada coisa individual é idêntica a si mesma. Mesmo que não conheçamos a Estátua da Liberdade, podemos estabelecer com certeza e por pura lógica que essa proposição é verdadeira.

II. O número dois é par.
III. O número dois é ímpar.

As proposições (II) e (III) exemplificam aquilo que chamamos de princípio da não contradição, ou seja, o número dois não pode ser, ao mesmo tempo, par e ímpar. Logo, se a proposição (II) é verdadeira, a proposição (III) necessariamente é falsa.

IV. Meu nome é Carlos.

Na proposição (IV) podemos observar o princípio do terceiro excluído. Você pode se chamar Carlos (verdade) ou não (falsidade). Não há uma terceira possibilidade para essa 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), como os paradoxos de Zenão de Eléia.

Exemplificando

Um exemplo interessante sobre paradoxos é o paradoxo do mentiroso, atribuído a Eubúlides de Mileto, filósofo do século IV a.C., em que um homem diz: “Estou mentindo”. O que este homem diz será verdadeiro ou falso? Se for verdadeiro, então ele está realmente a mentir e, por isso, o que diz é falso. Se for falso, isso concorda com o que ele afirma, parecendo então que o que diz é verdadeiro (SANTOS, 2014).

Solucionar esses tipos de paradoxo não é tarefa fácil! Filósofos, matemáticos e estudiosos da lógica tem se debruçado sobre eles ao longo de muitos anos.

Reflita

Uma variação mais elaborada do paradoxo do mentiroso pode ser observada na construção apresentada na Figura 1.1:

Figura 1.1 | Paradoxo do mentiroso

A afirmação abaixo é verdadeira.

A afirmação acima é falsa.

Fonte: elaborada pelo autor.

O que você pode concluir com relação às afirmações da figura?

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 (1815-1864), Georg Cantor (1845-1918) e Gottlob Frege (1848-1925).

George Boole foi o inventor da chamada Álgebra Booleana, que foi o primeiro sistema totalmente detalhado que lida com a lógica como cálculo. Colocando 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. Considere, por exemplo, as seguintes afirmações:

(A) O Brasil é um país da América do Sul.

(B) Pablo Picasso é um grande jogador de futebol.

Assumindo a primeira proposição como verdadeira e a segunda como falsa, podemos dizer que:

(A) = 1

(B) = 0

Na Álgebra Booleana a adição é associada ao conectivo *ou*. Assim, a proposição "O Brasil é um país da América do Sul *ou* Pablo Picasso é um grande jogador de futebol" pode ser representada como:

(A) + (B) = 1 + 0 = 1. (Verdadeira).

E na Álgebra Booleana a multiplicação é associada ao conectivo *e*. Assim, a proposição "O Brasil é um país da América do Sul *e* Pablo Picasso é um grande jogador de futebol" pode ser representada como:

(A) $\times$ (B) = 1 $\times$ 0 = 0 (Falsa).

Embora o cálculo dos valores seja parecido ao da aritmética convencional, o significado dessas operações é puramente lógico.

Georg Cantor foi o idealizador da Teoria de Conjuntos. A Álgebra dos Conjuntos, advinda da Teoria de Conjuntos, com operações particulares como União $(\cup)$ e Intersecção $(\cap)$ serviu não apenas como uma estrutura de linguagem para a lógica formal, mas também como alicerce de toda a Matemática Moderna.

Gottlob Frege foi o 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). Segundo Alcoforado (2009), o ponto de partida dos trabalhos de Frege consistiu em construir um sistema formal cujas noções básicas fossem fixadas com exatidão e clareza, e a seguir fossem estabelecidos os enunciados primitivos e regras de inferências que tornassem possível desenvolver sem qualquer lacuna uma demonstração nesse sistema. Ele foi assim levado a desenvolver, pela primeira vez, um sistema formal a partir do qual é possível entender com exatidão não só o que vem a ser uma prova como também obter provas pela exclusiva utilização de regras formais aplicadas aos axiomas. Podemos afirmar, portanto, que toda a notação (símbolos) utilizada pela lógica formal (lógica simbólica) nos dias de hoje teve origem com as notações introduzidas por Frege.

Segundo Alencar Filho (2002), quando pensamos, efetuamos muitas vezes certas operações sobre proposições, chamadas operações lógicas. Estas operações obedecem a 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:

Quadro 1.1 | Operadores lógicos

$\sim$	O til corresponde à operação lógica NEGAÇÃO. Alguns autores também utilizam o símbolo $\leftarrow$ para designar negação.
$\wedge$	A cunha corresponde à operação lógica CONJUNÇÃO. Em programação, a conjunção é representada pela palavra AND, ou pelo símbolo &, que corresponde ao conectivo e.
$\vee$	A letra v corresponde à operação lógica DISJUNÇÃO. Equivale à palavra ou em seu sentido inclusivo. Em programação, a conjunção é também representada pela palavra OR.
$\rightarrow$	A seta corresponde à operação CONDICIONAL. Em português, corresponde à relação "se ..., então ...".
$\leftrightarrow$	A dupla seta corresponde à operação BICONDICIONAL. Em português, corresponde à relação "se, e somente se, ...".

Fonte: adaptado de Alencar Filho (2002).

Exemplificando

Como escreveríamos em linguagem simbólica a seguinte proposição: "João não é gaúcho e Jaime não é paulista?

Considerando as proposições:

p: João é gaúcho e

q: Jaime é paulista.

Podemos utilizar os operadores lógicos para escrever: $\sim p \wedge \sim q$

Negamos p e q, utilizando o conectivo "e" de acordo com os símbolos apresentados no Quadro 1.1.

É no início do século XX – por volta de 1910 – que chegamos ao Período Atual da lógica. Seus maiores expoentes são Bertrand Russel (1872-1970) e Alfred North Whitehead (1861-1947). O Período Atual caracteriza-se pelo desenvolvimento de sistemas formais polivalentes, que trabalham não apenas com os valores lógicos verdadeiro e falso (lógica clássica), mas também com imprecisões e contradições, assumindo como valores lógicos o necessariamente verdadeiro, o necessariamente falso, o indeterminado, o indecidível, dentre outros. Segundo Buchsbaum (2006), tais sistemas são chamados de lógicas não clássicas, dentre as quais podemos destacar as lógicas paracompletas (que não respeitam o princípio do terceiro excluído), as lógicas paraconsistentes (que não respeitam o princípio da não contradição) e as lógicas modais (que estudam possíveis variações da veracidade ou falsidade).

Também destacamos, no conjunto de lógicas não clássicas, a lógica fuzzy, que tem apresentado contribuições para a Informática, no campo da Inteligência Artificial, com os Sistemas Especialistas. De acordo com Martins e Martins (2015), a lógica fuzzy é uma modalidade da lógica capaz de tratar conceitos vagos, imprecisos ou ambíguos – em geral descritos na linguagem natural humana – e convertê-los para um formato numérico, de fácil processamento computacional. No nosso dia a dia utilizamos conceitos subjetivos para classificar determinadas situações. Considere as seguintes afirmações:

- Para chegar ao posto de gasolina, prossiga na rodovia por mais **alguns** metros.
- Para atingir meu peso ideal preciso perder **alguns** quilos.
- Podemos dizer que, devido à atual conjuntura econômica, estamos com uma moeda **estável**.
- A previsão do tempo para amanhã indica que teremos um dia **parcialmente** chuvoso.

Os termos destacados nessas afirmações são conceitos vagos, que envolvem imprecisão. Nesse sentido, esses termos são chamados de fuzzy. O conceito "fuzzy" pode ser entendido como uma situação em que não podemos responder simplesmente "sim" ou "não". Mesmo conhecendo as informações necessárias sobre a situação, dizer algo entre "sim" e "não" como "talvez", "quase", …torna-se mais apropriado. (ABAR, 2004).

Aprendemos nesta seção como se deu a evolução da lógica desde seus primórdios, partindo das ideias de Aristóteles, até os dias atuais. Compreender essa evolução nos ajudará a melhor compreender as operações lógicas fundamentais e relacioná-las com diversas e atuais linguagens de programação.

Sem medo de errar

Caro aluno, agora que já estudamos a evolução da lógica em seus aspectos históricos, vamos retomar a situação-problema apresentada no início desta seção?

Trabalhando como um colaborador de uma *startup* de tecnologia, seu segundo desafio consiste em projetar um circuito de votação de projetos para o comitê diretor de uma empresa multinacional. O comitê diretor dessa empresa multinacional é formado por três membros: o diretor executivo, o vice-diretor financeiro e o vice-diretor de relações institucionais que votarão um projeto de criação de uma filial dessa empresa em um país emergente. O projeto só passará se o diretor executivo votar a favor e obtiver maioria. Você deverá projetar um circuito de modo que cada membro vote a favor apertando um botão e, ao final do processo, uma luz se acenderá caso o projeto seja aprovado. Lembre-se de que você deverá elaborar uma apresentação destacando como conceitos como aberto e fechado, ou ligado e desligado, podem ser representados e trabalhados com uma álgebra booleana.

Inicialmente vamos determinar como se dará o funcionamento do interruptor desse circuito. Um interruptor é um dispositivo ligado a um ponto de um circuito, que pode assumir um dos dois estados, "fechado" ou "aberto". No estado "fechado" (que indicaremos por 1) o interruptor permite que a corrente passe através do ponto, enquanto no estado "aberto" (que indicaremos por 0) nenhuma corrente pode passar pelo ponto (ABAR, 2004). Quando tivermos a passagem de corrente (estado 1), uma luz ligada ao circuito se acenderá. De modo análogo, quando não tivermos a passagem de corrente (estado 0) a luz não se acenderá.

Em nossa apresentação, representaremos o diretor executivo pela letra A, o vice-diretor financeiro pela letra B e o vice-diretor de relações institucionais pela letra C. Podemos então, elaborar uma tabela com a combinação de todos os valores lógicos (1 ou 0) para os votos dos membros do comitê:

Tabela 1.1 | Combinação de valores lógicos

A	B	C	Resultado
1	1	1	1
1	1	0	1
1	0	1	1
1	0	0	0
0	1	1	0
0	1	0	0
0	0	1	0
0	0	0	0

Fonte: elaborado pelo autor.

A tabela nos mostra que das oito combinações possíveis, em apenas três situações a luz se acenderá e o projeto será aprovado. O valor lógico para o circuito A (diretor executivo) obrigatoriamente deverá ser igual a 1. Além disso, para que o projeto seja aprovado, o circuito de um dos dois vice-presidentes (ou ambos) também deverá ser igual a 1. Logo, a luz ligada aos circuitos só se acenderá e o projeto só será aprovado, quando o valor lógico na ultima coluna da tabela for igual a 1.

Assim, projetando o circuito para acendimento da lâmpada conforme a configuração apresentada na tabela, teremos um protótipo que poderá ser utilizado não apenas nessa votação, mas também em outras votações desse comitê. Perceba que para elaboração desse projeto utilizamos uma linguagem simbólica (lógica simbólica) baseada na álgebra booleana, que ganhou notoriedade no Período Booleano estudado nesta seção.

Faça valer a pena

1. Em um sentido amplo, a lógica é o estudo da estrutura e dos princípios relativos à argumentação válida, sobretudo da inferência dedutiva e dos métodos de prova e demonstração. Podemos classificar a estudo da lógica em três grandes períodos: o Período Aristotélico, o Período Booleano e o Período Atual (JAPIASSÚ; MARCONDES, 2006).

Considerando a evolução histórica da lógica, faça a associação dos grandes períodos discriminados na Coluna A, com as características de cada período, apresentadas na Coluna B.

I. Período Aristotélico.	Caracteriza-se pelo desenvolvimento de sistemas formais polivalentes, que trabalham não apenas com os valores lógicos verdadeiro e falso, mas também com imprecisões e contradições, assumindo outros possíveis valores lógicos para as proposições.
II. Período Booleano.	Caracteriza-se pelo desenvolvimento da Lógica Formal, também chamada de Lógica Simbólica, na qual símbolos computáveis substituem palavras e proposições.
III. Período Atual.	Caracteriza-se como uma ciência cuja essência era a teoria do silogismo (certa forma de argumento válido). Corresponde à chamada Lógica Clássica, regida, basicamente, por três princípios: o da identidade, o da não contradição e o do terceiro excluído.

Assinale a alternativa que apresenta a associação correta entre as colunas.

a. I - 1; II - 2; III – 3.

b. I - 1; II - 3; III – 2.

c. I - 2; II - 1; III – 3.

d. I - 3; II - 2; III – 1.

e. I - 3; II - 1; III – 2.

2. Segundo Alencar Filho (2002), quando pensamos, efetuamos muitas vezes certas operações sobre proposições, chamadas operações lógicas. Estas operações obedecem a 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 $\sim, \wedge, \vee, \rightarrow, \leftrightarrow$.

Sejam as proposições *p*: João vai dirigir e *q*: Jaime vai beber. Traduzindo para a linguagem corrente a proposição: $\sim p \rightarrow \sim q$ temos:

a. João vai dirigir se e somente se Jaime vai beber.

b. João não vai dirigir ou Jaime não vai beber.

c. João vai dirigir e Jaime vai beber.

d. Se João vai dirigir, então Jaime não vai beber.

e. Se João não vai dirigir, então Jaime não vai beber.

3. Quando discorremos sobre o Período Aristotélico, estamos nos referindo à chamada Lógica Clássica, que corresponde ao tipo de lógica mais comum. A Lógica Clássica é regida por três princípios.

Considerando os princípios que regem a Lógica Clássica, preencha as lacunas das sentenças a seguir:

O ______ estabelece que todo objeto é idêntico a si mesmo. O ______ busca a especificidade de cada coisa, ou seja, é impossível que ela seja e não seja ao mesmo tempo. O ______ afirma que toda proposição é verdadeira ou falsa, não havendo uma terceira possibilidade para valoração da proposição.

Assinale a alternativa que preenche corretamente as lacunas:

a. princípio da identidade; princípio da não contradição; princípio do terceiro excluído.

b. princípio da identidade; princípio do terceiro excluído; princípio da não contradição.

c. princípio da não contradição; princípio da identidade; princípio do terceiro excluído.

d. princípio da não contradição; princípio do terceiro excluído; princípio da identidade.

e. princípio do terceiro excluído; princípio da identidade; princípio da não contradição.

Seção 3

Princípios matemáticos

Diálogo aberto

Caro aluno, após aprendermos o que é a lógica e estudarmos como se deu sua evolução, nesta terceira seção nos debruçaremos sobre alguns princípios matemáticos relacionados à lógica. Você pode já ter se deparado com problemas do tipo:

- De quantas maneiras podemos escolher um *password* válido para um computador?
- Qual é a probabilidade de eu ganhar um prêmio de loteria?
- Qual é a rota mais curta entre duas cidades, considerando um determinado meio de transporte?
- Como podemos ordenar uma lista de inteiros de modo que os inteiros fiquem em ordem crescente? Em quantos passos podemos fazer essa ordenação?

Esses problemas, além do raciocínio lógico, envolvem noções de listagem, contagem e agrupamentos, noções essas que teremos oportunidade de conhecer um pouco melhor nesta seção e ajudarão você a resolver seu terceiro desafio.

Lembre-se, você está trabalhando como um colaborador de uma *startup* de tecnologia e mais um desafio lhe é apresentado, agora relacionado ao número de dispositivos que têm acesso a uma determinada rede de computadores privada (do tipo intranet). Essa rede de computadores foi configurada de tal modo que, cada um dos 3 escritórios da empresa (A, B, C) tem cinco pontos de acesso à rede. Em um determinado momento havia 13 computadores conectados à rede e uma sobrecarga foi detectada por um software de gerenciamento de fluxo de dados. Essa sobrecarga foi causada em um escritório em que cinco computadores estavam conectados à rede. Deseja-se saber qual é a probabilidade (chance) de que essa sobrecarga tenha partido do escritório C. Você deverá apresentar a solução desse problema de forma detalhada a seu superior imediato. Será que você consegue?

Os conhecimentos discutidos nesta seção ajudarão você a resolver esse desafio. Bons estudos!

Não pode faltar

Caro aluno, na Seção 1 vimos que a lógica é a “arte de bem pensar” (FORBELLONE; EBERSPACHER 2005, p. 1). Na Seção 2 ampliamos nossa compreensão, admitindo que a lógica é o estudo da estrutura e dos princípios relativos à argumentação válida, sobretudo da inferência dedutiva e dos métodos de prova e demonstração (JAPIASSÚ; MARCONDES, 2006). Apesar de boa parte do desenvolvimento da lógica ter se dado a partir do estudo das relações entre premissas e conclusões, não podemos negligenciar a íntima relação existente entre a lógica e a matemática, especialmente um ramo da matemática denominado matemática discreta.

Segundo Picado (2008), a matemática discreta (também conhecida como matemática finita ou matemática combinatória) é um ramo da matemática voltado ao estudo de objetos e estruturas discretas ou finitas (estruturas discretas são estruturas formadas por elementos distintos desconexos entre si).

Genericamente, a matemática discreta é usada quando contamos objetos, quando estudamos relações entre conjuntos finitos e quando processos (algoritmos) envolvendo um número finito de passos são analisados. Nos últimos anos tornou-se uma disciplina importantíssima porque nos computadores a informação é armazenada e manipulada de forma discreta. A matemática discreta aborda fundamentalmente três tipos de problemas que surgem no estudo de conjuntos e estruturas discretas: *problemas de existência* (existe algum arranjo de objetos de um dado conjunto satisfazendo determinada propriedade?); *problemas de contagem* (quantos arranjos ou configurações desse tipo existem?); *problemas de otimização* (de todas as configurações possíveis, qual é a melhor, de acordo com determinado critério?) (PICADO, 2008).

Nesta seção, vamos abordar alguns princípios relacionados à matemática discreta, que nos ajudarão a resolver problemas desse tipo e a compreender melhor algumas situações lógico-matemáticas que estão por trás dos mais diversos sistemas computacionais.

Um princípio imprescindível na matemática discreta é o princípio da contagem. O ramo da Matemática que trata da contagem é a Combinatória. Tratar a contagem é importante; sempre que temos recursos finitos, por exemplo, os recursos computacionais, tais como a capacidade de processamento, espaço em disco, memória, tamanho das bases de dados. Além disso, é possível verificar a eficiência de um algoritmo, uma vez que um algoritmo pode ser elaborado de diferentes maneiras e, dependendo da forma de implementação e quantidade de entradas (número de variáveis), pode demandar um maior ou menor tempo para ser executado. Logo, o conhecimento sobre contagem também auxilia na análise do tempo de execução e quantidade de memória consumida, ou seja, a análise da complexidade de um algoritmo. Problemas de contagem normalmente se resumem em 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) não 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.

Exemplificando

A lista (2, 4, 8, 16) tem comprimento quatro.

A lista (3, 4, 5, 5, 6) tem comprimento cinco.

A lista (10, 11) tem comprimento dois e também é chamada de par ordenado.

A lista ( ) tem comprimento zero e é chamada de lista vazia.

Outra expressão utilizada para representar listas é *upla*. Uma lista de *n* elementos é conhecida como uma *n*-upla (lê-se: ênupla) (SCHEINERMAN, 2015).

Você pode ter associado a lista de dois elementos (par ordenado) à forma de representação de um ponto no plano cartesiano (coordenadas (x, y)). E é isso mesmo! As listas estão presentes em uma série de aplicações na matemática e muito além. Um número é uma lista de algarismos; uma palavra é uma lista de letras; um identificador em um programa de computador pode ser uma lista de letras e algarismos; a placa de um automóvel é uma lista de letras e algarismos, o número de um telefone é uma lista de algarismos, o código de barras é uma lista de algarismos; só para citar algumas aplicações das listas. E uma questão com a qual frequentemente nos deparamos é: quantas listas podemos formar?

Vamos considerar a necessidade de se fazer uma lista de comprimento dois em que seus elementos sejam uma das letras A, B, C ou D. Quantas listas podemos formar?

A forma mais direta de resolver esse problema é descrever todas as possibilidades, conforme indicado a seguir:

$$\begin{array}{cccc} (A,A) & (A,B) & (A,C) & (A,D) \\ (B,A) & (B,B) & (B,C) & (B,D) \\ (C,A) & (C,B) & (C,C) & (C,D) \\ (D,A) & (D,B) & (D,C) & (D,D) \end{array}$$

Procuramos organizar as listas de modo a não se esquecer de registrar nenhuma possibilidade. Perceba que na primeira linha registramos todas as listas que começam com a letra A; na segunda, todas as listas que começam com a letra B e assim por diante, chegando ao total de $4 \times 4 = 16$ listas de comprimento 2.

Consideremos agora um problema mais geral, em que desejamos descobrir quantas listas de comprimento 2 podemos formar com os algarismos de 1 a n (listas de 2 elementos em que há n escolhas possíveis). Vamos então descrever todas as possibilidades:

$$\begin{array}{cccc} (1,1) & (1,2) & \cdots & (1,n) \\ (2,1) & (2,2) & \cdots & (2,n) \\ \vdots & \vdots & \ddots & \vdots \\ (n,1) & (n,2) & \cdots & (n,n) \end{array}$$

Na primeira linha registramos todas as listas que começam com o algarismo 1; na segunda, todas as listas que começam com o algarismo 2; e assim por diante. Como há n linhas e cada linha tem exatamente n listas, chegamos ao total de $n \times n = n^2$ listas possíveis.

Assimile

Considere uma situação em que se deseja descobrir quantas listas de comprimento 2 podemos formar, sendo que para o primeiro elemento da lista temos *n* opções de escolha e para o segundo elemento da lista temos *m* opções de escolha. E agora?

Vamos supor que os elementos possíveis na primeira posição da lista sejam números inteiros de 1 a *n* e que os elementos possíveis para a segunda posição sejam números inteiros de 1 a *m*. Podemos construir o seguinte quadro para organizar todas as possibilidades:

$$\begin{matrix} (1,1) & (1,2) & \cdots & (1,m) \\ (2,1) & (2,2) & \cdots & (2,m) \\ \vdots & \vdots & \ddots & \vdots \\ (n,1) & (n,2) & \cdots & (n,m) \end{matrix}$$

Na primeira linha registramos todas as listas que começam com o algarismo 1; na segunda, todas as listas que começam com o algarismo 2*;* e assim por diante. Como há *n* linhas (para cada primeira escolha possível) e cada linha tem exatamente *m* listas, concluímos que o total de listas possíveis é

$$\underbrace{m+m+\ldots+m}_{n\ vezes}=m\times n$$

Consideremos, por exemplo, o problema de descobrir quantas listas de comprimento dois podem ser formadas, sabendo que o primeiro elemento da lista (*n)* será uma vogal e o segundo elemento da lista (*m*) será um algarismo de 1 a 9.

Como temos cinco opções para vogal (*A*, *E*, *I, O*, *U*) e nove opções de algarismos (1, 2, 3, 4, 5, 6, 7, 8, 9), o total de listas de comprimento dois que podem ser formadas é $m\times n=9\times 5=45$.

Ao resumir e generalizar essas situações, Scheinerman (2015) nos apresenta o princípio da multiplicação para listas com dois elementos:

Consideremos 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 $n\times m$.

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:

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

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, $6! = 6 \cdot 5 \cdot 4 \cdot 3 \cdot 2 \cdot 1 = 720$. Por definição, considera-se que $1! = 1$ e $0! = 1$ (SCHEINERMAN, 2015).

Exemplificando

De quantas maneiras distintas é possível ordenar três objetos *a, b* e *c*? Observe que neste exemplo não podemos computar o mesmo objeto mais de uma vez, ou seja, em nossa lista, cada um desses objetos aparecerá exatamente uma única vez. De quantos modos diferentes é possível ordená-los? Repare também que não foi mencionado o comprimento da lista, portanto ordenaremos os três objetos *a, b* e *c*, ou seja, o comprimento das listas será igual a três.

Para resolver esse problema podemos utilizar o conceito de fatorial. Como temos três elementos a serem ordenados e estamos interessados em uma ordenação sem repetição, o número total de listas fica determinado como:

$$3! = 3 \cdot 2 \cdot 1 = 6 .$$

De fato, se representarmos todas as listas possíveis temos:

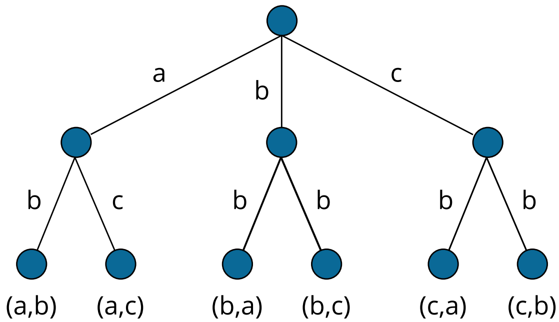
$$\begin{matrix} (a,b,c) & (a,c,b) \\ (b,a,c) & (b,c,a) \\ (c,a,b) & (c,b,a) \end{matrix}$$

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; na Figura 1.2 é representado pelo círculo. O nó raiz 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). Na Figura 1.2 podem ser observadas três ramificações partindo do nó raiz. As ramificações que partem de um nó representam as alternativas possíveis para se chegar a resultados ou ações e, nesse sentido, cada ramo indica um possível resultado. Cada um desses resultados leva a nós adicionais, que se ramificam em outras possibilidades. Assim, cria-se uma forma de árvore (LUCIDCHART, 2020).

Adaptando seu uso para a Combinatória, as árvores de decisão (chamadas neste caso de Diagramas de Árvore) servem para ilustrar o número de possibilidades de um evento baseado em uma série de opções possíveis. Considere, por exemplo, o caso particular de se determinar quantas listas de comprimento dois, de elementos distintos, é possível formar a partir da ordenação de três objetos arbitrários *a, b* e *c.* Para essa situação podemos construir a árvore de decisão indicada na Figura 1.2:

Figura 1.2 | Árvore de decisão para três elementos tomados três a três



Fonte: elaborada pelo autor.

Neste problema foi mencionado que o comprimento da lista deveria ser igual a dois. Por isso, todas as listas encontradas têm apenas dois elementos. A vantagem de utilização de uma árvore de decisão (Diagrama de Árvore) é que além de determinar a quantidade de listas que podem ser formadas, ela também discrimina quais são essas listas. Conhecer não apenas quantas listas podem ser formadas, mas também quais são essas listas, pode ser importante durante um processo de tomada de decisão. Por outro lado, se tivermos um número grande de elementos a combinar, o desenho da árvore de decisão apresentará muitos nós e ramificações. A representação poderá se tornar complexa e, dependendo da sua utilização, poderá até mesmo ser descartada em virtude de não colaborar para com uma melhor compreensão do problema em questão.

Vimos, até aqui, que a determinação da quantidade de listas que podem ser formadas a partir de um número qualquer de elementos é basicamente um problema de contagem. Vimos também que as listas podem conter ou não conter elementos repetidos e que, para determinar o número de listas que podem ser formadas, além de utilizarmos o princípio multiplicativo, podemos também utilizar as árvores de decisão. 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.

De acordo com Iezzi *et al.* (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 $A_{n,p} = \dfrac{n!}{(n-p)!}$. Para ilustrar esse conceito, 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, temos que:

$$A_{4,2} = \frac{n!}{(n-p)!} = \frac{4!}{(4-2)!} = \frac{4!}{2!} = \frac{4\cdot 3\cdot 2\cdot 1}{2\cdot 1} = \frac{24}{2} = 12$$

De fato, discriminando todos os arranjos chegamos a 12 possibilidades:

$$\begin{matrix} (1,2) & (1,3) & (1,4) & (2,1) & (2,3) & (2,4) \\ (3,1) & (3,2) & (3,4) & (4,1) & (4,2) & (4,3) \end{matrix}$$

Note que $(2,3) \neq (3,2)$, isto é, ao trocarmos a ordem dos elementos, obtemos um novo agrupamento. Como a ordem dos elementos é relevante, podemos afirmar que os arranjos são um tipo de lista.

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. Considere, por exemplo, 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 $n = p$:

$$A_{6,6} = \frac{n!}{(n-p)!} = \frac{6!}{(6-6)!} = \frac{6!}{0!} = \frac{6\cdot 5\cdot 4\cdot 3\cdot 2\cdot 1}{1} = 720$$

Ou seja, existem 720 maneiras distintas para organização dessa fila. Como nas permutações sempre temos $n = p$, podemos simplificar a fórmula matemática a ser utilizada. Observe:

$$P_n = A_{n,n} = \frac{n!}{(n-n)!} = \frac{n!}{0!} = \frac{n!}{1} = n!$$

Portanto, nesse exemplo de determinação do número de filas poderíamos simplesmente ter efetuado $P_6 = 6! = 720$.

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 $C_{n,p} = \frac{n!}{p!(n-p)!}$.

Considere, por exemplo, que dos cinco funcionários *A*, *B*, *C*, *D* e *E* de uma empresa do setor de Tecnologia da Informação, três serão promovidos. Queremos determinar todas as combinações desses cinco funcionários, tomados dois a dois. Podemos calcular o número de combinações utilizando a fórmula $C_{n,p} = \frac{n!}{p!(n-p)!}$, em que $n = 5$ e $p = 2$.

$$C_{5,2} = \frac{n!}{p!(n-p)!} = \frac{5!}{2!(5-2)!} = \frac{5!}{2!3!} = \frac{5\cdot 4\cdot 3\cdot 2\cdot 1}{(2\cdot 1)(3\cdot 2\cdot 1)} = \frac{120}{2\cdot 6} = \frac{120}{12} = 10.$$

Temos, portanto, 10 possibilidades de escolha de três funcionários para serem promovidos. Discriminamos a seguir todas essas possibilidades.

$$(A,B,C)\quad (A,B,D)\quad (A,B,E)\quad (A,C,D)\quad (A,C,E)$$
$$(A,D,E)\quad (B,C,D)\quad (B,C,E)\quad (B,D,E)\quad (C,D,E)$$

Observe que, para esse problema, a ordem dos elementos não faz diferença, ou seja, tanto faz considerarmos o conjunto (A,B,C) ou (B,C,A), pois os funcionários que estariam recebendo a promoção seriam os mesmos. Os agrupamentos do tipo combinação, por não serem ordenados, não são considerados listas.

Atenção

Tanto arranjo como combinação são agrupamentos de *p* elementos distintos escolhidos a partir de um conjunto de *n* elementos. A diferença é que, no arranjo, se mudarmos a ordem dos elementos de certo agrupamento, obteremos um novo agrupamento; na combinação, mudando a ordem dos elementos de certo agrupamento, obtemos o mesmo agrupamento (IEZZI *et al.*, 2004).

Mais importante do que nos preocuparmos com as fórmulas é sabermos interpretar o problema e dispormos de ferramentas variadas para sua resolução. Vamos, por fim, abordar uma situação em que o conhecimento do número de agrupamentos possíveis de serem realizados é relevante, mas que não depende necessariamente do uso de fórmulas, e sim de uma boa dose de interpretação e raciocínio lógico.

Considere, por suposição, que um diretório possui 4 pastas, denominadas pastas *a*, *b*, *c* e *d*. Em cada pasta podem ser criadas, no máximo, 5 subpastas. Um usuário do diretório criou, ao acaso, 18 subpastas. Qual é a probabilidade de haver exatamente 4 subpastas na pasta *a*?

Sabemos que a probabilidade é a perspectiva favorável (chance) de que algo venha a acontecer, e pode ser calculada dividindo-se o número de possibilidades favoráveis (de interesse) pelo número total de possibilidades.

Pelo princípio multiplicativo sabemos que poderiam ser criadas um total de $4 \cdot 5 = 20$ subpastas, oriundas das quatro pastas distintas *a*, *b*, *c* e *d*. Como o usuário criou 18 subpastas, teremos duas configurações possíveis: (I) 3 pastas com 5 subpastas e 1 pasta com 3 subpastas ou; (II) 2 pastas com 5 subpastas e as outras duas com 4 subpastas. Vamos, de maneira intuitiva, descrever todas as possibilidades:

Na configuração (I) temos as seguintes possibilidades:

A	*B*	*C*	*D*
3	5	5	5
5	3	5	5
5	5	3	5
5	5	5	3

Na configuração (II) temos as seguintes possibilidades:

A	*B*	*C*	*D*
5	5	4	4
5	4	5	4
5	4	4	5
4	5	5	4
4	5	4	5
4	4	5	5

Considerando as duas configurações temos um total de 10 maneiras de alocação dessas subpastas, sendo que dessas 10 possibilidades, apenas 3 apresentam exatamente 4 subpastas na pasta *a*.

Logo, a probabilidade de haver exatamente 4 subpastas na pasta *a* é $\frac{3}{10} = 0,3 = 30\%$.

Reflita

A árvore de decisão é uma importante ferramenta para auxiliar na tomada de decisões e pode ser muito utilizada em diversos algoritmos, o qual se trata de uma sequência ordenada de instruções a serem seguidas para se chegar a um objetivo, resultado e ou uma ação. Assim, como seria uma árvore de decisão para o mapeamento de um algoritmo de recomendação, em que encontre o tipo de filme mais adequado para diferentes perfis de usuários com base no que já foi visto e recomende novos filmes?

Nesta seção tomamos contato com alguns conteúdos da chamada matemática discreta, compreendendo as diferentes formas com que objetos podem ser agrupados. Vimos o que são listas, o que é uma árvore de decisão (Diagrama de Árvore) e o que são Arranjos, Permutações e Combinações. Muitos problemas de lógica acabam recaindo em princípios matemáticos, e os princípios aqui estudados serão de grande utilidade no decurso de sua formação.

Sem medo de errar

Você está lembrado do desafio apresentado no início da seção? Como colaborador de uma *startup* de tecnologia, você se deparou com o seguinte problema: uma rede de computadores privada da empresa foi configurada de tal modo que cada um dos 3 escritórios da empresa (A, B, C) tem cinco pontos de acesso à rede. Em um determinado momento, havia 13 computadores conectados à rede e uma sobrecarga foi detectada por um software de gerenciamento de fluxo de dados. Essa sobrecarga foi causada em um escritório em que cinco computadores estavam conectados à rede. Deseja-se saber qual é a probabilidade (chance) de que essa sobrecarga tenha partido do escritório C.

Pelo princípio multiplicativo sabemos que há um total de $3 \cdot 5 = 15$ pontos de acesso a essa rede. Como no momento da detecção da sobrecarga havia 13 computadores conectados à rede, temos dois cenários possíveis: (I) havia dois escritórios com todos os pontos de acesso utilizados e um terceiro escritório utilizando apenas três pontos de acesso ou; (II) havia um escritório utilizando todos os pontos de acesso e dois escritórios utilizando 4 pontos de acesso.

No cenário (I) temos as seguintes possibilidades:

A	*B*	*C*
5	5	3
5	3	5
3	5	5

No cenário (II) temos as seguintes possibilidades:

A	*B*	*C*
5	4	4
4	5	4
4	4	5

Considerando os dois cenários, temos um total de 6 possibilidades de utilização dessa rede no momento da sobrecarga. Dessas 6 possibilidades, em três cenários temos o escritório *C* utilizando todos os seus pontos de acesso.

Poderíamos, portanto, presumir que há $\frac{3}{6} = 0{,}5 = 50\%$ de probabilidade de a sobrecarga ter sido causada pelo escritório *C*. Uma análise mais detalhada, no entanto, revela que em dois dos três cenários em que o escritório *C* utiliza todos os seus pontos de acesso, há outros escritórios que também estão utilizando todos os pontos de acesso. Ao apresentar sua conclusão ao seu superior imediato, esse detalhe não poderá ser omitido.

Faça valer a pena

1. A combinatória é o ramo da Matemática que trata da contagem. Tratar a contagem é importante sempre que temos recursos finitos (quanto espaço um banco de dados consome? quantos usuários a configuração de um computador pode suportar?) ou sempre que estamos interessados em eficiência (quantos cálculos um determinado algoritmo envolve?) (GERSTING, 2017).

Analise as seguintes afirmações acerca de alguns conceitos estudados em Combinatória:

I. Uma lista é uma sequência ordenada de objetos.

II. Ao trabalharmos com agrupamentos do tipo arranjo, se mudarmos a ordem dos elementos obteremos um novo agrupamento.

III. Ao trabalharmos com agrupamentos do tipo combinação, se mudarmos a ordem dos elementos obteremos o mesmo agrupamento.

Considerando os princípios matemáticos relacionados à combinatória, é correto o que se afirma em:

a. I, apenas.

b. I e II, apenas.

c. I e III, apenas.

d. II e III, apenas.

e. I, II e III.

2. Em uma aula de Análise Combinatória foi proposto o seguinte problema:

Um serviço de assinatura de jogos eletrônicos oferece um portfólio com 15 jogos a seus assinantes. O pacote de assinaturas Gold permite ao assinante escolher e baixar três jogos pertencentes a esse portfólio. De quantas maneiras o assinante pode escolher três desses jogos?

Diante da situação apresentada, avalie as seguintes asserções e a relação proposta entre elas.

I. Este é um problema que envolve agrupamentos do tipo arranjo.

PORQUE

II. A ordem dos jogos escolhidos não faz diferença para o assinante.

A respeito dessas asserções, assinale a alternativa correta.

a. As asserções I e II são proposições verdadeiras, e a II é uma justificativa da I.

b. As asserções I e II são proposições verdadeiras, mas a II não é uma justificativa da I.

c. A asserção I é uma proposição verdadeira, e a II é falsa.

d. A asserção I é uma proposição falsa, e a II é verdadeira.

e. As asserções I e II são proposições falsas.

3. Um sistema operacional (SO) é uma coleção de programas que inicializam o hardware do computador. Fornece rotinas básicas para controle de dispositivos. Fornece gerência, escalonamento e interação de tarefas. Mantém a integridade de sistema (LOPES, 2017).

Um sistema operacional de computador permite atribuir nomes aos arquivos utilizando qualquer combinação de letras maiúsculas (A-Z) e de algarismos (0-9), mas o número de caracteres do nome do arquivo deve ser no máximo 8 (e deve haver ao menos um caractere no nome do arquivo). São exemplos de nomes válidos: *Y56, G, 8JJ* e *FGHI7890*. São exemplos de nomes inválidos *B*32* (por ter um caractere não permitido) e *CLARINETE* (por possuir mais do que 8 caracteres (SCHEINERMAN, 2015, adaptado).

Considerando a regra de atribuição de nomes explicitada, a quantidade possível de nomes de arquivos diferentes nesse sistema operacional é determinada por:

a. $\frac{36!}{(36-8)!}$.

b. $\frac{36!}{8!(36-8)!}$.

c. $36+36^2+36^3+36^4+36^5+36^6+36^7+36^8$.

d. $36\cdot35\cdot34\cdot33\cdot32\cdot31\cdot30\cdot29$.

e. $36\cdot36\cdot36\cdot36\cdot36\cdot36\cdot36\cdot36$.

Referências

ABAR, C. A. A. P. **Noções de lógica matemática:** Esboço do desenvolvimento da lógica. Pontifícia Universidade Católica de São Paulo, Faculdade de Ciências Exatas e Tecnologia. [*S. l.*], 2004. Disponível em: https://www.pucsp.br/~logica/. Acesso em: 14 jan. 2020.

ABBAGNANO, N. **Dicionário de filosofia**. São Paulo: Martins Fontes, 2007.

ALCOFORADO, P. Introdução. *In*: FREGE, G. **Lógica e filosofia da linguagem**. 2. ed. rev. São Paulo: Editora da Universidade de São Paulo, 2009.

ALENCAR FILHO, E. **Iniciação à lógica matemática**. São Paulo: Nobel, 2002.

ARISTÓTELES. **Lógica Aristotélica**. 1. ed. Memphis: Books LLC, 2011.

BOYER, C. B. **História da Matemática**. 2. ed. São Paulo: Edgard Blücher, 1996.

BUCHSBAUM, A. **Lógica geral**. São José: UFSC, 2006.

CABRAL, J. F. P. Lógica de Aristóteles. **Brasil Escola**, [s.d.]. Disponível em: https://brasilescola.uol.com.br/filosofia/logica-aristoteles.htm. Acesso em: 14 jan. 2020.

FAJARDO, R. **Lógica Matemática**. 1 ed. São Paulo: Edusp, 2017.

FORBELLONE, A. L. V.; EBERSPACHER, H. F. **Lógica de Programação:** a construção de algoritmos e estruturas de dados. 3. ed. São Paulo: Person Prentice Hall, 2005.

GERSTING, J. L. **Fundamentos matemáticos para a ciência da computação:** matemática discreta e suas aplicações. 7. ed. Rio de Janeiro: LTC, 2017. Disponível em: https://integrada.minhabiblioteca.com.br/#/books/9788521633303/cfi/6/10!/4/50@0:3.16. Acesso em: 9 jan. 2020.

IEZZI, G. *et al*. **Matemática:** ciências e aplicações. 2. ed. São Paulo: Atual, 2004.

JAPIASSÚ, H.; MARCONDES, D. **Dicionário básico de filosofia**. 4. ed. aum. Rio de Janeiro: Jorge Zahar, 2006.

KANT, I. **Crítica da Razão Pura**. Petrópolis: Editora Vozes, 2015.

LOPES, S. **O que é um sistema operacional?** Oficina da Net, [*s. l.*], 30 mar. 2017. Disponível em: https://www.oficinadanet.com.br/artigo/851/o_que_e_um_sistema_operacional. Acesso em: 24 jan. 2020.

LUCIDCHART. **O que é um diagrama de árvore de decisão?** Lucid Software Inc., [*s. l.*], 2020. Disponível em: https://www.lucidchart.com/pages/pt/o-que-e-arvore-de-decisao#section_0. Acesso em: 23 jan. 2020.

MARTINS, E. T.; MARTINS, I. T. A lógica fuzzi na operacionalização de conhecimentos em interação de tarefas humano-computador em máquinas complexas: a aprendizagem em conjuntos de significância. **Revista Internacional de Aprendizaje y Cibersociedad**, [*s. l.*], v. 19, n. 2, p. 153-177, 2015.

MUNDIM, R. P. A Lógica Formal – princípios elementares. **Revista Economia & Gestão**, Belo Horizonte, v. 2, n. 3, jan./jun. 2002.

OLIVEIRA, K. E. C. S. **Uma introdução sobre lógicas não-clássicas**. Bauru: Universidade Estadual Paulista, 2010. Disponível em: http://www2.fc.unesp.br/matematica/semana/arquivos/lnc.pdf. Acesso em: 15 jan. 2020.

PICADO, J. **Que é a matemática discreta?** Coimbra: Departamento de Matemática da Universidade de Coimbra, 2008. Disponível em: http://www.mat.uc.pt/~picado/ediscretas/2008/apontamentos/oque.pdf. Acesso em: 22 jan. 2020.

PINCELLI, R. O paradoxo do queijo suíço contra o câncer. *In*: HIPERCUBIC. **O paradoxo do queijo suíço contra o câncer**. [*S. l.*], 26 fev. 2016. Disponível em: http://scienceblogs.com.br/hypercubic/2016/02/o-paradoxo-do-queijo-suo-contra-o-cncer/. Acesso em: 15 jan. 2020.

ROSEN, K. H. **Matemática discreta e suas aplicações**. 6. ed. Porto Alegre: AMGH, 2010. Disponível em: https://integrada.minhabiblioteca.com.br/#/books/9788563308399/cfi/2!/4/4@0.00:47.7. Acesso em: 24 jan. 2020.

SANTOS, R. **Paradoxos semânticos**. Lisboa: Centro de Filosofia da Universidade de Lisboa, 2014.

SCHEINERMAN, E. R. **Matemática discreta:** uma introdução. São Paulo: Cengage Learning, 2015.

TARSKI, A.; HELMER, O. **Introduction to logic and to the methodology of the deductive sciences.** 1. ed. Nova York: Oxford University Press, 1946.

ZEGARELLI, M. **Lógica para leigos**. Rio de Janeiro: Alta Books, 2013.

Unidade 2

Gilberto Vieira

Álgebra de conjuntos

Convite ao estudo

Olá, caro aluno, seja bem-vindo! Nesta unidade você terá a oportunidade de conhecer a Teoria de Conjuntos (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 ligados a várias áreas, como a computação.

Considere, por exemplo, situações em que seja necessário contabilizar o número de subconjuntos (possibilidades) derivados de outro conjunto; identificar a quantidade de elementos que gozam de determinada característica e/ou propriedade; e estudar relações entre conjuntos. Compreender a linguagem de conjuntos possibilitará a abordagem desses problemas. Tais situações serão apresentadas, exemplificadas e discutidas nesta unidade. Na primeira seção, você aprenderá a definição de conjunto, as diferentes formas de apresentação de conjuntos e a simbologia (sintaxe) utilizada para representar as relações entre conjuntos, entre elas as relações de pertinência e continência. Já na segunda seção, iniciaremos o estudo das operações de conjuntos. Algumas dessas operações, como as operações união e intersecção, por exemplo, você já deve ter estudado no Ensino Médio. Será uma excelente oportunidade para retomar e aprofundar os estudos a respeito dessas operações, relacionando-as aos conectivos lógicos e $(\wedge)$ e ou $(\vee)$, utilizados para a compreensão da lógica de programação e importantes para entendimento e elaboração de algoritmos computacionais para resolução de problemas do cotidiano. Já na última seção desta unidade, estudaremos algumas aplicações da Teoria de Conjuntos em Computação, destacando as relações lógicas implícitas nas diferentes operações de conjuntos. A partir dos conteúdos estudados nesta unidade, além de conhecer e entender as teorias e álgebras de conjuntos, suas operações e diferenças, você irá desenvolver a habilidade de conhecer os conceitos de conjuntos e suas aplicações em diferentes casos. Bons estudos!

Seção 1

Teoria dos conjuntos

Diálogo aberto

Prezado aluno, nesta seção serão introduzidos os principais conceitos da Teoria de Conjuntos. Partindo da noção intuitiva de conjunto como uma coleção não ordenada de objetos, iremos estudar uma série de conceitos, como subconjuntos, igualdade de conjuntos, formas de representação de conjuntos (sentenças e diagramas), quantificadores (universal e existencial) e relações de pertinência (elemento – conjuntos) e continência (conjunto – conjunto). Além disso, também teremos um primeiro contato com a notação de conjuntos, que compreende uma série de símbolos e signos específicos.

Considere as seguintes questões: você sabe o que é a cardinalidade de um conjunto? Você sabe demonstrar a igualdade entre dois conjuntos? Você sabe determinar e identificar todos os subconjuntos derivados de um determinado conjunto? Nesta seção, você aprenderá a responder questões como essas. Profissionais da área da computação constantemente se deparam com situações desse tipo.

Você foi contratado por uma grande empresa do setor de Tecnologia da Informação (TI) e está trabalhando em uma equipe no desenvolvimento de um aplicativo para telefones móveis. Você ficou responsável por elaborar diferentes chaves de acesso para esse aplicativo, e durante esse trabalho, você se deparou com o problema de identificar quantos e quais são os subconjuntos derivados de um conjunto constituído por quatro elementos arbitrários {1, 2, 3, 4}. As possíveis combinações (subconjuntos) encontradas deverão ser apresentadas para o restante da equipe que utilizará essas informações para finalizar o layout da tela inicial do aplicativo. Você conseguiria determinar quantos subconjuntos podem ser formados, independentemente do seu “tamanho” (número de elementos) a partir da conjugação dos quatro elementos arbitrários? Você saberia identificar todos esses subconjuntos? Determinar quantos subconjuntos podem ser formados, independentemente do seu “tamanho” (número de elementos) e identificá-los (um a um) é uma tarefa que exige raciocínio combinatório, metodologia e organização. Para resolver tal problema, será necessário compreender o significado de subconjunto e as relações entre conjuntos.

Prossiga estudando e aprenda os conceitos necessários para a resolução dessa situação-problema!

Não pode faltar

Conjuntos podem ser definidos como coleções não ordenadas de objetos que podem ser, de alguma forma, relacionados (FERREIRA, 2001). Considere, por exemplo, o conjunto A das cores da bandeira do Brasil. Temos que A = {verde, amarelo, azul, branco}. Em geral, objetos de um mesmo conjunto gozam de uma propriedade em comum. Assim, qualquer objeto que verifique a propriedade "ser cor da bandeira do Brasil" será considerado um elemento do conjunto A. Costuma-se utilizar letras maiúsculas do nosso alfabeto para representar os conjuntos.

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

- Listando todos os elementos do conjunto, como foi feito com o conjunto A, que representa as cores da bandeira do Brasil. Dependendo do conjunto que se deseja descrever, essa maneira de representação pode se revelar trabalhosa, sendo mais indicada para conjuntos finitos, com um pequeno número de elementos.

- Indicando os primeiros elementos do conjunto (presumindo que os elementos do conjunto possam ser ordenados) que denotem um padrão para uma listagem indefinida. Por exemplo, considere o conjunto $B = \{2, 4, 6, \ldots\}$. É possível deduzir, a partir do padrão indicado, que o conjunto B é um conjunto infinito, constituído pelos números inteiros positivos pares.

- Escrevendo uma propriedade que caracterize os elementos que constituem o conjunto. Por exemplo, considere o conjunto C = { $x | x$ é um número inteiro e $4 < x \leq 9$ }. Lê-se: C é o conjunto de todos os x , tal que x é inteiro, maior do que 4 e menor ou igual a 9. Listando todos os elementos de C, tem-se que: $C = \{5, 6, 7, 8, 9\}$. 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 possuem um número muito grande de elementos (ou até mesmo conjuntos infinitos), a listagem de todos os elementos do conjunto não se torna viável.

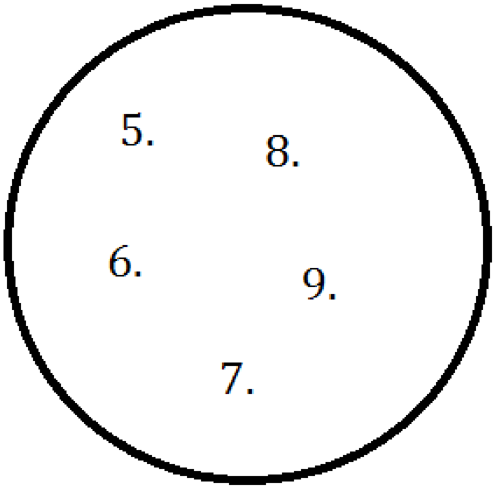
Assimile

Considere o conjunto $D = \{3, 5, 7, \ldots\}$ representado a partir de seus primeiros elementos. Tal forma de representação pode suscitar dúvidas: trata-se do conjunto dos números inteiros ímpares maiores do que 1 ou dos conjuntos dos números inteiros primos maiores ou iguais a 3? Tal descrição não possibilita inferirmos de que grupo se trata. Por isso, escrever por extenso a propriedade característica dos elementos do conjunto, mostra-se como uma descrição mais clara e precisa do que as outras maneiras de descrição apresentadas.

Há ainda uma maneira alternativa de representação de conjuntos com forte apelo visual. Trata-se dos Diagramas 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 baseado nos círculos eulerianos, por isso, alguns autores referem-se aos diagramas de Venn como diagramas de Euler-Venn (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 $4 < x \leq 9 \}$ pode ser representado pelo diagrama apresentado na Figura 2.1.

Figura 2.1 | Conjunto C



Fonte: elaborada pelo autor.

Um objeto pertencente a um conjunto é chamado de elemento do conjunto. A relação de pertinência é indicada pelo símbolo $\in$, e a relação de não pertinência, pelo símbolo $\notin$. Assim, a indicação $x \in A$ significa que o objeto x é um elemento do conjunto A. Retomando o conjunto das cores da bandeira do Brasil A = {verde, amarelo, azul, branco}, podemos afirmar que verde$\in A$ e que vermelho$\notin A$. A relação $\in$ pode ser lida como "é membro de" ou "está em" ou "é elemento de" ou "pertence a".

Além de saber descrever um conjunto a partir das propriedades comuns de seus elementos, também é importante saber determinar quantos elementos constituem o conjunto. De acordo com Scheinerman (2015), ao número de objetos (elementos) de um determinado conjunto dá-se o nome de cardinalidade. As barras de valor absoluto em torno de um conjunto representam a cardinalidade ou o tamanho do conjunto (isto é, o número de elementos do conjunto). Assim, considerando o conjunto A = {verde, amarelo, azul, branco}, pode-se afirmar que a cardinalidade de A é igual a 4, ou seja, $|A| = 4$. De maneira análoga, considerando o conjunto C = { $x|x$ é um número inteiro e $4 < x \leq 9$ }, pode-se afirmar que a cardinalidade de C é igual a 5, ou seja, $|C| = 5$.

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.

Exemplificando

Considere os seguintes conjuntos: A = { $x|x$ é um mês com exatamente 30 dias}, B = {1, 4, 9, 16, 25,...} e C = { $x|x$ é um número inteiro e $x > 3$ e $x < 4$ }.

Tem-se que A = {abril, junho, setembro, novembro} e, portanto, $|A| = 4$. O conjunto B é o conjunto dos números quadrados perfeitos. Como existem infinitos números quadrados perfeitos, não é possível determinar a cardinalidade de B. Assim, dizemos que o conjunto B é infinito. Já o conjunto C é o conjunto dos números inteiros que são, ao mesmo tempo, maiores do que 3 e menores do que 4. Como não existe nenhum número inteiro que atenda simultaneamente a essas duas propriedades, dizemos que o conjunto C é um conjunto vazio. Essa afirmação pode ser representada como C = { } ou C = $\varnothing$. Dizemos que a cardinalidade de C é igual a zero, ou seja, $|C| = 0$. Como a cardinalidade de um conjunto vazio é igual a zero, e zero é um número inteiro, podemos afirmar que o conjunto vazio também é um conjunto finito.

É muito comum nos deparamos com alguns conjuntos padrão, largamente utilizados na matemática:

$\mathbb{N}$ = conjunto de todos os números inteiros não negativos. Perceba que $0 \in \mathbb{N}$.

$\mathbb{Z}$ = conjunto de todos os números inteiros.

$\mathbb{Q}$ = conjunto de todos os números racionais.

$\mathbb{R}$ = conjunto de todos os números reais.

$\mathbb{C}$ = conjunto de todos os números complexos.

Quando dois conjuntos têm exatamente os mesmos elementos, dizemos que esses conjuntos são iguais. Para provar que dois conjuntos A e B são iguais, devemos mostrar que todo elemento de A é também elemento de B, e vice-versa.

Reflita

Vimos que dois conjuntos são iguais se, e somente se, eles contêm os mesmos elementos. Considere os seguintes conjuntos:

E = { $x \in \mathbb{Z} \middle| x$ é par} e

F = { $y \in \mathbb{Z} \middle| y = a + b$, em que a e b são ímpares}.

Podemos afirmar que $E = F$, ou seja, que todo elemento de E é também elemento de F e vice-versa? Reflita sobre as propriedades desses conjuntos e tente elaborar um esquema que valide ou refute a igualdade $E = F$.

Na Teoria de Conjuntos, há certas afirmações que não podem ser escritas adequadamente por meio de símbolos proposicionais, parênteses e conectivos lógicos. Elas contêm um elemento quantificador. Os quantificadores são frases como "para todo", "para cada" ou "para algum", que indicam, de alguma forma, quantos objetos têm uma determinada propriedade (GERSTING, 1995).

Por exemplo, considere as seguintes afirmações:

Todo inteiro é par ou ímpar.

Existe um número natural que é primo e par.

A primeira afirmação traduz uma relação de universalidade, já a segunda afirmação traduz uma relação de existência. Utilizaremos uma notação específica para representar cada uma dessas situações.

O quantificador universal é simbolizado por um A de cabeça para baixo, $\forall$, e é lido "para todo" ou "qualquer que seja". A forma geral para essa notação é $\forall x \in A$, afirmações sobre x . A primeira afirmação "todo inteiro é par ou ímpar" ficaria representada como $\forall x \in \mathbb{Z}$, x é par ou x é ímpar.

O quantificador existencial é simbolizado por um E espelhado, $\exists$, e é lido como "há" ou "existe". A forma geral para essa notação é $\exists x \in A$, afirmações sobre x . A segunda afirmação "existe um número natural que é primo e par" ficaria representada como $\exists x \in \mathbb{N}$, x é primo e par.

Exemplificando

Um exemplo de utilização do quantificador universal pode ser verificado no enunciado a seguir, em que se deseja provar que todo número inteiro divisível por 10 é par.

Seja A = { $x \in \mathbb{Z} \mid 10$ divide x }. Queremos provar que $\forall x \in A$, x é par.

Prova: Seja $x \in A$, ou seja, x é um número inteiro divisível por 10. Isso significa que existe um inteiro y , de modo que $x = 10y$. Podemos escrever essa igualdade como $x = (2 \cdot 5)y = 2 \cdot (5y)$. Portanto, x é divisível por 2 e, consequentemente, x é par.

Considere, agora, os conjuntos $A = \{2,5,7,9\}$ e $B = \{1,2,3,4,5,6,7,8,9\}$. Perceba que todos os elementos pertencentes ao conjunto A também pertencem ao conjunto B. Nesse caso, dizemos que A é um subconjunto de B.

Sejam os conjuntos A e B. Dizemos que A é um subconjunto de B se, e somente se, todo elemento de A também for elemento de B. A notação $A \subseteq B$ significa que A é subconjunto de B (SCHEINERMAN, 2015).

Se A é um subconjunto de B, mas $A \neq B$, ou seja, existe pelo menos um elemento de B que não é elemento de A, então, A é chamado de subconjunto próprio de B. Subconjuntos próprios podem alternativamente ser representados pelo sinal $\subset$.

Atenção

Observe que $\subseteq$ e $\in$ têm significados relacionados, porém, diferentes!

O símbolo $\in$ é utilizado para representar uma relação de pertinência entre um objeto e um conjunto. Por exemplo, seja o conjunto $A = \{5,7,9,11,13\}$, podemos afirmar que $7 \in A$. Já o símbolo $\subseteq$ é utilizado para representar uma relação de continência (subconjunto) entre conjuntos. Por exemplo, seja $A = \{5,7,9,11,13\}$ e $B = \{5,11,13\}$, podemos afirmar que $B \subseteq A$.

Tome cuidado! Os sinais $\subseteq$ e $\in$ não podem ser permutados.

Para provar que um conjunto N é subconjunto de um conjunto M, devemos mostrar que todo elemento de N é também elemento de M (perceba que, nesse caso, a recíproca não precisa ser verdadeira). Por exemplo, considere os conjuntos M = $\{x|x$ é múltiplo de 5} e N = $\{x|x$ é múltiplo de 10}. Vamos provar que $N \subseteq M$.

Temos que x satisfaz a propriedade característica de N, ou seja, x é múltiplo de 10. Logo, podemos escrever $x = 10 \cdot y$ para algum inteiro y. Essa igualdade pode ser reescrita como $x = (5 \cdot 2) \cdot y = 5 \cdot (2.y)$ ou $x = 5 \cdot k$, em que $k = 2 \cdot y$, de tal forma que k também é um inteiro. Isso mostra que x também é múltiplo de 5, ou seja, x também satisfaz a propriedade característica de M, portanto, $N \subseteq M$.

Um problema recorrente envolvendo subconjuntos diz respeito à determinação do número de subconjuntos de um determinado conjunto. Por exemplo, quantos subconjuntos tem o conjunto $A = \{a, b, c\}$?

Uma maneira para resolver esse problema é listar todas as possibilidades. Como a cardinalidade de A é igual a 3 $(|A| = 3)$, qualquer subconjunto de A pode ter de zero a três elementos. Consideremos o Quadro 2.1 com a descrição de todas as possibilidades:

Quadro 2.1 | Subconjuntos de A (cardinalidade 3)

Número de elementos	Subconjuntos	Número de subconjuntos
0	$\varnothing$	1
1	{a}, {b}, {c}	3
2	{a, b}, {a, c}, {b, c}	3
3	{a, b, c}	1
Total		8

Fonte: elaborado pelo autor.

Portanto, $A = \{a, b, c\}$ tem oito subconjuntos. É importante, nesse momento, retomar a definição de conjunto como uma coleção não ordenada de objetos. Isso significa dizer que o subconjunto {a, b} e o subconjunto {b, a} são iguais e, portanto, contabilizados uma única vez. Também é importante destacar que o conjunto vazio e o próprio conjunto A também são subconjuntos de A.

Nesse exemplo, conseguimos listar e contabilizar todos os subconjuntos de A. Como $|A| = 3$, essa tarefa não se mostrou muito trabalhosa. No entanto, imagine no caso de um conjunto com cardinalidade igual a 10: quantos subconjuntos poderiam ser contabilizados?

O teorema a seguir permite contabilizar o número de subconjuntos de um conjunto qualquer conhecendo-se a sua cardinalidade.

Teorema: seja A um conjunto finito. O número de subconjuntos de A é $2^{|A|}$ (SCHEINERMAN, 2015).

Exemplificando

Seja A um conjunto com cardinalidade igual a 10, quantos subconjuntos de A poderiam ser contabilizados?

Como $|A| = 10$, utilizando o resultado do Teorema anterior, o número de subconjuntos de A é igual a $2^{|A|} = 2^{10} = 1024$.

Nesta seção, você, aluno, aprendeu o que são conjuntos, como podemos representar os conjuntos, o que é cardinalidade de um conjunto, o que são quantificadores (universal e existencial) e como podemos listar e contabilizar os subconjuntos de um determinado conjunto. Todos esses conceitos serão de suma importância na resolução de situações-problema que envolvam conjuntos e relações entre conjuntos.

Sem medo de errar

Você se lembra da situação-problema apresentada no início da seção? Vamos retomá-la?

Você foi contratado por uma grande empresa do setor de Tecnologia da Informação (TI) e está trabalhando em uma equipe no desenvolvimento de um aplicativo para telefones móveis. Durante o desenvolvimento do aplicativo, você se deparou com o problema de identificar quantos e quais são os subconjuntos derivados de um conjunto constituído por quatro elementos arbitrários $\{1, 2, 3, 4\}$. Você conseguiria determinar quantos subconjuntos podem ser formados, independentemente do seu "tamanho" (número de elementos) e a partir da conjugação desses quatro elementos arbitrários? Você saberia identificar todos esses subconjuntos? Lembre-se de que as possíveis combinações (subconjuntos) encontradas deverão ser apresentadas para o restante da equipe, que utilizará essas informações para finalizar o layout da tela inicial do aplicativo.

Em primeiro lugar, vamos elucidar o significado da palavra arbitrário. Quando dizemos que x é um elemento arbitrário de um conjunto A, queremos dizer que x pode ser qualquer elemento de A e não podemos fazer qualquer outra suposição sobre x.

Denotando o conjunto $\{1,2,3,4\}$ por A, podemos determinar a sua cardinalidade. Temos que $|A| = 4$, logo, o número de subconjuntos de A é igual a $2^{|A|} = 2^4 = 16$. Resta, agora, identificar todos esses 16 subconjuntos. Como a cardinalidade de A é igual a 4, podemos ter subconjuntos com 0, 1, 2, 3 ou 4 elementos. Por exemplo, {1, 3, 4} é subconjunto de A, pois $\{1,3,4\} \subseteq A$. {2, 3} também é subconjunto de A, pois $\{2,3\} \subseteq A$.

Para identificar todos os subconjuntos de A, vamos elaborar uma lista (Quadro 2.2):

Quadro 2.2 | Subconjuntos de A (cardinalidade 4)

Número de elementos	Subconjuntos	Número de subconjuntos
0	$\varnothing$	1
1	{1}, {2}, {3}, {4}	4
2	{1, 2}, {1, 3}, {1, 4}, {2, 3}, {2, 4}, {3, 4}	6
3	{1, 2, 3}, {1, 2, 4}, {1, 3, 4}, {2, 3, 4}	4
4	{1, 2, 3, 4}	1
Total		16

Fonte: elaborado pelo autor.

Vale ressaltar que, como a cardinalidade do conjunto A era igual a 4, não tivemos maiores dificuldades em listar todos os 16 subconjuntos. Problemas envolvendo conjuntos de maior cardinalidade costumam exigir a contabilização do número de subconjuntos, sem necessariamente precisar identificá-los. Inúmeros problemas envolvendo o raciocínio computacional exigem essa habilidade.

Você conseguiria determinar quantos subconjuntos poderiam ser obtidos a partir de um conjunto com cardinalidade igual a 20? Que estratégia você utilizaria para identificá-los (um a um)? Pense nisso!

Faça valer a pena

1. Para compreender bem as definições e os teoremas que constituem as teorias matemáticas, é indispensável habituarmo-nos a utilizar uma linguagem mais precisa e rigorosa do que a que utilizamos, em geral, na vida corrente. A aquisição desse hábito pode ser muito facilitada pelo recurso de algumas noções e alguns símbolos da Lógica Matemática (FERREIRA, 2001).

Seja o conjunto $A = \{1,2,3,4,7,8,10,13\}$ e o conjunto $B = \{4,7,8,10\}$, considere as seguintes afirmações:

I. $4 \in A$.

II. $6 \notin A$.

III. $B \in A$.

Considerando as relações de pertinência e continência entre conjuntos, estão adequadamente representadas as relações indicadas em:

a. I, apenas.

b. I e II, apenas.

c. I e III, apenas.

d. II e III, apenas.

e. I, II e III.

2. Determinar a quantidade de subconjuntos que pode ser formada a partir de um conjunto, independentemente de sua cardinalidade (número de elementos), é uma tarefa que exige raciocínio combinatório, metodologia e organização. Problemas dessa natureza frequentemente acometem profissionais da área da Ciência da Computação.

Pode-se afirmar que o conjunto $A = \{1,2,3,4,5,6\}$ possui:

a. 12 subconjuntos.

b. 32 subconjuntos.

c. 64 subconjuntos.

d. 128 subconjuntos.

e. 256 subconjuntos.

3. Para descrevermos determinado conjunto, é necessário identificarmos os seus elementos. A maneira mais usual de descrever um conjunto é escrevendo uma propriedade que caracterize os elementos que constituem o conjunto. Para tanto, faz-se necessário o domínio da notação (simbologia) inerente à Teoria de Conjuntos.

Considere o conjunto A descrito como:

$A = \{\, x \mid (\exists y)(y \in \{0, 1, 2\} \text{ e } x = y^3) \,\}$

Portanto, o conjunto A pode ser representado como:

a. {0, 1, 2}.

b. {0, 3, 6}.

c. {1, 2, 3}.

d. {1, 3, 9}.

e. {0, 1, 8}.

Seção 2

Álgebra de conjuntos

Diálogo aberto

Prezado aluno, nesta seção estudaremos as operações de conjuntos. Conjuntos são coleções não ordenadas de objetos que podem ser, de alguma forma, relacionados (FERREIRA, 2001). Estudaremos, portanto, que relações são essas, direcionando nosso olhar especialmente para as operações de união $(\cup)$, intersecção $(\cap)$ e diferença $(-)$ entre conjuntos.

Lembre-se de que você foi contratado por uma grande empresa do setor de Tecnologia da Informação (TI) e está trabalhando em uma equipe de desenvolvimento de um aplicativo para telefones móveis. No desenvolvimento desse aplicativo, por sua vez, surgiu um novo problema: a equipe de desenvolvimento de software percebeu que alguns comandos de busca desse aplicativo (ações solicitadas pelo usuário para sua execução) direcionam a busca do usuário para o Banco de Dados A, enquanto outros comandos direcionam a busca do usuário para o Banco de Dados B e, ainda, há comandos que realizam essa busca em ambos os Bancos de Dados (A e B).

Considere que há 20 comandos que direcionam a busca do usuário para o Banco de Dados A, que no total há 60 comandos distintos, e que 12 comandos direcionam a busca para os Bancos de Dados A e B. Você saberia informar quantos desses 60 comandos realizam a busca no Banco de Dados B?

Você deverá apresentar a representação e resolução desse problema à equipe de desenvolvimento de software para que ela decida sobre a viabilidade de unificação desses Bancos de Dados.

Perceba que, apesar de parecer ser um problema de simples solução, é necessário tomar cuidado para não contabilizar comandos em duplicidade. E agora, como resolver essa situação?

Nesta seção, você aprenderá a encontrar a união, a interseção e a diferença entre conjuntos e a utilizar essas relações na resolução de situações-problema como essa.

Bons estudos!

Não pode faltar

Em Matemática, quando nos referimos a operações, automaticamente nos recordamos das operações numéricas fundamentais (adição, subtração, multiplicação e divisão), porém, em Teoria de Conjuntos também há várias operações que 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. Dentre essas operações, as mais fundamentais são denominadas união e intersecção. A operação união é representada pelo símbolo $\cup$ e a operação intersecção pelo símbolo $\cap$.

Consideremos, por exemplo, o conjunto M constituído por todos os alunos de uma determinada universidade. Podemos afirmar que o conjunto A, formado pelos alunos do curso de Análise e Desenvolvimento de Sistemas dessa mesma universidade, é um subconjunto de M ($A \subseteq M$), ou seja, cada aluno que pertence ao conjunto A (alunos do curso de Análise e Desenvolvimento de Sistemas) também pertence ao conjunto M (também são alunos da universidade). Analogamente, consideremos também o conjunto B, formado pelos alunos do curso de Matemática dessa mesma universidade, logo, podemos afirmar que B também é subconjunto de M ($B \subseteq M$).

Um novo conjunto de alunos pode ser definido como consistindo em todos os alunos que sejam estudantes do curso de Análise e Desenvolvimento de Sistemas ou Matemática (ou ambos). Esse conjunto é chamado de união de A e B. Segundo Gersting (1995), a operação de união de A e B pode ser denotada como $A \cup B = \{x | x \in A \text{ ou } x \in B\}$. Nesse exemplo, a união $A \cup B$ compreende todos os alunos que cursam Análise e Desenvolvimento de Sistemas (A) ou Matemática (B) ou ambos os cursos.

Outro conjunto pode ser definido como sendo composto por todos os alunos do curso de Análise e Desenvolvimento de Sistemas e do curso de Matemática, ou seja, alunos que cursam simultaneamente ambos os cursos. Esse novo conjunto (que pode ser vazio) é chamado de intersecção de A e B. Segundo Gersting (1995), a operação de intersecção de A e B pode ser denotada como $A \cap B = \{x | x \in \text{A e } x \in \text{B}\}$. Nesse exemplo, a união $A \cap B$ compreende todos os alunos que cursam simultaneamente Análise e Desenvolvimento de Sistemas (A) e Matemática (B).

Exemplificando

Exemplo 1: 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 todos os 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 $(\cup)$, esses elementos são contabilizados uma única vez. Em relação à cardinalidade desses conjuntos, temos que: $|A|=6$, $|B|=7$ e $|A\cup B|=10$.

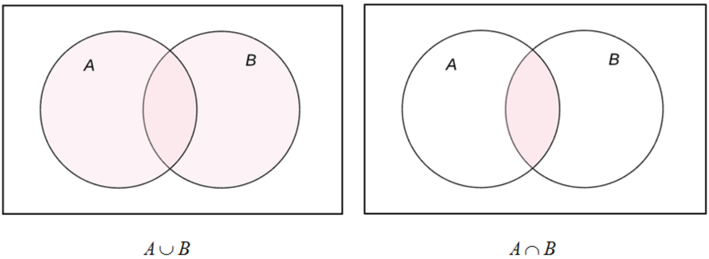
Já o conjunto $A\cap B$ consiste no conjunto formado pelos elementos comuns aos conjuntos A e B.

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

Temos ainda que $|A\cap B|=3$.

Os diagramas de Venn podem ser utilizados para ilustrar as operações binárias de união e intersecção de conjuntos. Na Figura 2.2 visualizamos a imagem mental dessas operações:

Figura 2.2 | União e intersecção de conjuntos

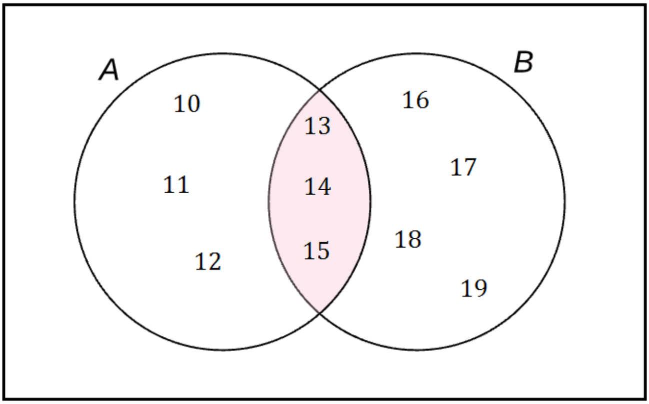


Fonte: elaborada pelo autor.

A região sombreada no primeiro diagrama representa $A\cup B$; já a região sombreada no segundo diagrama representa $A\cap B$.

A representação do Exemplo 1 por meio de diagramas de Venn pode ser observada na Figura 2.3:

Figura 2.3 | Exemplo 1



Fonte: elaborada pelo autor.

As operações $\cup$ e $\cap$ verificam diversas propriedades algébricas. Apresentaremos a seguir as leis básicas da álgebra de conjuntos usando a notação padrão da Teoria de Conjuntos (YAGLOM, 1998).

- Propriedades comutativas: $A \cup B = B \cup A$ e $A \cap B = B \cap A$.
- Propriedades associativas: $A \cup (B \cup C) = (A \cup B) \cup C$ e
$A \cap (B \cap C) = (A \cap B) \cap C$.
- Propriedades do conjunto vazio: $A \cup \varnothing = A$ e $A \cap \varnothing = \varnothing$.
- Propriedades distributivas: $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$ e
$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$.
- Leis de idempotência: $A \cup A = A$ e $A \cap A = A$.

Vamos demonstrar a propriedade associativa para a união. Considere os conjuntos A, B e C. Queremos provar que $A \cup (B \cup C) = (A \cup B) \cup C$. Temos que:

$A \cup (B \cup C) = \{ x | x \in A \text{ ou } x \in (B \cup C) \}$

Pela definição de união, segue que:

$A \cup (B \cup C) = \{ x | x \in A \text{ ou } x \in B \text{ ou } x \in C \}$

$A \cup (B \cup C) = \{ x | x \in (A \cup B) \text{ ou } x \in C \}$

Logo: $A \cup (B \cup C) = (A \cup B) \cup C$.

Constatamos, portanto, que as condições $A \cup (B \cup C)$ e $(A \cup B) \cup C$ são logicamente equivalentes.

Reflita

Utilizando as definições das operações união $(\cup)$ e intersecção $(\cap)$ de conjuntos, como você demonstraria a propriedade distributiva da união em relação à intersecção?

Utilize a demonstração da propriedade associativa para a união como referência e tente redigir a sua própria demonstração.

Além das operações união $(\cup)$ e intersecção $(\cap)$, outra operação que costuma aparecer com frequência ao estudarmos álgebra de conjuntos é a operação diferença de conjuntos. Sejam A e B dois conjuntos, de acordo com Scheinerman (2015), a diferença $A-B$ é o conjunto de todos os elementos de A que não estão em B, ou seja: $A-B=\{x \mid x \in A \text{ e } x \notin B\}$. Considere, por exemplo, os conjuntos $A=\{1,2,3,4,5\}$ e $B=\{4,5,6,7\}$. Para determinarmos a diferença $A-B$ temos de verificar quais elementos pertencem ao conjunto A, mas não pertencem ao conjunto B, ou seja, $A-B=\{1,2,3\}$. Analogamente, podemos definir o conjunto $B-A$, que consiste em todos os elementos pertencentes a B, mas que não pertencem ao conjunto A, ou seja, $B-A=\{6,7\}$.

Assimile

Outra operação utilizada na álgebra de conjuntos é a chamada diferença simétrica. Segundo Scheinerman (2015), a diferença simétrica de A e B pode ser denotada por $A\Delta B$. A diferença simétrica de A e B é o conjunto de todos os elementos que pertencem a A, mas não pertencem a B ou que pertencem a B, mas não pertencem a A. Essa diferença simétrica pode ser representada como $A\Delta B=(A-B)\cup(B-A)$.

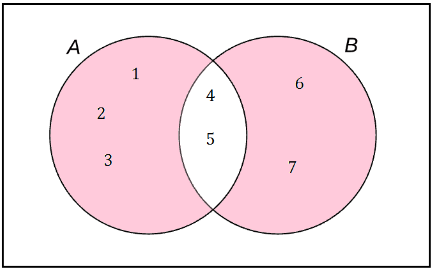
Considere, por exemplo, os conjuntos $A=\{1,2,3,4,5\}$ e $B=\{4,5,6,7\}$.

A diferença simétrica $A\Delta B$ ficaria definida como:

$A\Delta B=(A-B)\cup(B-A)=(1,2,3)\cup(6,7)=(1,2,3,6,7)$.

Utilizando diagramas de Venn, a diferença simétrica $A\Delta B$ ficaria representada como observado na Figura 2.4.

Figura 2.4 | Diferença simétrica $A\Delta B$



Fonte: elaborada pelo autor.

Até agora nos ocupamos em abordar as principais operações entre conjuntos. No entanto, em muitos problemas (especialmente quando se trabalha com conjuntos finitos, porém numerosos) realizamos essas operações levando em consideração o número de elementos (cardinalidade) de cada conjunto. Suponha que desejamos contar o número de objetos que goza de uma ou outra propriedade específica. Podemos abordar esse tipo de problema utilizando um caso especial de um método de contagem chamado inclusão-exclusão (SCHEINERMAN, 2015), que consiste na seguinte fórmula: $|A \cup B| = |A| + |B| - |A \cap B|$. Observe que para determinarmos o numero de elementos de $A \cup B$. devemos somar o número de elementos de A com o número de elementos de B e subtrair o número de elementos de $A \cap B$. A subtração se faz necessária para não incorrermos no risco de contabilizar o mesmo elemento duas (ou mais) vezes. Essa fórmula é recomendada quando o cálculo de $|A|$, $|B|$ e $|A \cap B|$ é mais fácil do que o cálculo de $|A \cup B|$.

Considere, por exemplo, que em uma pesquisa com profissionais do setor de Tecnologia da Informação de uma grande empresa responsável pela elaboração e manutenção de páginas (homepages) na Internet foi perguntado qual era o navegador de Internet utilizado durante o trabalho. Essa pesquisa apresentou os seguintes resultados: 280 profissionais alegaram utilizar o navegador A, 300 disseram utilizar o navegador B e 120 profissionais afirmaram utilizar ambos os navegadores. Sabendo que todos os funcionários do setor de Tecnologia da Informação dessa empresa responderam a essa pesquisa, como você determinaria o número desses profissionais?

Para responder a esse problema, podemos recorrer ao método de contagem chamado inclusão-exclusão. Vamos considerar como conjunto A o conjunto dos profissionais que utilizam o navegador A, e como conjunto B o conjunto dos profissionais que utilizam o navegador B. Temos que:

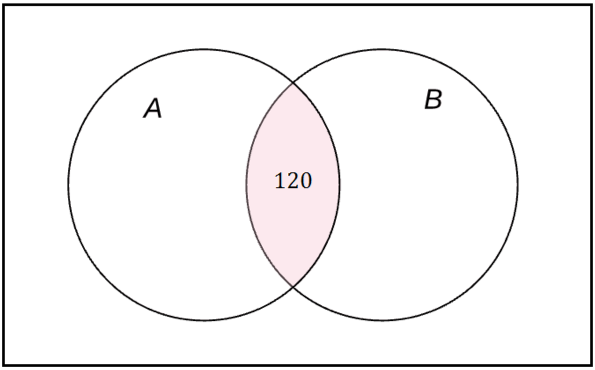
$$|A \cup B| = |A| + |B| - |A \cap B| = 280 + 300 - 120 = 460 .$$

Ou seja, essa empresa tem 460 profissionais trabalhando no setor de Tecnologia da Informação.

Dica

Poderíamos resolver essa situação problema por meio de um diagrama de Venn. Ao esboçarmos um diagrama, devemos sempre que possível iniciar pela intersecção dos conjuntos, como pode ser observado na Figura 2.5.

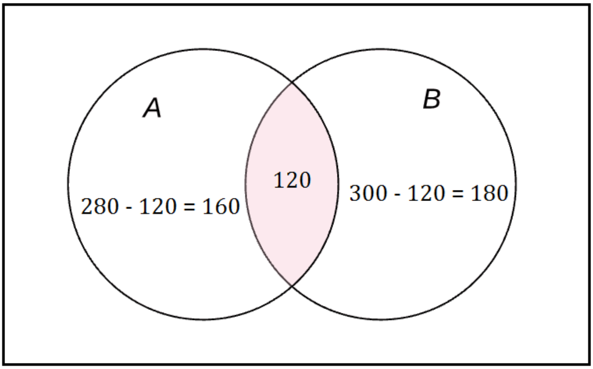
Figura 2.5 | Problema dos navegadores de internet (parte 1)



Fonte: elaborada pelo autor.

Em seguida, procuramos determinar o número de elementos pertencente a cada conjunto, lembrando de subtrair aqueles elementos que já estão representados na intersecção do diagrama, conforme representação da Figura 2.6.

Figura 2.6 | Problema dos navegadores de internet (parte 2)



Fonte: elaborada pelo autor.

Por fim, contabilizamos o número de elementos do conjunto de interesse.

É importante destacar que, ao se utilizar diagramas de Venn para resolver esse tipo de problema, os números registrados no diagrama corresponderão ao número de elementos (cardinalidade) de cada conjunto.

Nesse caso, temos que $|A \cup B| = 160 + 120 + 180 = 460$.

Há situações em que a intersecção entre os conjuntos é vazia. Nesse caso, dizemos que os conjuntos são disjuntos. Sejam $A=\{1,2,3\}$ e $B=\{7,8,9\}$. Como $A\cap B=\varnothing$, podemos afirmar que A e B são conjuntos disjuntos.

Há uma relação íntima entre as operações união $(\cup)$ e intersecção $(\cap)$ da álgebra de conjuntos e os conectivos lógicos largamente utilizados em diversas linguagens de programação OU (or) e E (and), respectivamente. Os conectivos lógicos OU e E são simbolizados por $\vee$ e $\wedge$.

Assim, podemos estabelecer as seguintes analogias:

I. $x\in(A\cup B)\Leftrightarrow(x\in A)\vee(x\in B)$.

II. $x\in(A\cap B)\Leftrightarrow(x\in A)\wedge(x\in B)$.

Em I, temos que o elemento x pertence à união de A e B se, e somente se, x pertence à A ou x pertence à B (podendo, inclusive, pertencer a ambos o conjuntos).

Em II, temos que o elemento x pertence à intersecção de A e B se, e somente se, x pertence à A e x pertence à B.

Nesta seção, nós conhecemos as principais operações de conjuntos (união, intersecção e diferença) e aprendemos a utilizá-las na resolução de problemas. Também pudemos perceber a importância da utilização dos diagramas de Venn para representação dessas situações. Esses conhecimentos, além de serem componentes importantes no desenvolvimento do raciocínio lógico, podem ser utilizados em inúmeras situações do nosso dia a dia, como aquelas que exigem a contagem do número de objetos que goza de uma ou outra propriedade específica.

Sem medo de errar

Você se lembra da situação-problema apresentada no início da seção? Vamos retomá-la?

Você está trabalhando em uma grande empresa do setor de Tecnologia da Informação (TI) e a sua equipe está responsável pelo desenvolvimento de um aplicativo para telefones móveis. Foi então que você se deparou com um novo problema: a equipe de desenvolvimento de software percebeu que alguns comandos de busca desse aplicativo direcionam a pesquisa do usuário para o Banco de Dados A, outros comandos direcionam a pesquisa do usuário para o Banco de Dados B, e ainda outros que realizam essa busca em ambos os Bancos de Dados (A e B). Sabendo que há 20 comandos que direcionam a busca do usuário para o Banco de Dados A, que no total há 60 comandos distintos e que há 12 comandos que direcionam a busca para os Bancos de Dados A e B, como você determinaria quantos desses 60 comandos realizam a busca no Banco de Dados B? Lembre-se de que você deverá apresentar a representação e resolução desse problema à equipe de desenvolvimento de software para que juntos decidam sobre a viabilidade de unificação desses Bancos de Dados.

Trata-se de um problema envolvendo operações de conjuntos. Uma vez que $|A| = 20$, $|A \cap B| = 12$ e $|A \cup B| = 60$, é preciso saber quanto é $|B|$.

Podemos recorrer ao método de contagem chamado inclusão-exclusão, que consiste em:

$$|A \cup B| = |A| + |B| - |A \cap B|.$$

Assim, temos que:

$$60 = 20 + |B| - 12$$

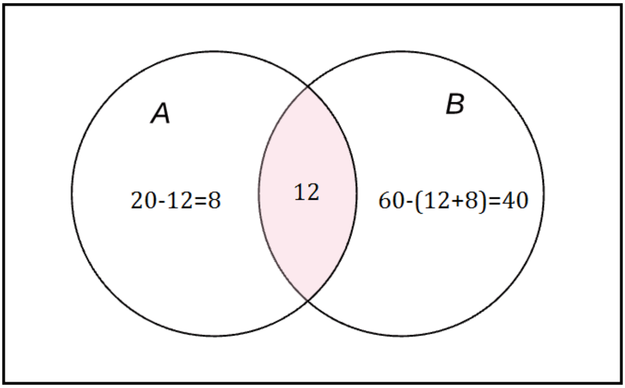
$$|B| = 60 - 20 + 12$$

$$|B| = 52$$

Portanto, dos 60 comandos existentes, há 52 comandos que realizam a busca no Banco de Dados B (incluindo 12 comandos que também realizam a busca no Banco de Dados A).

Para apresentar esse resultado à equipe de desenvolvimento de software, você pode recorrer a um diagrama de Venn, conforme ilustrado na Figura 2.7.

Figura 2.7 | Problema dos comandos de busca



Fonte: elaborada pelo autor.

O diagrama permite ainda aferir que 8 comandos efetuam a busca APENAS no Banco de Dados A enquanto que 40 comandos efetuam a busca APENAS no

Banco de Dados B. É devido ao poder de síntese que a utilização de diagramas é recomendada ao se trabalhar com problemas envolvendo conjuntos.

Faça valer a pena

1. Segundo Gersting (1995), a operação de ______ de A e B pode ser denotada como A_______B $= \{ x | x \in A \text{ e } x \in B \}$. Há situações em que a intersecção entre os conjuntos é vazia. Nesse caso, dizemos que os conjuntos são ______.

Assinale a alternativa que preenche corretamente as lacunas:

a. união; $\cup$; disjuntos.

b. união; $\cap$; conexos.

c. diferença; Δ ; disjuntos.

d. intersecção; $\cap$; conexos.

e. intersecção; $\cap$; disjuntos.

2. Uma pesquisa realizada com jovens da faixa etária de 18 a 25 anos questionou a preferência desses jovens em relação à utilização de redes sociais. A pergunta realizada a esses jovens foi: Quanto às redes sociais A (rede social de compartilhamento de fotos), B (rede social de compartilhamento de vídeos) e C (rede social profissional), qual(is) delas você utiliza? Os resultados dessa pesquisa podem ser observados no Quadro a seguir:

Quadro | Resultado de pesquisa

Rede Social	A	B	C	A e B	A e C	B e C	A, B e C	Nenhuma
Usuários	210	210	250	60	70	50	20	100

Fonte: elaborado pelo autor.

Considerando os resultados apresentados no quadro, pode-se afirmar que o número de jovens que responderam a essa pesquisa é igual a:

a. 300.

b. 510.

c. 610.

d. 670.

e. 970.

3. O setor de Tecnologia de Informação (TI) de uma empresa do ramo de seguros dispõe de computadores de mesa (desktops) e computadores portáteis (laptops) de dois fabricantes (A e B). Há 35 desktops, sendo 7 do fabricante A e 15 laptops do fabricante B. Sabe-se, ainda, que 18 computadores (entre desktops e laptops) são do fabricante A.

Considerando os dados apresentados, o número de computadores desktops fabricados por A é igual a:

a. 18.

b. 35.

c. 43.

d. 46.

e. 53.

Seção 3

Aplicações de teoria dos conjuntos

Diálogo aberto

Prezado aluno, nesta seção estudaremos mais algumas operações e relações entre conjuntos, dentre as quais citamos o complemento de um conjunto, a diferença simétrica e o produto cartesiano.

Imagine, por exemplo, que em vez de determinar o conjunto de usuários que avaliou positivamente um determinado aplicativo, você esteja interessado no conjunto de usuários que não forneceu uma avaliação positiva. Ou então que você precise determinar todas as maneiras pelas quais um elemento de um conjunto possa se relacionar com um elemento de outro conjunto. Essas duas situações servem para ilustrar os conceitos de complemento de conjunto e produto cartesiano que serão estudados nesta seção.

Como funcionário de uma grande empresa do setor de Tecnologia da Informação (TI), você se depara agora com uma situação inusitada: após trabalhar com o desenvolvimento de um aplicativo para telefones móveis, você e sua equipe foram contemplados com um curso de aprimoramento profissional na área de lógica computacional. E um dos desafios do curso consiste em representar uma relação arbitrária de três conjuntos, utilizando diferentes esquemas de representação, como as representações diagramáticas (com diagramas) e tabular (com tabelas). Você já compreende algumas relações entre conjuntos, mas sua missão agora é apresentar de maneira original uma relação arbitrária entre três conjuntos A, B e C, ou seja, você deverá elaborar um esquema que indique se um determinado elemento pertence ao conjunto A, B ou C, considerando todas as intersecções possíveis. Após elaborar sua solução, você deverá realizar uma apresentação para a sua equipe para demonstrar o esquema de representação escolhido e explicar sua interpretação.

Conseguir pensar nas relações entre conjuntos de maneira genérica é um componente importante daquilo que chamamos de raciocínio computacional. Além disso, saber representar tais relações de uma forma inteligível, que se faça compreender por outras pessoas de sua equipe de trabalho, é um exercício importante, pois muitas vezes não conseguimos expressar nossas ideias no papel.

Nesta seção você aprenderá a articular representações de conjuntos utilizando Diagramas de Venn, números binários e tabelas-verdade. Você está preparado? Mãos à obra e bons estudos!

Não pode faltar

Em Teoria dos Conjuntos, as operações com as quais trabalhamos mais frequentemente são as operações união $(\cup)$, intersecção $(\cap)$ e diferença $(-)$ de conjuntos. Uma nova relação que aprenderemos nesta seção é a operação denominada complemento ou complementar de um conjunto. O complemento de um conjunto é um conceito estreitamente relacionado com a operação de diferença de conjuntos. O Dicionário Houaiss da Língua Portuguesa (HOUAISS, 2009) define complemento como um elemento que se integra a um todo para completá-lo ou aperfeiçoá-lo. Relacionando essa definição com a Teoria de Conjuntos, podemos, de forma simplista, assumir que o complemento de um conjunto significa preencher o que falta.

Quando falamos em complementos de conjuntos, é necessário ter em mente um conjunto mais abrangente (U) sobre o qual estabeleceremos a relação de complementaridade.

Seja o conjunto A um subconjunto de U, ou seja, $A \subseteq U$. O complemento do conjunto A contido no conjunto U, denotado por $C_U A$, é a diferença entre os conjunto U e A ou seja, é o conjunto de todos os elementos que pertencem ao conjunto U e não pertencem ao conjunto A (PEREIRA; SODRÉ, 2006). Podemos representar simbolicamente essa relação como $C_U A = U - A = \{x | x \in U \text{ e } x \notin A\}$, que significa que o complemento do conjunto A em relação ao conjunto U equivale à diferença entre os conjuntos U e A que compreende todos os elementos que pertencem a U, mas não pertencem a A.

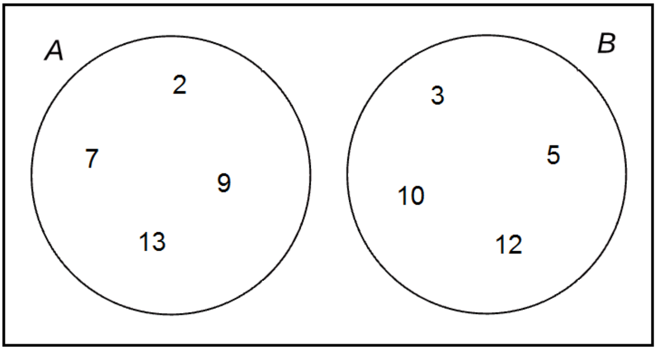
Exemplificando

Seja $A = \{1,2,3,4,5,6,7,8,9,10\}$ e $B = \{1,2,3,5,8,9\}$. O complemento de B em relação a A consiste no conjunto constituído por todos os elementos pertencentes a A que não pertencem a B. Temos, portanto: $C_A B = \{4,6,7,10\}$.

Em outras palavras, podemos dizer que o complemento de B em relação a A consiste no conjunto formado por elementos que pertencem exclusivamente a A, quando comparados com os elementos de B.

Segundo Ferreto (2019), há dois fatos que merecem atenção no estudo do complemento de um conjunto. O primeiro fato ao qual devemos ficar atentos é que para que um conjunto complementar a outro exista, há uma condição que deve ser respeitada: para que haja o cálculo do conjunto complementar de B em relação a A, B deve ser subconjunto de A. Então, ao nos depararmos com conjuntos disjuntos, ou com conjuntos que possuem apenas alguns elementos em comum, não é possível obter o conjunto complementar de um deles em relação ao outro. Considere, a título de ilustração, os conjuntos $A = \{2,7,9,13\}$ e $B = \{3,5,10,12\}$ (Figura 2.9).

Figura 2.9 | Conjuntos disjuntos



Fonte: elaborada pelo autor.

Esses dois conjuntos são conjuntos disjuntos, pois a intersecção entre eles é um conjunto vazio. Como B não é subconjunto de A, então não é possível determinar o complemento de B em relação a A. De modo análogo, também concluímos que não é possível determinar o complemento de A em relação a B.

Outro fato que também merece atenção diz respeito à maneira como o conjunto complementar é representado simbolicamente. É importante notar que $C_A B \neq C_B A$. Para determinarmos $C_A B$ fazemos $A - B$, enquanto que para determinarmos $C_B A$ fazemos $B - A$ (lembrando que nem sempre é possível determinar o complemento de um determinado conjunto).

Reflita

Em uma atividade sobre Álgebra dos Conjuntos foram apresentados dois conjuntos definidos como $M = \{10, 20, 30, 40, 50, 60, 70, 80\}$ e $N = \{30, 50, 70\}$. Em seguida, foi solicitada a determinação dos complementos $C_M N$ e $C_N M$.

O complemento de N em relação a M ficou determinado como $C_M N = \{10, 20, 40, 60, 80\}$, porém não foi possível determinar o complemento de M em relação a N. Reflita: por que não foi possível determinar $C_N M$?

Ainda em relação à representação simbólica (notação) do complementar de um conjunto não há, na literatura, um consenso em relação a essa representação. A representação do complementar do conjunto A em relação ao conjunto universo U pode se dar tanto como $C_U A$ quanto C_U^A .

Quando não há dúvida sobre o universo U em que estamos trabalhando, podemos simplesmente representar o complementar de A em relação a U como $\overline{A}$, ou A' , ou ainda, A^C .

Desse modo, quando não tivermos dúvida em relação ao conjunto universo U com o qual estamos trabalhando, representaremos o complementar de A em relação a U simplesmente utilizando a notação A^C . Por exemplo, considere $U = \{12,13,14,15,16,17,18\}$ e $A = \{12,13,14,15\}$. Neste caso, podemos representar o complementar de A em relação a U simplesmente como $A^C = \{16,17,18\}$.

Vimos, portanto, que o complemento do conjunto A em relação ao conjunto U equivale à diferença entre os conjuntos U e A, dado que $A \subseteq U$.

Cabe, nesse momento, chamar a atenção para outro tipo de diferença de conjuntos, denominado diferença simétrica. Segundo Scheinerman (2015), diferença simétrica de A e B é o conjunto de todos os elementos que pertencem a A, mas não pertencem a B, ou que pertencem a B, mas não pertencem a A. Essa diferença simétrica pode ser representada como $A \Delta B = (A - B) \cup (B - A)$. Seja $A = \{1,2,3,4,5,6,7\}$ e $B = \{5,6,7,8,9\}$. A diferença simétrica $A \Delta B$ ficaria definida como:

$$A \Delta B = (A - B) \cup (B - A) = (1,2,3,4) \cup (8,9) = (1,2,3,4,8,9).$$

Assimile

A diferença simétrica entre os conjuntos A e B também pode ser definida como o conjunto de todos os elementos que pertencem à união dos conjuntos A e B e não pertencem à intersecção dos conjuntos A e B (CABRAL, 2017). Vamos mostrar que as duas definições de diferença simétrica são equivalentes, considerando o exemplo anterior, com $A = \{1,2,3,4,5,6,7\}$ e $B = \{5,6,7,8,9\}$.

Queremos mostrar que

$$A \Delta B = (A - B) \cup (B - A) = (A \cup B) - (A \cap B)$$

Tem-se que

$(A - B) \cup (B - A) = (1,2,3,4) \cup (8,9) = (1,2,3,4,8,9)$ e que

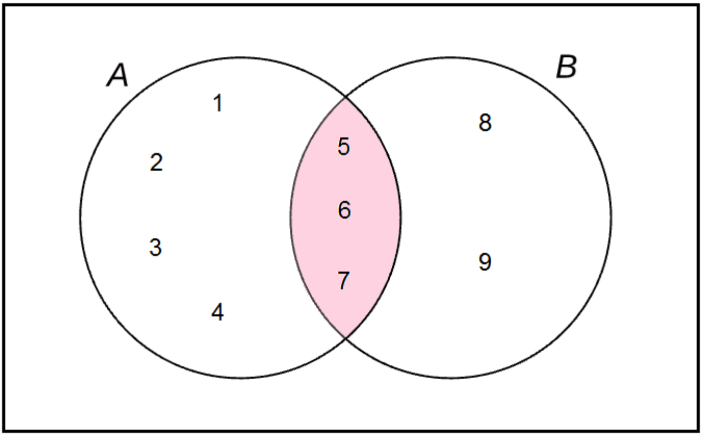
$(A \cup B) - (A \cap B) = (1,2,3,4,5,6,7,8,9) - (5,6,7) = (1,2,3,4,8,9) \cdot$

Logo,

$$A \Delta B = (A - B) \cup (B - A) = (A \cup B) - (A \cap B).$$

Essa propriedade pode ser mais bem visualizada por meio do Diagrama de Venn apresentado na Figura 2.10:

Figura 2.10 | Diferença simétrica de conjuntos

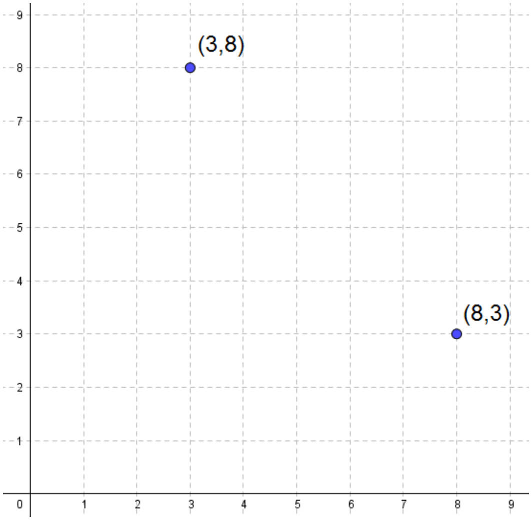


Fonte: elaborada pelo autor.

Outra aplicação que estudaremos no âmbito da Teoria de Conjuntos é a operação denominada produto cartesiano. Mas, para entender o que é o produto cartesiano, vamos primeiro definir o que é uma operação binária.

Um conjunto, por si só, não é objeto de muito interesse até que se faça algo com seus elementos (GERSTING, 1995). Podemos, por exemplo, realizar diversas operações aritméticas sobre os elementos de um determinado conjunto numérico. Tomemos, a título de ilustração, a operação aritmética subtração no conjunto dos números inteiros $\mathbb{Z}$. Dizemos que a subtração é uma operação binária em $\mathbb{Z}$. Ela é chamada de operação binária pois atua sobre dois números. Considerando a operação aritmética de subtração em $\mathbb{Z}$ temos que para quaisquer dois inteiros x e y, a operação $x - y$ resultará em uma resposta única, e essa resposta também será um número inteiro. Dizemos, portanto, que a operação de subtração foi realizada sobre um par ordenado de números. Denotamos um par ordenado por (x, y), em que x representa o primeiro componente do par ordenado e y, o segundo. A ordem é importante, pois muitas operações não são comutativas. Por exemplo, $8 - 3$ não gera o mesmo resultado que $3 - 8$. Atente para o fato de que os conjuntos $\{3,8\}$ e $\{8,3\}$ são iguais, mas os pares ordenados $(3,8)$ e $(8,3)$, não. Essa diferença fica muito bem ilustrada quando representamos os pares ordenados em um plano cartesiano, conforme podemos observar na Figura 2.11:

Figura 2.11 | Representação gráfica de pares ordenados



Fonte: elaborada pelo autor.

A representação de pares ordenados por meio de pontos no plano cartesiano facilita a compreensão a respeito da importância de se estabelecer uma ordem entre os elementos do par ordenado. O ponto $(3,8)$ é diferente do ponto $(8,3)$.

Assimile

Tomando como referência o conjunto $\mathbb{Z}$, considere que $(2x-y,x+y)=(7,-1)$. Vamos determinar os valores de x e y.

Como estamos trabalhando com igualdade entre pares ordenados, temos que o primeiro elemento do primeiro par ordenado é igual ao primeiro elemento do segundo par ordenado, ou seja, $2x-y=7$. Analogamente, temos que o segundo elemento do primeiro par ordenado é igual ao segundo elemento do segundo par ordenado, ou seja, $x+y=-1$. Assim, temos um sistema de equações lineares:

$$\begin{cases} 2x-y=7 \\ x+y=-1 \end{cases}$$

Resolvendo esse sistema, temos que $x=2$ e $y=-3$.

Agora que já aprendemos o que é uma operação binária entre conjuntos, voltemos à definição de produto de cartesiano. Sejam os conjuntos A e B. O produto cartesiano de A e B, denotado por $A \times B$, é o conjunto de todos os pares ordenados (listas de dois elementos) formados, tomando-se um elemento de A juntamente com um elemento de B de todas as maneiras possíveis. Ou seja, $A \times B = \{(a,b) | a \in A, b \in B\}$ (SCHEINERMAN, 2015).

Exemplificando

Considere os conjuntos $A = \{4,5,6\}$ e $B = \{6,7,8\}$. Vamos definir os produtos cartesianos $A \times B$ e $B \times A$.

$$A \times B = \{(4,6),(4,7),(4,8),(5,6),(5,7),(5,8),(6,6),(6,7),(6,8)\}$$

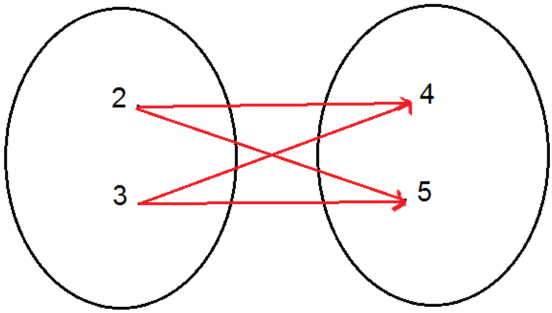
$$B \times A = \{(6,4),(6,5),(6,6),(7,4),(7,5),(7,6),(8,4),(8,5),(8,6)\}$$

Note que $A \times B \neq B \times A$. Isso acontece porque a operação produto cartesiano não é uma operação comutativa.

Cabe aqui fazer uma observação em relação à cardinalidade do conjunto obtido através da operação produto cartesiano. No exemplo apresentado temos que $|A| = 3$, $|B| = 3$ e conjunto obtido pela operação produto cartesiano $A \times B$ tem cardinalidade igual a 9, ou seja, $|A \times B| = 9$. Como o produto cartesiano é obtido tomando-se um elemento de A juntamente com um elemento de B de todas as maneiras possíveis, podemos aferir que, sendo A e B conjuntos finitos, $|A \times B| = |A| \times |B|$.

Os Diagramas de Venn podem ser adaptados para representar o produto cartesiano entre dois conjuntos. Considere, por exemplo, os conjuntos $A = \{2,3\}$ e $B = \{4,5\}$. O produto cartesiano $A \times B$ pode ser representado pelos diagramas da Figura 2.12:

Figura 2.12 | Produto cartesiano em Diagrama de Venn



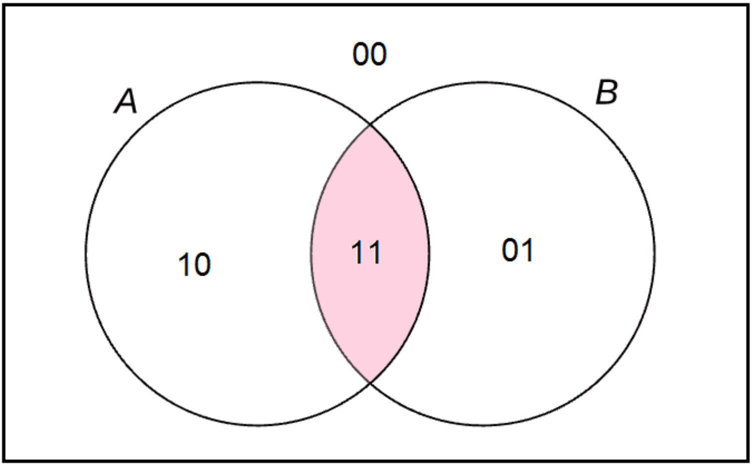
Fonte: elaborada pelo autor.

Temos, portanto, o produto cartesiano $A \times B = \{(2,4),(2,5),(3,4),(3,5)\}$ como o conjunto formado por todos os pares possíveis formados com os elementos de A e de B.

Já vimos, na Seção 2.2, 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. O Diagrama de Venn 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.

Vamos aqui utilizar um Diagrama de Venn para demonstrar uma relação arbitrária entre dois conjuntos A e B (Figura 2.13). No caso de dois conjuntos A e B, 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 compartimento 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.

Figura 2.13 | Representação diagramática de uma relação arbitrária entre dois conjuntos



Fonte: adaptada de Novaes (2014).

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 conjunto B. Já o número 11 (um, um) representa objetos que pertençam simultaneamente aos conjuntos A e B (intersecção de A e B). Por fim, o número 00 (zero, zero) representa objetos que não pertencem a nenhum dos conjuntos A e B.

Uma maneira equivalente de representar essa relação arbitrária é a utilização de tabelas-verdade, que serão apresentadas posteriormente. Apresentamos na Figura 2.14 a tabela-verdade para a relação arbitrária entre dois conjuntos:

Figura 2.14 | Tabela-verdade de uma relação arbitrária entre dois conjuntos

	$x \in A$	$x \in B$	$x \in A \cap B$
00	F	F	F
01	F	V	F
10	V	F	F
11	V	V	V

Fonte: elaborada pelo autor.

Na primeira coluna da tabela indicamos os $2^2 = 4$ compartimentos que compõem 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 $A \cap B$.

Nesta seção pudemos aprender um pouco mais sobre as operações entre conjuntos e suas aplicações. Além de nos debruçarmos sobre as operações de complementar de um conjunto, diferença simétrica e produto cartesiano, também pudemos aplicar a utilização de Diagramas de Venn, tanto na representação de tais operações, quanto na representação de relações arbitrárias entre conjuntos. Compreender a natureza dessas operações é componente importante no desenvolvimento do raciocínio computacional.

Sem medo de errar

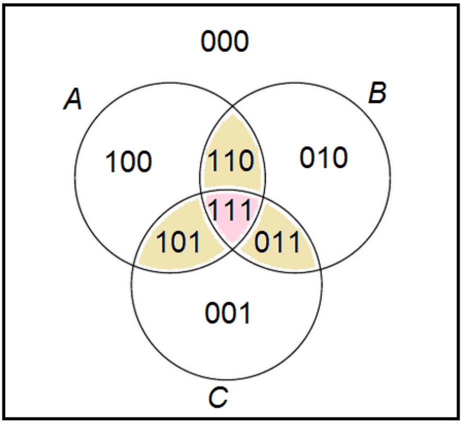
Você se lembra do desafio apresentado no início desta seção?

Você está fazendo um curso de aprimoramento profissional sobre lógica computacional e um dos desafios do curso consiste em representar uma relação arbitrária de três conjuntos, A, B e C, utilizando diferentes esquemas de representação, ou seja, você deverá elaborar um esquema que indique se um determinado elemento pertence ao conjunto A, B ou C, considerando todas as intersecções possíveis e, após elaborar sua solução, você deverá realizar uma apresentação para a sua equipe para demonstrar o esquema de representação escolhido e explicar sua interpretação.

Vimos que o Diagrama de Venn, além ser um esquema para ajudar o raciocínio, também consiste em 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 (NOVAES, 2014).

Portanto, vamos utilizar um Diagrama de Venn para demonstrar uma relação arbitrária entre três conjuntos A, B e C. Utilizaremos uma numeração binária, composta apenas pelos algarismos 0 e 1, em que o primeiro algarismo é 0 ou 1, conforme um objeto desse compartimento pertença ou não ao conjunto A, o segundo algarismo é 0 ou 1, conforme um objeto desse compartimento pertença ou não ao conjunto B e o terceiro algarismo é 0 ou 1, conforme um objeto desse compartimento pertença ou não ao conjunto C (Figura 2.15).

Figura 2.15 | Representação diagramática de uma relação arbitrária entre três conjuntos



Fonte: elaborada pelo autor.

A numeração binária mostra que $2^3 = 8$ compartimentos esgotam todas as possibilidades lógicas para um objeto do universo. O número 100 (lê-se: um, zero, zero) representa objetos que pertençam exclusivamente ao conjunto A, o número 010 (zero, um, zero) representa objetos que pertençam exclusivamente ao conjunto B e o número 001 (zero, zero, um) representa objetos que pertençam exclusivamente ao conjunto C. Já o número 000 (zero, zero, zero) representa objetos que não pertencem a nenhum dos conjuntos A, B e C. Temos ainda as intersecções:

$$110 = A \cap B \cap C^C$$

$$101 = A \cap B^C \cap C$$

$$001 = A^C \cap B \cap C$$

$$111 = A \cap B \cap C$$

Vejamos na Figura 2.16 como ficaria a tabela-verdade para representação dessa relação:

Figura 2.16 | Tabela-verdade de uma relação arbitrária entre dois conjuntos

	$x \in A$	$x \in B$	$x \in C$	$x \in B \cap C$	$x \in A \cup (B \cap C)$
000	F	F	F	F	F
001	F	F	V	F	F
010	F	V	F	F	F
011	F	V	V	V	V
100	V	F	F	F	V
101	V	F	V	F	V
110	V	V	F	F	V
111	V	V	V	V	V

Fonte: elaborada pelo autor.

Na primeira coluna da tabela indicamos os $2^3 = 8$ compartimentos que compõem o Diagrama de Venn para três 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. Na quarta coluna anotamos V (verdadeiro) ou F (falso) conforme o objeto representado pelo seu respectivo número binário pertença, ou não, ao conjunto C. Na quinta coluna anotamos V (verdadeiro) ou F (falso) conforme o objeto representado pelo seu respectivo número binário pertença, ou não, à intersecção dos conjuntos B e C. Por fim, na última coluna, anotamos V (verdadeiro) ou F (falso) conforme o objeto representado pelo seu respectivo número binário pertença, ou não, ao conjunto $A \cup (B \cap C)$. As proposições apresentadas na primeira linha da tabela foram escolhidas a título de ilustração.

Assim, conseguimos representar uma relação arbitrária entre três conjuntos A, B e C utilizando representações com diagramas e tabelas.

Faça valer a pena

1. Sejam agora A e B dois conjuntos quaisquer. Chama-se produto cartesiano de A e B, e designa-se pelo símbolo $A \times B$, o conjunto de todos os pares ordenados (a,b) tais que $a \in A$ e $b \in B$. Simbolicamente:

$A \times B = \{(x, y) : x \in A \wedge y \in B\}$.

Se, em particular, é $A = B$, o produto cartesiano $A \times B$ (ou $A \times A$) chama-se quadrado cartesiano de A e designa-se usualmente por A^2 (FERREIRA, 2001).

Sendo $A = \{1,2,3\}$, o quadrado cartesiano de A, representado por A^2, corresponde ao conjunto:

a. $\{2,4,6\}$.

b. $\{1,4,9\}$.

c. $\{(1,1),(2,2),(3,3)\}$.

d. $\{(1,2),(1,3),(2,1),(2,3),(3,1),(3,2)\}$.

e. $\{(1,1),(1,2),(1,3),(2,1),(2,2),(2,3),(3,1),(3,2),(3,3)\}$.

2. A representação simbólica do complementar do conjunto A em relação ao conjunto U é $C_U A = U - A = \{x | x \in U \text{ e } x \notin A\}$, que significa o complemento do conjunto A em relação ao conjunto U. Em relação ao conceito de complemento de um conjunto, complete as lacunas da sentença a seguir:

Quando os matemáticos falam de complementos de conjuntos, eles em geral têm em mente um conjunto mais _____. Por exemplo: no decorrer de uma prova ou discussão, se A é um conjunto que contém apenas números inteiros, A^C corresponde ao conjunto de todos os inteiros que _____ em A. Se U é o conjunto de todos os objetos em consideração e $A \subseteq U$, então o _____ de A é o conjunto de todos os objetos de U que não estão em A (SCHEINERMAN, 2015).

Assinale a alternativa que preenche corretamente as lacunas:

a. restrito; estão; complemento.

b. abrangente; estão; simétrico.

c. restrito; não estão; complemento.

d. abrangente; não estão; complemento.

e. abrangente; não estão; simétrico.

3. Assim como os números poder ser somados ou multiplicados e os valores verdade podem ser combinados com $\wedge$ e $\vee$, há várias operações que podemos fazer sobre conjuntos.

Considerando as operações sobre conjuntos estudadas nesta seção, julgue as afirmativas a seguir em (V) Verdadeiras ou (F) Falsas.

( ) A diferença simétrica $A \Delta B$ é o conjunto de todos os elementos que estão em A, ou em B, mas não em ambos.

( ) O produto cartesiano $A \times B$ é o conjunto de todos os pares ordenados da forma (a,b) em que $a \in A$ e $b \in B$.

( ) O produto cartesiano $A \times B$ é uma operação comutativa.

Assinale a alternativa que apresenta a sequência correta:

a. V – V – V.

b. V – V – F.

c. V – F – V.

d. F – V – V.

e. V – F – F.

Referências

BARBOSA, M. A. **Introdução à lógica matemática para acadêmicos**. Curitiba: InterSaberes, 2017. Disponível em: https://plataforma.bvirtual.com.br/Leitor/Publicacao/49489/pdf. Acesso em: 2 dez. 2019.

BENZECRY, V. S. J.; RANGEL, K. A. **Como desenvolver o raciocínio lógico:** soluções criativas na teoria dos conjuntos. Rio de Janeiro: LTC, 2008. Disponível em: https://integrada.minhabiblioteca.com.br/#/books/978-85-216-1991-8/cfi/4!/4/4@0.00:56.1. Acesso em: 9 dez. 2019.

BÖHM, D.; SCHREIBER, J.; FROZZA, R. DVenn – um software de auxílio ao aprendizado de lógica nos cursos de graduação em computação. In: COMPUTER ON THE BEACH, 2012, Florianópolis. **Anais...**. Florianópolis, UNIVALI; CTTMar, 2012. p. 21-30. Disponível em: https://pdfs.semanticscholar.org/fa52/b39a8d2f620a04812ce1e503a2417fae5255.pdf?_ga=2.156141811.914687476.1578595524-288639821.1578595524. Acesso em: 9 jan. 2020.

CABRAL, R. M. P. **Matemática Discreta.** Fortaleza: EdUECE, 2017.

CERIOLI, M. R. **Diagramas de Venn**. 2004. Disponível em: https://www.cos.ufrj.br/~cerioli/cursos/matfin-MAA/Venn.html. Acesso em: 19 dez. 2019.

CLUBES DE MATEMÁTICA DA OBMEP. **Diagrama de Venn:** problemas de raciocínio lógico. Rio de Janeiro, 2019. Disponível em: http://clubes.obmep.org.br/blog/diagrama-de-venn-problemas-de-raciocinio-logico/. Acesso em: 19 dez. 2019.

FERREIRA, J. C. **Elementos de lógica matemática e teoria dos conjuntos**. Lisboa: Departamento de Matemática do Instituto Superior Técnico, 2001.

FERRETO, D. Complementar de um conjunto. **Professor Ferreto**, 3 jul., 2019. Disponível em: https://blog.professorferretto.com.br/complementar-de-um-conjunto/. Acesso em: 8 jan. 2020.

GERSTING, J. L. **Fundamentos matemáticos para a Ciência da Computação**. Rio de Janeiro: LTC, 1995.

GERSTING, J. L. **Fundamentos matemáticos para a ciência da computação:** matemática discreta e suas aplicações. 7. ed. Rio de Janeiro: LTC, 2017. Disponível em: https://integrada.minhabiblioteca.com.br/#/books/9788521633303/cfi/6/10!/4/50@0:3.16. Acesso em: 9 jan. 2020.

HOUAISS, A. **Dicionário eletrônico Houaiss da Língua Portuguesa 3.0**. [*S. l.*]: Objetiva, 2009. CD-ROM.

LUCIDCHART. **Visualize, entenda, resolva.** 2019. Disponível em: https://www.lucidchart.com/pages/pt/. Acesso em: 18 dez. 2019.

NOVAES, G. P. Reflexões sobre o ensino de conjuntos Diagramas de Venn. **Revista do Professor de Matemática**, São Paulo, v. 32, n. 84, p. 44-47, maio/ago. 2014.

PEREIRA, R. M. M.; SODRÉ, U. **Teoria dos conjuntos**. Londrina, 17 nov. 2006. Disponível em: http://www.uel.br/projetos/matessencial/medio/conjuntos/conjunto.htm#conj10. Acesso em: 8 jan. 2020.

SCHEINERMAN, E. R. **Matemática discreta:** uma introdução. São Paulo: Cengage Learning, 2015.

SOUZA, J. A. L. **Lógica matemática**. São Paulo: Pearson Education do Brasil, 2016. Disponível em: https://plataforma.bvirtual.com.br/Leitor/Publicacao/150814/pdf. Acesso em: 2 dez. 2019.

YAGLOM, I. M. **Álgebra booleana**. São Paulo: Atual, 1998.

Unidade 3

Vanessa Cadan Scheffer

Fundamentos da Lógica

Convite ao estudo

Caro estudante, bem-vindo a mais uma unidade do seu livro de lógica computacional. Nesta unidade daremos um passo importante para a compreensão da lógica no mundo computacional, principalmente no que se refere à lógica usada na construção de algoritmos. Esse conhecimento será construído através do estudo e entendimento da lógica proposicional e seus conectivos, que permitem criar regras e valorar seus resultados como verdadeiro ou falso.

Logo nos primeiros anos de vida, aprendemos a falar, mesmo que criando pequenas frases, que muitas vezes somente os pais entendem. Com o desenvolvimento da linguagem, nossa comunicação vai ficando mais fluida e complexa, pois vamos aprendendo "regras" que permitem fazer essa construção. Essas regras vão além do vocabulário e sintaxe de uma língua como, por exemplo, a língua portuguesa, pois elas envolvem a construção do raciocínio lógico. Utilizando esse raciocínio tomamos decisões ao longo de nossa vida e, quanto mais tivermos ciência dessas regras lógicas, melhor serão nossas análises e, consequentemente, nossas decisões.

Você está participando de um processo seletivo para desenvolvedor trainee em uma grande empresa de tecnologia. Você já passou a primeira fase, composta por entrevistas com o gestor e o setor de recursos humanos, agora chegou a hora de mostrar que você manda bem na lógica e que tem capacidade para se tornar um grande desenvolvedor. Essa etapa do processo consiste em três testes, no primeiro você deverá criar proposições simples e compostas para resolver um problema com as formas geométricas. No segundo desafio, você deverá criar estruturas condicionais usando os operadores lógicos para resolver um problema com fórmulas. No último desafio, você deverá usar os recursos da lógica para demonstrar a veracidade de um argumento.

Para concluir com êxito essa fase do processo seletivo e garantir sua vaga na empresa, você aprenderá na primeira seção da unidade sobre a lógica proposicional e os conectores de conjunção, disjunção e negação. Na segunda seção iremos explorar novas proposições compostas aprendendo outros conectores. Na terceira seção, veremos como as regras lógicas nos permitem criar novas conclusões por meio de métodos dedutivos e inferenciais.

Vamos juntos dar esse importante passo rumo à formação de programadores!

Bons estudos!

Seção 1

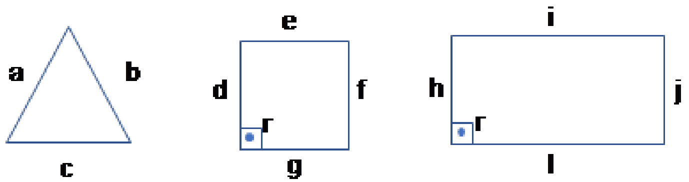
Introdução à lógica proposicional

Diálogo aberto

Caro estudante, iniciamos agora nossa introdução à lógica proposicional. Esse conhecimento o capacitará a construir condições lógicas embasadas em regras e a valorar o resultado dessa condição. Sem dúvida, esse conhecimento é imprescindível na construção de algoritmos computacionais.

Como candidato a uma vaga de desenvolvedor trainee em uma grande empresa de tecnologia, após passar pelas entrevistas com seu futuro gestor e com o setor de recursos humanos, chegou a hora de vencer mais uma etapa: o teste de lógica. A empresa faz questão desse teste, pois sabe que um bom desenvolvedor deve mandar bem na lógica. Nesse primeiro teste, a empresa lhe forneceu uma folha com três figuras geométricas, conforme ilustra a Figura 3.1.

Figura 3.1 | Figuras geométricas



Fonte: elaborada pela autora.

Usando as letras que representam os lados das figuras geométricas na Figura 3.1, sua missão é construir proposições, simples e compostas, que representem as regras necessárias para a construção das três figuras. Nessa mesma folha, além das figuras também vieram algumas dicas sobre a construção dos elementos geométricos apresentados:

- Para que um triângulo possa ser construído é necessário que a soma de dois lados seja sempre maior que o outro lado.
- Um quadrado é composto por quatro lados com medidas iguais e quatro ângulos de noventa graus.
- Um retângulo é composto por quatro lados com medidas paralelas iguais e quatro ângulos de noventa graus.

Agora é com você! Como você poderá usar as dicas e transformá-las em proposições simples e/ou compostas? Quantas proposições serão necessárias para cada figura geométrica? Quais operadores lógicos você terá que usar para construir as regras?

Para vencer esse desafio, nessa seção iremos aprender sobre a lógica proposicional, o que são as proposições simples e como criar as proposições compostas, utilizando os conectivos lógicos de conjunção e disjunção.

Vamos em frente garantir essa vaga de emprego!

Não pode faltar

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 Formal desenvolvida por Aristóteles entre 300 e 400 anos antes de Cristo (MACHADO; CUNHA, 2008). 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, p. 15).

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, p. 16). Observe o argumento de uma professora sobre o desempenho de um certo aluno: "É lógico que Pedro 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, conforme ilustra o Quadro 3.1.

Quadro 3.1 | Premissas e conclusões

Premissas (razões)	1. Pedro é inteligente. 2. Pedro estuda muito. 3. Todos os alunos inteligentes e estudiosos são aprovados.
Conclusão	Pedro será aprovado

Fonte: adaptado de Machado e Cunha (2008, p. 16).

Veja no Quadro 3.1 que separamos a frase em 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. Além disso, imagina que queiramos criar um algoritmo para classificar se um aluno foi aprovado ou reprovado, essas premissas precisam ser programadas em forma de regras, as quais aprenderemos em breve.

Para ser um 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.

5. ?

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).

Assimile

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.

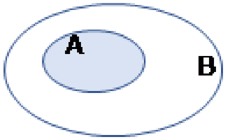
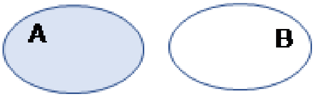
Nem sempre classificar se uma sentença é ou não uma proposição é uma tarefa trivial. Para que seja uma proposição, a sentença declarativa não pode deixar dúvidas quanto ao resultado. Por exemplo, a sentença "Está chovendo agora" não pode ser classificada como V ou F, pois deixa dúvida (por exemplo, pode estar chovendo em um ponto da cidade e em outro não). Para que essa frase se torne uma proposição ela precisa de um contexto, por exemplo, "Está chovendo agora na minha rua", ou seja, o locutor da frase especificou o local, então agora é possível valorar se o que ele disse é verdadeiro ou falso.

Reflita

As proposições (premissas) são usadas para sustentar uma conclusão em um argumento. Sabendo que essa conclusão também é uma proposição, podemos afirmar que uma conclusão verdadeira é sempre composta por premissas que são suficientes para validá-la?

Sabendo que uma proposição é uma sentença declarativa que pode ser classificada como V ou F, podemos encontrá-la nas mais diversas formas. Observe o Quadro 3.2. Nele temos diagramas de Euler representando conjuntos e sentenças que podem ser classificadas como V ou F, baseado no diagrama, portanto, são proposições.

Quadro 3.2 | Proposições e conjuntos

Diagrama de Euler	Proposição
	Todo A é B.
	Nenhum A é B.

Fonte: adaptado de Souza (2016, p. 35).

Segundo Gersting (2017), podemos utilizar letras maiúsculas do alfabeto para representar as proposições, portanto a partir de agora adotaremos essa notação para representar as proposições. Por exemplo:

A: Quinze é menor que vinte.

B: Todos os italianos são europeus.

Segundo Bispo e Castanheira (2011, p. 4), toda proposição deve seguir três princípios básicos:

- **Princípio da Identidade**: “Toda proposição é idêntica a si mesma”. Ou seja, sendo P uma proposição: **P é P**.
- **Princípio da Não Contradição**: “Uma proposição não pode ser verdadeira e falsa ao mesmo tempo”. Sendo P uma proposição tem-se: não **(P e não P).**
- **Princípio do Terceiro excluído**: “Toda proposição ou é verdadeira ou é falsa, não existindo um terceiro valor que ela possa assumir”. Sendo P uma proposição tem-se: **P ou não P.**

As proposições podem ser classificadas como simples ou compostas. A proposição será simples quando existir uma única afirmação na frase. Por outro lado, uma proposição é composta quando for constituída de, pelo menos, duas proposições simples “ligadas” por um conectivo lógico, também chamado de conector lógico, conectivo proposicional ou operação lógica. (BISPO; CASTANHEIRA, 2011).

Exemplificando

Observe as proposições A, B, C a seguir:

A: 11 é um número ímpar.

B: 11 é um número primo.

C: 11 é um número ímpar e primo.

As proposições A e B são compostas por uma única verdade.

A: verdadeira

B: verdadeira

Já a proposição C é composta por duas proposições simples (nesse caso, pelas proposições A e B) que são ligadas pela palavra "e". A valoração da proposição C depende da valoração independente de A e de B, aplicada ao operador lógico que está unindo as duas proposições simples. Veremos em breve como fazer essa valoração.

Antes de avançarmos com a valoração das proposições compostas, vejamos mais dois exemplos, extraídos de Bispo e Castanheira (2011, p. 4).

Exemplo a: Considere a frase: "Os suíços fabricam os melhores relógios e os franceses, o melhor vinho". Se extrairmos as proposições simples das frases teremos:

P: Os suíços fabricam os melhores relógios.

S: Os franceses fabricam o melhor vinho.

Podemos reescrever a frase, utilizando uma notação simbólica, então o resultado será: P **e** S. Veja que temos a palavra "e" ligando as duas proposições simples.

Exemplo b: Considere a frase: "Se eu prestar atenção na aula, então tirarei boa nota na prova". Novamente, se extrairmos as proposições simples das frases teremos:

A: Eu presto atenção na aula.

R: Eu tiro boa nota na prova.

Veja que ao extrair 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. Para compreendermos melhor, considere as proposições a seguir:

A: Quatro é um número par.

B: Três é um número ímpar.

C: Cinco é maior que dez.

P: Quatro é um número par e três é um número ímpar.

R: Quatro é um número par e cinco é maior que dez.

Vamos começar valorando as proposições simples (A, B, C):

A: verdadeira.

B: verdadeira.

C: falsa.

Para facilitar nossa compreensão, vamos reescrever as proposições P, R utilizando notação simbólica, portanto temos:

P: $A \wedge B$

R: $A \wedge C$

Na proposição P temos que as proposições simples A, B são verdadeiras, portanto P = V e V, o que resulta em verdadeiro, ou seja, a proposição P pode ser valorada como verdadeira. Já na proposição R, temos que A é verdadeiro, mas C é falso, portanto R = V e F, o que resulta em falso, ou seja, a proposição R deve ser valorada como falsa, já que para essa operação lógica ambas proposições precisam ser verdadeiras para o resultado também ser verdadeiro.

Além da palavra "e" outras podem ser usadas para representar a conjunção entre duas proposições: ***mas***, ***todavia***, ***contudo***, ***no entanto***, ***visto que***, ***enquanto***, ***além disso***, ***embora***. (BISPO; CASTANHEIRA, 2011).

Veja alguns exemplos:

A: João foi ao cinema.

B: Maria foi ao shopping.

C: João foi ao cinema, mas Maria foi ao shopping.

D: João foi ao cinema, enquanto Maria foi ao shopping.

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**. Veremos nesta seção a forma inclusiva do operador.

O operador lógico de disjunção usado na forma inclusiva 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. Para compreendermos melhor, considere as proposições a seguir:

A: Quatro é um número par.

B: Três é um número ímpar.

C: Cinco é maior que dez.

D: Sete é um número par.

P: Quatro é um número par ou três é um número ímpar.

R: Quatro é um número par ou cinco é maior que dez.

S: Cinco é maior que dez ou sete é um número par.

Vamos começar valorando as proposições simples (A, B, C, D):

A: verdadeira.

B: verdadeira.

C: falsa.

D: falsa.

Para facilitar nossa compreensão, vamos reescrever as proposições P, R, S utilizando notação simbólica, portanto temos:

P: $A \vee B$

R: $A \vee C$

S: $C \vee D$

Na proposição P temos que as proposições simples A, B são verdadeiros, portanto P = V ou V, o que resulta em verdadeiro, ou seja, a proposição P pode ser valorada como verdadeira. Na proposição R, temos que A é verdadeiro, mas C é falso, portanto R = V ou F, como se trata da disjunção inclusiva, o resultado será verdadeiro, pois para ser falso ambas proposições simples têm que ser falsas. Já na proposição S, temos que C, D são proposições simples falsas, portanto S = F ou F, o que resultado em falso.

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 binária). 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 $\sim$, $\leftarrow$, ‘ . 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: $\sim A$, $\leftarrow B$, 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. Vejamos um exemplo, considere a proposição:

A: Luís gosta de viajar.

A negação de A ($\sim A$) pode ser lida como:

$\sim A$: Luís não gosta de viajar.

Ou ainda como:

$\sim A$: É falso que Luís gosta de viajar.

Ou ainda

$\sim A$: Não é verdade que Luís gosta de viajar.

A negação pode ser aplicada ao resultado de uma outra operação, como por exemplo, $\sim (A \wedge B)$. Nesse caso o resultado da conjunção será invertido.

No universo dos algoritmos computacionais, esses três conectores são amplamente utilizados para construir estruturas de decisões. Por exemplo, imagine que você esteja trabalhando em um sistema web para uma universidade. Em uma das páginas do sistema, deverá ser implementada a opção para o usuário escolher o curso, o semestre e a idade dos alunos. Nesse cenário vamos considerar as seguintes proposições:

A: Todos os alunos são do curso de engenharia.

B: Todos os alunos são do segundo semestre.

C: Todos os alunos possuem idade superior a 30 anos.

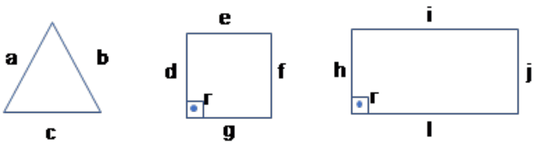
Agora, suponhamos que o usuário do sistema queira listar alunos que não são dos cursos de engenharia, alunos que estão no segundo semestre, alunos que possuem idade superior a 30 anos. Como essa regra (proposição) deve ser construída? Qual combinação de conectores deve ser usada para produzir o resultado desejado? Considerando as proposições dadas, a lógica a ser criada deve ser: $\sim A \wedge B \wedge C$.

Como você pode notar, sem a Lógica Formal, jamais seríamos capazes de construir nossos algoritmos computacionais. Portanto, continue estudando esse importante tema!

Sem medo de errar

Chegou a hora de pensarmos como resolver seu desafio de lógica para a vaga de desenvolvedor trainee de uma grande empresa de tecnologia. Você recebeu uma folha contendo três figuras geométricas e algumas dicas, se lembra? As figuras geométricas estão representadas na Figura 3.2.

Figura 3.2 | Figuras geométricas para o teste



Fonte: elaborada pela autora.

As figuras geométricas são: um triângulo de lados a, b, c, um quadrado de lados d, e, f, g e ângulo r e um retângulo de lados h, i, j, l e ângulo r. Além da figura, você recebeu as seguintes dicas:

1. Para que um triângulo possa ser construído é necessário que a soma de dois lados seja sempre maior que o outro lado.

2. Um quadrado é composto por quatro lados com medidas iguais e quatro ângulos de noventa graus.
3. Um retângulo é composto por quatro lados com medidas paralelas iguais e quatro ângulos de noventa graus.

Com essas informações você deve construir proposições simples e compostas que representem as regras necessárias para a construção das três figuras.

Pois bem, o primeiro passo é construir as proposições simples. Vamos começar pelo triângulo.

A: A soma das medidas do lado a com o lado b do triângulo abc é menor que a medida do lado c.

B: A soma das medidas do lado b com o lado c do triângulo abc é menor que a medida do lado a.

C: A soma das medidas do lado a com o lado c do triângulo abc é menor que a medida do lado b.

Agora que temos as proposições simples, vamos usar a dica 1 para construir uma expressão lógica que traduza essa regra. Como as três proposições precisam ser verdadeiras para que seja possível construir um triângulo, teremos como resultado a expressão: $A \wedge B \wedge C$. Ou seja, foi necessário usar a conjunção para construir a proposição composta que representa a regra.

Agora vamos construir as proposições simples para criar a regra para a construção do quadrado.

P: A medida do lado d é igual à do lado e.

Q: A medida do lado f é igual à do lado e.

S: A medida do lado g é igual à do lado f.

T: A medida do ângulo r é diferente de noventa graus.

Com a criação das proposições simples, agora podemos criar a proposição composta que representa a regra: $P \wedge Q \wedge S \wedge \sim T$. Veja, que na maneira como construímos a proposição simples T, foi necessário usar a negação para construir a regra correta. É importante você ficar atento ao aspecto sintático da expressão.

Por fim, vamos construir as proposições simples para o retângulo.

X: A medida do lado h é igual à do lado j.

Z: A medida do lado i é igual à do lado l.

W: A medida do ângulo r é igual a noventa graus.

Agora basta usar o(s) conector(es) corretos para criar a proposição composta que representa a regra: $X \wedge Z \wedge W$.

Veja que, a partir da utilização de proposições simples e conectivos lógicos, foi possível construir formas e regras que podem ser implementadas computacionalmente.

Faça valer a pena

1. "Um argumento é uma construção cujos elementos são proposições. Em um argumento sempre existe uma conclusão, que é sustentada por uma ou mais premissas. Argumentar significa garantir a verdade da conclusão tendo por base a verdade das premissas" (MACHADO; CUNHA, 2008, p. 22).

Dada a frase: "Como a gasolina é extraída do petróleo, que é importado, e todos os produtos importados são caros, a gasolina é cara".

É correto afirmar que:

a. A frase não pode ser classificada como um argumento.

b. A frase é um argumento composto por três premissas e uma conclusão.

c. A frase é um argumento composto por duas premissas e uma conclusão.

d. A frase é um argumento composto por uma premissa e uma conclusão.

e. A frase é um argumento composto por uma premissa e duas conclusões.

2. O Cálculo Proposicional é a parte da Lógica Matemática que estuda a validade de argumentos apresentados em uma linguagem própria, a linguagem proposicional. Nessa linguagem é possível distinguir dois aspectos: o sintático e o semântico. O sintático estabelece símbolos, regras de formação e regras de dedução de validade. O aspecto semântico consiste na valoração das fórmulas com atribuição da propriedade verdadeiro ou falso (BISPO; CASTANHEIRA, 2011, p. 2).

Avalie as sentenças a seguir.

I. Hoje está fazendo sol.

II. Ela é muito talentosa.

III. Existem outras formas de vida em outros planetas no universo.

IV. Maria conseguiu concluir com êxito o segundo semestre da faculdade.

É correto o que se afirma em:

a. I apenas.

b. I e II, apenas.

c. II e III, apenas.

d. III e IV, apenas.

e. I, II, III, IV.

3. "Na linguagem comum, usam-se palavras explícitas ou não para interligar frases dotadas de algum sentido. Tais palavras são substituídas, na Lógica Matemática, por símbolos denominados conectivos lógicos" (BISPO; CASTANHEIRA, 2011, p. 5).

Analise as afirmativas a seguir:

I. Uma proposição composta é formada por uma sequência de, no mínimo, duas proposições simples.

II. A valoração de uma proposição composta por uma conjunção só será verdadeira quando pelo menos uma das proposições simples for verdadeira.

III. A valoração de uma proposição composta por uma disjunção inclusiva só será falsa quando pelo menos uma das proposições simples for falsa.

IV. Quando o operador de negação for aplicado no resultado lógico de uma proposição composta por uma conjunção, o resultado da conjunção será invertido.

É correto o que se afirma em:

a. III e IV, apenas.

b. I e II, apenas.

c. I apenas.

d. I e IV, apenas.

e. I, II, III e IV.

Conectivos e classificação textual

Diálogo aberto

Caro estudante, iniciamos agora mais uma seção do nosso estudo de introdução à lógica proposicional. Nesta nova etapa, você aprenderá novos conectivos lógicos, o que lhe possibilitará valorar proposições compostas mais complexas, desenvolvendo suas habilidades de aplicar a lógica proposicional com suas regras e formas. Você já deve ter ouvido e falado muito frases do tipo "Se chover, eu não vou sair de casa", ou então "Se eu ganhar na loteria, não vou mais trabalhar!". Esse tipo de frase possui uma estrutura lógica embasada em uma condição: se algo acontecer, então outro evento também irá acontecer. Nesta seção, vamos entender esse tipo de estrutura lógica, bem como outros importantes conceitos.

Dando sequência ao seu teste de lógica para uma vaga de desenvolvedor *trainee* em uma grande empresa de tecnologia, chegou a hora de vencer mais um desafio, no qual você deverá criar estruturas condicionais usando os operadores lógicos para resolver um problema com fórmulas. Foram-lhe passadas duas fórmulas:

$$\boxed{x = \frac{-b \pm \sqrt{\Delta}}{2a}} \quad \boxed{\Delta = b^2 - 4ac} \; .$$

$$\neg(A \wedge B) \Leftrightarrow \neg A \vee \neg B \; .$$

A primeira é a famosa fórmula de Bhaskara, usada para calcular as raízes de uma equação do segundo grau ($y = ax^2 + bx + c$). Já a segunda fórmula pertence ao cálculo proposicional e é uma das leis de De Morgan.

Utilizando as constantes (a, b, c) da equação do segundo grau e da fórmula de Bhaskara, você deverá escrever proposições simples e, a partir delas, criar implicações lógicas, utilizando a notação simbólica, para as seguintes regras:

Se o coeficiente ***a*** for positivo, então a parábola tem a concavidade virada para cima.

Se o coeficiente ***a*** for negativo, então a parábola tem a concavidade virada para baixo.

Se o valor do delta (Δ) for positivo, então a equação possui duas raízes reais distintas.

Se o valor do delta (Δ) for negativo, então a equação não possui raízes reais.

Se o coeficiente a for positivo e o valor do delta (Δ) for positivo, então a parábola tem a concavidade virada para cima e a equação possui duas raízes reais distintas.

Para a segunda fórmula, você deverá utilizar as regras do cálculo proposicional e fazer a demonstração, passo a passo, da veracidade da lei anunciada por Augusto De Morgan.

Com base no seu conhecimento, quantas proposições simples devem ser usadas para traduzir as regras apresentadas para a fórmula da equação do segundo grau e de Bhaskara? Quais conectores devem ser utilizados para escrever a fórmula de maneira correta? Qual mecanismo pode ser usado para fazer a demonstração da lei de De Morgan? Existe alguma sequência específica para a resolução? Como a fórmula deverá ser valorada?

Para vencer esse desafio proposto, você aprenderá novos conectivos lógicos, bem como construir e valorar fórmulas do cálculo proposicional.

Leia com atenção o conteúdo da seção. Bons estudos!

Não pode faltar

O cálculo proposicional fornece mecanismos para validar argumentos, tais mecanismos envolvem a utilização de proposições, que podem ser simples (apenas uma afirmação) ou compostas. Nesse segundo caso, temos um encadeamento de proposições simples usando conectivos lógicos.

Uma proposição composta 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 inclusiva ou exclusiva.

Conectivo lógico de disjunção – ou (exclusivo)

Considere as seguintes proposições simples:

A: João é estudante.

B: João é trabalhador.

C: João é Paulista.

D: João é Carioca.

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

R: João é estudante ou é trabalhador.

S: João é Paulista ou é Carioca.

A proposição R representa uma disjunção inclusiva, pois João pode ser estudante e também trabalhador. Já a proposição S 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 R, pode ser escrita como $A \vee B$. Já a disjunção exclusiva é representada pelo símbolo $\veebar$, ou seja, a proposição S pode ser escrita como C $\veebar$ D.

Na valoração de uma disjunção exclusiva, o resultado será verdadeiro se, e somente se, apenas umas 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 $A \rightarrow B$.

Exemplificando

Considere as proposições a seguir:

P: João estuda para a prova.

R: João passa de ano.

A proposição $P \rightarrow R$ (lê-se Se P, então R), deve ser traduzida como: **Se** João estudar para a prova, **então** passará de ano.

Veja que para fazer sentido a composição da sentença condicionada, ajustamos os verbos estudar e passar. Mas o que realmente importa é entender a condição que foi criada. Veja que a proposição B está condicionada à proposição A, ou seja, B depende de A para acontecer.

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 for falso, o resultado será falso ($V \rightarrow F = F$). Vejamos um exemplo:

A: O interruptor da sala foi desligado.

B: A luz da sala apagou.

C: $A \rightarrow B$.

A proposição C 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, C é verdade. Porém, se o interruptor for desligado, mas por algum motivo a luz não se apagar, então temos o caso $V \rightarrow F = F$, ou seja, C é falso.

Reflita

Em um condicional existe o antecedente e o consequente, ou seja, uma proposição depende da outra. Se trocarmos a ordem do antecedente pelo consequente alteramos o resultado da condição? Quais implicações essa troca poderia trazer ao algoritmo?

Na construção de algoritmos, os condicionais são amplamente utilizados, veja o que nos diz um grande autor da área da computação: "Do ponto de vista computacional, uma condição é uma expressão booleana cujo resultado é um valor lógico falso ou verdadeiro. Assim, uma expressão booleana como condição é conseguida com uma relação lógica entre dois elementos e um operador relacional" (MANZANO; OLIVEIRA, 2019, p. 68). Veja que na computação usamos o mesmo recurso da Lógica Formal. A expressão booleana mencionada refere-se à álgebra Booleana que está embasada na Lógica Clássica. Uma expressão booleana é uma proposição que resultará em verdadeiro ou falso.

Na construção de algoritmos, o condicional aparece nas estruturas de decisão, também chamada Desvio Condicional. O nome "desvio" representa exatamente o que acontece em um algoritmo, quando aparece um condicional, pois dependendo do resultado (V ou F) o programa fará uma ação diferente. Por exemplo, imagine que estamos implementando um software para uma loja que oferece opções de pagamento à vista ou a prazo. Caso o comprador pague à vista ele terá um desconto de 10% na compra, que deve ser aplicado pelo próprio sistema. Pois bem, dadas as proposições:

A: Pagamento feito à vista.

B: Conceder desconto de 10%.

No algoritmo deverá ser implementada a regra: $A \to B$. "Se o pagamento for à vista, então será concedido um desconto de 10%".

A expressão "Se... então" é a mais comum de se utilizar para o condicional, até mesmo porque na construção de algoritmos usamos exatamente essas palavras. Mas a implicação lógica pode ser escrita de outras formas, conforme ilustra o Quadro 3.3, que nos apresenta 8 formas diferentes de "traduzir" essa expressão lógica.

Quadro 3.3 | Expressões para o condicional

Expressão em português	Conectivo lógico	Expressão lógica
1. Se A, então B 2. A condicional B. 3. A, logo B. 4. A só se B; A somente se B. 5. B segue de A. 6. A é uma condição suficiente para B. 7. Basta A para B. 8. B é uma condição necessária para A.	Condicional	$A \to B$

Fonte: adaptado de Gersting (2017, p. 4).

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:

P: Lucas receberá o dinheiro.

Q: Lucas completará o trabalho.

S: $A \leftrightarrow B$.

A proposição S, deve ser traduzida como "Lucas receberá o dinheiro se, e somente se, completar o trabalho".

Segundo Gersting (2017), o bicondicional é um atalho para a expressão lógica: $(A \to B) \wedge (B \to A)$. Nessa expressão temos a conjunção entre o resultado de duas condicionais que alteram seus antecedentes e consequentes. Para ficar mais claro, usando as proposições P, Q criadas anteriormente, dizer que "Lucas receberá o dinheiro se, e somente se, completar o trabalho" é o mesmo que dizer "Se Lucas receber o dinheiro então completará o trabalho e se Lucas completar o trabalho então receberá o dinheiro". Como podemos observar, o bicondicional resume a sentença, facilitando até mesmo a compreensão.

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 $F \leftrightarrow F = V$.

Fórmula bem-formulada ou fbf

Assim como fez Aristóteles, a partir de agora, vamos focar na forma e nos valores lógicos que as expressões podem assumir. Já sabemos que é possível criar proposições compostas, fazendo conexões entre proposições simples. Na verdade, podemos encadear preposições, conectivos e parênteses (ou colchetes) e formar novas expressões lógicas, as quais chamamos fórmula. Por exemplo, como a que vimos no conectivo bicondicional $(A \rightarrow B) \wedge (B \rightarrow A)$. Embora "Uma sequência qualquer de elementos do vocabulário do cálculo proposicional constitui uma fórmula" (BISPO; CASTANHEIRA, 2011, p. 12), nem toda fórmula é válida. Segundo Gersting (2017), certas regras de sintaxe precisam ser seguidas, assim como acontece em qualquer linguagem de programação. Podemos fazer uma analogia entre as fórmulas do cálculo proposicional com as fórmulas matemáticas. Os conectivos lógicos são como os operadores matemáticos (soma, subtração, etc.), portanto sempre teremos um conectivo entre duas proposições. O operador de negação é como o sinal negativo na matemática e, por isso, ele pode aparecer perto de outro conector. Uma fórmula que segue as regras de sintaxe é chamada de fórmula bem-formulada ou ainda, em inglês, *well-formed formula* - wff (BISPO; CASTANHEIRA, 2011; GERSTING, 2017). Observe o Quadro 3.4. Nele temos três exemplos de fórmulas matemáticas, três de fórmulas válidas (fbf) e três de fórmulas inválidas.

Quadro 3.4 | Fórmulas matemáticas e proposicional

Expressão matemática	fbf	Não fbf
$(2+3)*5$	$(A \rightarrow B) \vee C$	$AA \wedge B$
$(3+4)*(2+3)$	$(A \rightarrow B) \wedge (B \rightarrow A)$	$\wedge \vee AB$
$2+3*-5$	$A \rightarrow B \wedge \neg C$	$\wedge \neg B$

Fonte: elaborado pela autora.

Vamos começar a analisar as fórmulas do Quadro 3.4 falando sobre os parênteses. Na fórmula matemática da linha 1, o resultado é 25, certo? E se não tivesse parênteses delimitando a adição o resultado seria o mesmo? A resposta é não, o resultado seria 17, afinal a multiplicação tem precedência de resolução sobre a adição. Na fbf da linha 1 também temos parênteses e o motivo é o mesmo da expressão matemática: queremos que a condicional seja resolvida antes da disjunção. Portanto, os parênteses no cálculo proposicional também têm o papel de delimitar e indicar quais operações devem ser efetuadas primeiro. Assim como os operadores matemáticos, os conectivos lógicos também possuem ordem de precedência, sendo ela:

1. Para conectivos dentro de vários parênteses, efetuam-se primeiro as expressões dentro dos parênteses mais internos.
2. Negação ($\leftarrow$).
3. Conjunção e disjunção ($\wedge$, $\vee$).
4. Condicional ($\rightarrow$).
5. Bicondicional ($\leftrightarrow$).

Quando dois operadores tiverem a mesma ordem de precedência, será valorado primeiro o que estiver mais à esquerda, da mesma forma que acontece em uma fórmula matemática.

Agora que sabemos a ordem de precedência dos conectivos lógicos, podemos interpretar a fbf da linha 2 no Quadro 3.4. Veja que primeiro será valorada a fórmula $(A \rightarrow B)$, em seguida $(B \rightarrow A)$ e, por fim, a conjunção entre os dois resultados.

Já na fbf da linha 3, não temos parênteses, então devemos seguir a ordem de precedência e valorar da esquerda para direita. Então seria primeiro efetuada a negação $\leftarrow C$, em seguida seria calculado a conjunção de B com o resultado da negação: $B \wedge \neg C$ e, por fim, seria feita a condicional entre A o resultado da conjunção.

Esse entendimento é muito importante, então vamos valorar a fbf da linha 3, dadas as seguintes entradas para as proposições A, B, C.

A é verdadeira.

B é falsa.

C é falsa.

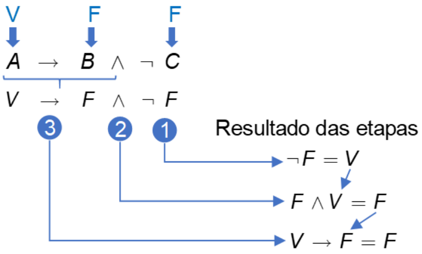
Observe a Figura 3.3, ao submeter os valores-lógicos de entrada das proposições, seguindo a ordem de execução, tem-se a seguinte sequência:

1. $\leftarrow F$ que resulta em V.
2. $F \wedge V$ que resulta em F.
3. $V \rightarrow F$ que resulta em F.

Portanto, o resultado da fbf $A \rightarrow B \wedge \neg C$ é falso.

Ao observar o resultado das etapas na Figura 3.3, fica claro que o resultado de uma etapa é usado em outra etapa, pois a fórmula precisa ser resolvida em partes seguindo a ordem de precedência.

Figura 3.3 | Esquema de valoração de fórmulas



Fonte: elaborada pela autora.

Como você pode notar a valoração de uma fbf deve ser feita em etapas, seguindo a ordem de precedência, assim como acontece em uma fórmula matemática.

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**.

Assimile

Quando duas fórmulas apresentam o mesmo resultado lógico para as todas as possíveis combinações de entradas, então diz-se que elas são equivalentes. O símbolo usado para representar a equivalência lógica é o $\Leftrightarrow$.

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

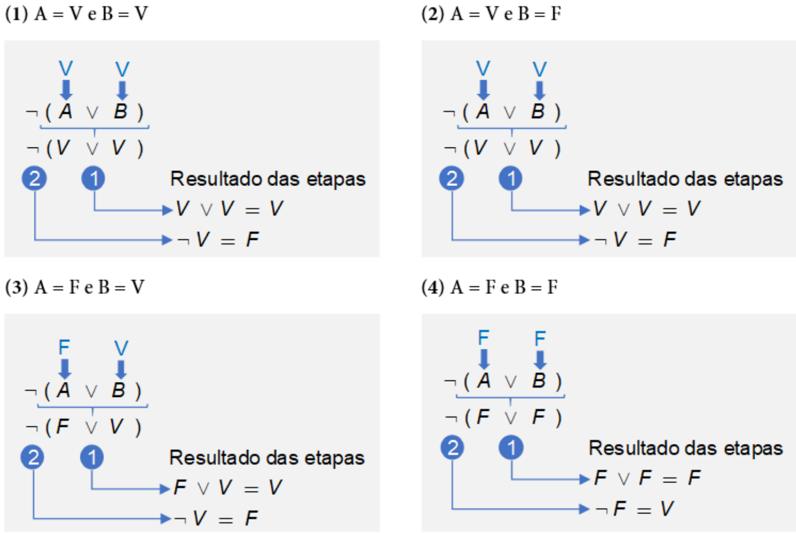
I. $\neg(A \vee B) \Leftrightarrow \neg A \wedge \neg B$.

II. $\neg(A \wedge B) \Leftrightarrow \neg A \vee \neg B$.

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 de entrada. Então temos as seguintes combinações possíveis para as proposições A, 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 $\neg(A \vee B)$. A Figura 3.4 ilustra cada passo para a valoração 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.

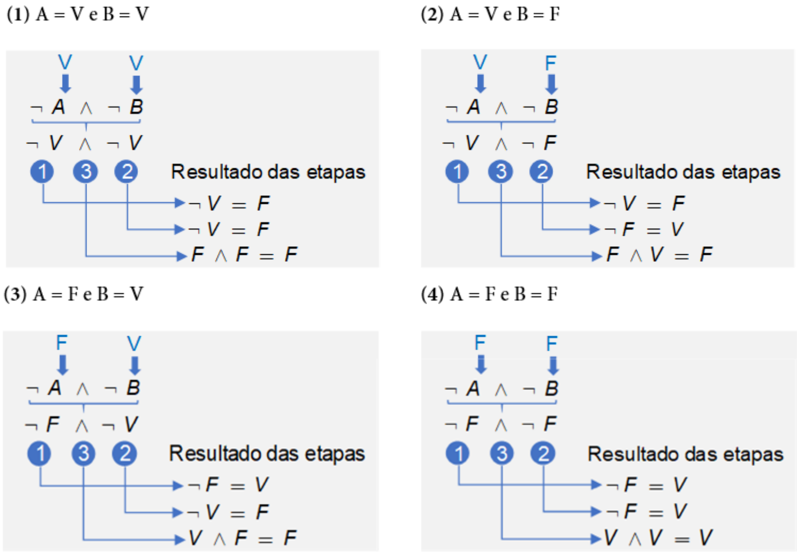
Figura 3.4 | Valoração da fórmula $\neg(A \vee B)$



Fonte: elaborada pela autora.

Agora vamos fazer a valoração da fórmula, do lado direito da equivalência, ou seja, $\neg A \wedge \neg B$, para cada uma das combinações possíveis de entrada. A Figura 3.5 ilustra cada passo para a valoração da fórmula. 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.

Figura 3.5 | Valoração da fórmula $\neg A \wedge \neg B$



Fonte: elaborada pela autora.

Como podemos ver, os resultados lógicos das fórmulas $\neg(A \vee B)$ e $\neg A \wedge \neg B$, para todas as combinações possíveis de entradas, são os mesmos, portanto demonstramos que essas fórmulas são equivalentes.

Com essa demonstração, finalizamos nossa seção. Agora que já conhecemos todos os conectivos lógicos e como criar fórmulas válidas (fbf) para fazer a valoração de uma fbf, devemos seguir a ordem de precedência e ir resolvendo parte a parte da fórmula.

Até a próxima!

Sem medo de errar

Caro estudante, chegou o momento de vencer mais uma etapa do teste para o cargo de trainee em uma grande empresa de tecnologia. Você foi desafiado a construir implicações lógicas que traduzam regras referentes à equação do segundo grau e à fórmula de Bhaskara, e também a fazer a demonstração de uma das leis de De Morgan.

Vamos relembrar as regras que devem ser traduzidas para forma simbólica:

Se o coeficiente ***a*** for positivo, então a parábola tem a concavidade virada para cima.

Se o coeficiente ***a*** for negativo, então a parábola tem a concavidade virada para baixo.

Se o valor do delta (Δ) for positivo, então a equação possui duas raízes reais distintas.

Se o valor do delta (Δ) for negativo, então a equação não possui raízes reais.

Se o coeficiente a for positivo e o valor do delta (Δ) for positivo, então a parábola tem a concavidade virada para cima e a equação possui duas raízes reais distintas.

Como primeiro passo você deve escrever as proposições simples, as quais nos possibilitarão construir as implicações lógicas para a equação do segundo grau e a fórmula de Bhaskara. A seguir, uma das possibilidades de se escrever essas proposições.

A: O coeficiente ***a*** da equação é positivo e diferente de zero.

B: A parábola tem a concavidade virada para cima.

C: A parábola tem a concavidade virada para baixo.

D: O valor do delta é positivo e diferente de zero.

E: A equação possui duas raízes reais distintas.

F: A equação não possui raízes reais.

Com as proposições simples definidas, agora podemos escrever os condicionais que representam, simbolicamente, as regras elencadas.

Para a regra 1, podemos escrever: $A \rightarrow B$.

Para a regra 2, podemos escrever: $\neg A \rightarrow C$.

Para a regra 3, podemos escrever: $D \rightarrow E$.

Para a regra 4, podemos escrever: $\neg D \rightarrow F$.

Para a regra 5, podemos escrever: $(A \wedge D) \rightarrow (B \wedge E)$.

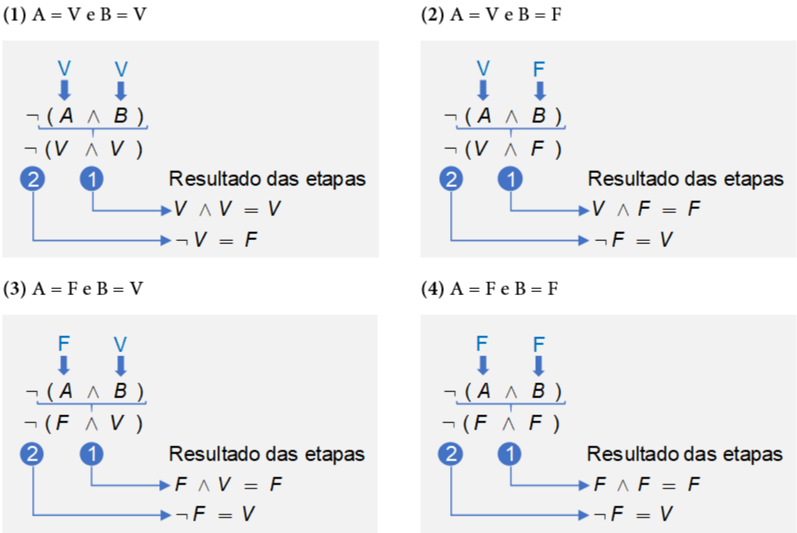
Veja que na regra 5 temos uma condição que envolve a conjunção entre duas proposições. Para construir, basta ficar atento aos conectivos que estão sendo usados na frase e na forma como se anunciou as proposições simples.

Agora é hora de fazer a demonstração da lei de De Morgan $\neg(A \wedge B) \Leftrightarrow \neg A \vee \neg B$. Essa fórmula apresenta uma equivalência, ou seja, De Morgan afima que a fórmula $\neg(A \wedge B)$ apresenta os mesmos resultados que a fórmula $\neg A \vee \neg B$ para todas as combinações possíveis de entradas lógicas para A, B. Portanto, para resolver essa parte do desafio é necessário testar todas as entradas possíveis para as proposições em cada uma das fórmulas e verificar se os resultados são iguais.

Primeiro vamos estabelecer a sequência de combinações de entrada que vamos usar: **(1)** A = V e B = V; **(2)** A = V e B = F; **(3)** A = F e B = V; **(4)** A = F e B = F;

Agora é necessário submeter essa sequência a cada uma das fórmulas. A Figura 3.6 apresenta os resultados para a fórmula $\neg(A \wedge B)$. Como resultado lógico temos para a entrada (1) o resultado é F, já para as entradas (2), (3) e (4) o resultado é V.

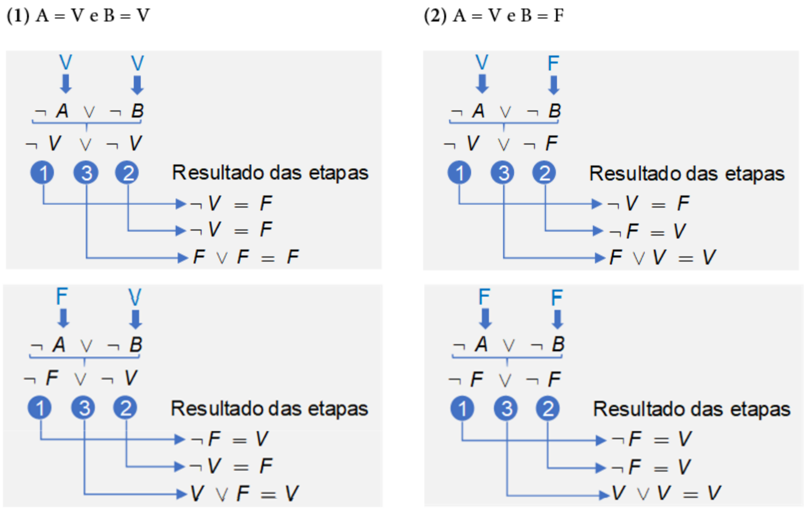
Figura 3.6 | Valoração da fórmula $\neg(A \wedge B)$



Fonte: elaborada pela autora.

A Figura 3.7 apresenta os resultados para a fórmula $\neg A \vee \neg B$. Como resultado lógico temos para a entrada (1) o resultado é F, já para as entradas (2), (3), e (4) o resultado é V.

Figura 3.7 | Valoração da fórmula $\neg A \vee \neg B$



Fonte: elaborada pela autora.

Como podemos ver, os resultados lógicos das fórmulas $\neg(A \wedge B)$ e $\neg A \vee \neg B$, para todas as combinações possíveis de entradas, são os mesmos, portanto, demonstramos que essas fórmulas são equivalentes.

Com utilização da lógica proposicional conseguimos construir estruturas condicionais e fazer a demonstração de um importante resultado. As estruturas condicionais são amplamente utilizadas na construção de algoritmos, portanto quanto mais praticar mais preparado estará para os desafios profissionais.

Faça valer a pena

1. Segundo Gersting (2017) através do encadeamento de preposições, conectivos e parênteses (ou colchetes) é possível formar novas expressões lógicas, chamadas fórmulas. Essas fórmulas são como fórmulas matemáticas, por exemplo, (1 - 3) * 5. Assim como existem regras para se escrever uma fórmula matemática, também existem para uma fórmula do cálculo proposicional. Quando escrita da maneira correta a fórmula é chamada de fbf (fórmula bem-formulada).

Quanto às fórmulas bem formuladas (fbf), analise as afirmativas a seguir:

I. $(A \wedge \neg B) \rightarrow C$.

II. $A \neg \wedge B \rightarrow D$.

III. $\neg D \rightarrow B \wedge D$.

IV. $B \wedge (\neg A) \rightarrow C$.

É correto o que se afirma em:

a. Apenas a fórmula I é uma fbf.

b. Apenas as fórmulas I e IV são fbfs.

c. Apenas as fórmulas I, III e IV são fbfs.

d. Apenas as fórmulas II, III e IV são fbfs.

e. As fórmulas I, II, III e IV são fbfs.

2. Através de valores lógicos de entrada para as proposições, uma fbf pode ser valorada em verdadeira ou falsa. Essa valoração precisa ser feita em partes, assim como acontece em uma fórmula matemática, por exemplo $(2+3)*4$. Primeiro é feita a adição entre os parênteses, que resulta em 5, na sequência é feita a multiplicação de 5 por 4, que resulta em 20, ou seja, o resultado da primeira parte é usado como entrada para a próxima operação. A mesma ideia deve ser usada para valorar uma fórmula proposicional, seguindo as regras de precedência dos operadores (GERSTING, 2017).

Neste contexto, avalie as seguintes asserções e a relação proposta entre elas.

I. Dada a seguinte combinação de entradas: A = F, B = F, C = V para as proposições A, B, C, a valoração da fórmula $\neg A \rightarrow B \wedge \neg C$ é falsa.

PORQUE

II. Primeiro devem ser resolvidas as negações, que inverterão o resultado das entradas para as proposições, em seguida deve ser resolvida a implicação lógica e, por fim, a conjunção entre os resultados.

A respeito dessas asserções, assinale a alternativa correta:

a. As asserções I e II são proposições verdadeiras, mas a II não justifica a I.

b. As asserções I e II são proposições verdadeiras e a II justifica a I.

c. A asserção I é uma proposição verdadeira e a II, falsa.

d. A asserção I é uma proposição falsa e a II, verdadeira.

e. As asserções I e II são proposições falsas.

3. Em uma fbf os conectivos lógicos são como os operadores matemáticos (soma, subtração, etc.), ou seja, se para fazer a adição é necessário usar o sinal "+" entre dois números, para fazer a conjunção, ou disjunção, ou o condicional, ou o bicondicional é preciso duas proposições, uma antes, outra após o conectivo. O operador lógico de negação é como o sinal negativo na matemática e, por isso, ele pode aparecer perto de outro conector, sem a necessidade de vir entre duas proposições.

Considerando a seguinte combinação de entradas: A = V, B = F, C = F, D = V para as proposições A, B, C, D, analise as afirmativas a seguir:

I. A valoração para a fórmula $A \wedge B \vee C$ é falsa.

II. A valoração para a fórmula $D \rightarrow B$ é falsa.

III. A valoração para a fórmula $B \vee D \neg C$ é verdadeira.

IV. A valoração para a fórmula $C \vee A \rightarrow D$ é verdadeira.

É correto o que se afirma em:

a. I e II, apenas.

b. I, II e IV, apenas.

c. I e IV, apenas.

d. II e III, apenas.

e. I, II, III e IV.

Seção 3

Métodos dedutivos e inferência lógica

Diálogo aberto

Caro estudante, dando continuidade ao nosso estudo de lógica computacional, chegou um momento muito importante no qual aprenderemos a fazer demonstrações lógicas. O raciocínio usado para fazer uma dedução lógica é similar ao usado para construir sequências lógicas em um algoritmo, pois, em ambos os casos, é necessário escrever uma sequência correta de passos, tanto em termos de sintaxe, quanto em relação ao raciocínio e as regras necessárias para alcançar o resultado desejado.

Dando continuidade ao processo seletivo para a vaga de *trainee*, nessa última fase do processo, os contratantes querem testar seu raciocínio lógico, bem como seu conhecimento sobre as regras de dedução da Lógica Formal. Você recebeu dois argumentos:

a. Se o papel de tornassol ficar vermelho, então a solução e ácida. O papel de tornassol ficou vermelho. Portanto, a solução é ácida.

b. Se treino, eu venço o campeonato de xadrez. Se não jogo vôlei, então eu treino xadrez. Não venci o campeonato de xadrez. Portanto, joguei vôlei.

Seu desafio é traduzir para forma simbólica os dois argumentos e provar a veracidade, usando as regras de dedução da Lógica Formal. Cada passo na sequência de demonstração deve ser comentado, para que os avaliadores tenham certeza que você conhece o processo. Quantos passos serão necessários para demonstrar cada argumento? Será possível fazer uma demonstração usando somente regras de inferência? Sabendo que é mais importante conhecer o processo do que decorar regras, os avaliadores permitiram que você usasse a Internet para consultar as regras de equivalência e inferência lógica.

Para vencer esse desafio, você verá nessa seção as regras para avaliar um argumento e verificar se ele é válido ou não. Também aprenderá como estruturar a sequência de demonstração, desenvolvendo, assim, o raciocínio e dando um passo a mais na direção de se tornar um grande desenvolvedor.

Vamos em frente!

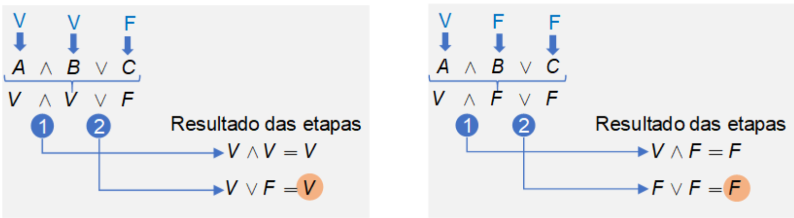
Não pode faltar

A lógica proposicional é composta por proposições e conectivos lógicos que permitem criar uma série de fórmulas que quando escritas corretamente são chamadas fbfs (fórmula bem-formulada). Uma fbf é valorada em verdadeira (V) ou falsa (F), a partir da valoração das proposições com o conectivo lógico em questão, respeitando a ordem de precedência dos operadores lógicos. A valoração de uma fórmula também depende dos valores lógicos de entrada para cada uma das proposições. Por exemplo, dada a fórmula $A \wedge B \vee C$ ela será verdadeira ou falsa? A valoração depende dos valores lógicos de entrada para as proposições A, B, C. Observe a Figura 3.8a, na qual fazemos a valoração para as entradas A = V, B = V, C = F, veja que o resultado final é V. Considerando os valores lógicos: A = V, B = F, C = F, a mesma fórmula teve valoração final F (Figura 3.8b). Portanto, a valoração de uma fbf depende das entradas lógicas e dos conectivos lógicos que são usados.

Figura 3.8 | Valoração para a fórmula $A \wedge B \vee C$

a. A = V, B = V, C = F

b. A = V, B = F, C = F



Fonte: elaborada pela autora.

Quando uma fórmula apresenta um conjunto de proposições, das quais uma delas é uma conclusão, dizemos que tal fórmula é um argumento. “Um argumento é um conjunto de proposições, ou de fórmulas, nas quais uma delas (conclusão) deriva, ou é consequência, das outras (premissas)” (BISPO; CASTANHEIRA, 2011, p. 31). Um argumento pode ser representado de forma simbólica por:

$$\boxed{P_1 \wedge P_2 \wedge P_3 \wedge ... \wedge P_n \rightarrow C} \ ,$$

onde $P_1, P_2, P_3, ..., P_n$, são as **hipóteses**. Essas hipóteses podem ser tanto proposições simples, como uma fbf. A letra representa C a **conclusão** do argumento, a qual também pode ser tanto uma proposição simples como uma fbf (BISPO; CASTANHEIRA, 2011; GERSTING, 2017).

Assimile

Um argumento é composto por hipóteses e conclusão, e ambas podem ser compostas por proposições simples ou fbf. No argumento, as proposições são ligadas logicamente pelo conectivo de conjunção (e), as quais implicam logicamente a conclusão. Por isso, a ligação entre as hipóteses e a conclusão é feita por meio do conectivo condicional.

Dado um argumento é importante validar se ele é válido ou inválido, o grande desafio é como fazer essa validação. A lógica possui mecanismos que permitem validá-lo, os quais são compostos pelas regras de equivalência e inferência lógica. Essas regras vão nos permitir avaliar a relação entre as hipóteses e a conclusão, que também pode ser chamada de consequência lógica, dedução lógica, conclusão lógica ou implicação lógica. "Uma proposição pode ser verdadeira ou falsa e não pode ser válida ou inválida; do mesmo modo, um argumento pode ser válido ou inválido e não pode ser verdadeiro ou falso" (BISPO; CASTANHEIRA, 2011, p. 36).

Exemplificando

Vamos analisar o seguinte argumento adaptado de Gersting (2017, p. 24).

D. Pedro I proclamou a independência do Brasil e Thomas Jefferson escreveu a Declaração de Independência dos Estados Unidos. Portanto, todo dia tem 24 horas.

Vamos separar as proposições do argumento em hipóteses e conclusão.

A: D. Pedro I proclamou a independência do Brasil.

B: Thomas Jefferson escreveu a Declaração de Independência dos Estados Unidos.

C: Todo dia tem 24 horas.

Nosso conhecimento nos permite valorar as três proposições, logo, A, B, C são todas verdadeiras. Embora, tanto as hipóteses, quanto a conclusão sejam proposições verdadeiras, o argumento é inválido, pois a conclusão nada tem a ver com as hipóteses.

Esse exemplo deixa claro que, basear-se apenas no conteúdo de um argumento não é suficiente para dizer se ele é válido ou não.

Segundo a Lógica Formal, devemos nos basear apenas nas regras para validar um argumento e não no conteúdo. O exemplo mencionado anteriormente deixa claro o motivo dessa restrição. Mas, afinal, quais são essas regras?

Para começarmos a entender as regras, vamos traduzir o argumento: D. Pedro I proclamou a independência do Brasil e Thomas Jefferson escreveu a

Declaração de Independência dos Estados Unidos. Portanto, todo dia tem 24 horas, para notação simbólica, logo temos a seguinte fórmula: $A \wedge B \rightarrow C$. Nessa fórmula quando o valor lógico de entrada da proposição A for verdadeiro e de B for falso, o resultado da implicação será falso, ou seja, existe pelo menos uma combinação de entradas, para a qual a fórmula resultará em falsa, logo essa fórmula não é uma **tautologia** e, consequentemente, não é um argumento válido. Segundo Gersting (2017), tautologia é um resultado no qual todas as entradas possíveis de uma fórmula obtêm verdadeiro como resultado. Ainda, segundo a mesma autora, um argumento só é válido quando a fórmula é uma tautologia.

Reflita

Um argumento composto por preposições verdadeiras não necessariamente é um argumento válido. Um argumento que contenha proposições falsas, pode ser um argumento válido?

Para saber se um argumento é válido ou não, precisamos saber se ele é uma tautologia. Para fazer essa checagem, poderíamos testar todas as combinações de entrada possíveis para o argumento. Porém, se tratando da Lógica Formal, podemos usar um sistema de **regras de dedução** e, seguindo uma **sequência de demonstração** provar se o argumento é válido ou não. "Uma sequência de demonstração é uma sequência de fbfs nas quais cada fbf é uma hipótese ou o resultado de se aplicar uma das regras de dedução do sistema formal a fbfs anteriores na sequência" (GERSTING, 2017 p. 25). Para construir a sequência de demonstração, cada hipótese, que é uma proposição e que pode ser uma fbf, deve ser disposta da seguinte forma:

1. P_1	(hipótese)
2. P_2	(hipótese)
...	...
n) P_n	(hipótese)
n+1) fbf_1	(resultado da aplicação de uma regra de dedução a hipóteses anteriores.)
n+2) fbf_2	(resultado da aplicação de uma regra de dedução a hipóteses anteriores.)
...	...
n+n) C	(resultado da aplicação de uma regra de dedução a hipóteses anteriores.)

Na sequência de demonstração, cada proposição deve ficar em uma linha, enumeramos para facilitar na hora de aplicar as regras de dedução. Na frente de cada linha devemos indicar o que ela representa, se é uma hipótese ou então a regra que foi aplicada. Após elencar todas as proposições é hora de começar a aplicar as regras e, consequentemente, obter novas fbfs. A quantidade varia de caso para caso, mas as regras de dedução devem ser aplicadas até que se consiga provar que o argumento é verdadeiro, ou então, que não existam mais regras a serem aplicadas, mostrando, assim, que o argumento é falso.

Regras de equivalência de dedução para a Lógica Proposicional

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 as 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 $\neg A \wedge \neg B$, pois ambas são equivalentes. No Quadro 3.5 estão elencadas as regras de equivalência que iremos utilizar. Existem outras além dessas. (Para mais detalhes, consultar da página 38 a 41 da obra: BISPO, F.; CASTANHEIRA, L. B. **Introdução à lógica matemática**. São Paulo: Cengage Learning, 2011, disponível em sua biblioteca virtual.)

Expressão (fbf)	Equivalente (fbf)	Nome/Abreviação
$P \vee Q$ $P \wedge Q$	$Q \vee P$ $Q \wedge P$	Comutatividade/com
$(P \vee Q) \vee R$ $(P \wedge Q) \wedge R$	$P \vee (Q \vee R)$ $P \wedge (Q \wedge R)$	Associatividade/ass
$\neg(P \vee Q)$ $\neg(P \wedge Q)$	$\neg P \wedge \neg Q$ $\neg P \vee \neg Q$	Leis de De Morgan/De Morgan
$P \rightarrow Q$	$\neg P \vee Q$	Condicional/cond
P	$\leftarrow(\leftarrow P)$	Dupla negação/dn
$P \leftrightarrow Q$	$(P \rightarrow Q) \wedge (Q \rightarrow P)$	Definição de equivalência/que

Fonte: adaptada de Gersting (2017, p. 26).

O Quadro 3.5 nos traz 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 substitui-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.

Regras de inferência de dedução para a Lógica Proposicional

Além das regras de equivalência, o processo de dedução lógica também possui as regras de inferência. Na inferência, dada uma determinada fbf, ela poderá ser substituída por outra que atenda a regra de inferência. Veja que aqui não é necessário ser uma tautologia (e realmente não será), mas é preciso seguir as regras da inferência.

A primeira regra de inferência que vamos estudar é chamada **Modus Ponens (MP)**. Essa regra envolve uma implicação e uma conjunção e possui a seguinte estrutura: $(P \to Q) \wedge P \to Q$. Vejamos um exemplo para ficar mais clara a regra. Considerando o seguinte argumento: *Se João receber seu salário, ele irá ao cinema*, vamos separar as proposições P, Q, portanto:

P: João recebe o salário.

Q: João vai ao cinema.

Agora vamos representar cada parte da fórmula de Modus Ponens:

I. $(P \to Q)$: Se João receber seu salário, ele irá ao cinema.

II. *P*: João recebe o salário.

Ao fazer a conjunção entre a primeira parte com a segunda, conseguimos inferir a conclusão, pois, se João receber seu salário, ele irá ao cinema **E** João recebeu o salário, logo podemos inferir (concluir) que João vai ao cinema.

A regra de Modus Ponens também pode ser representada pelo esquema:

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

. Aqui fica claro que quando se tem uma implicação e o antecedente é verdade, então a conclusão é o consequente.

Outra importante regra de inferência é o **Modus Tollens (MT)**. Essa regra, além de envolver uma implicação e uma conjunção, também envolve a negação de uma das proposições. Sua estrutura é dada pela fbf: $(P \to Q) \wedge \neg Q \to \neg P$. Vejamos um exemplo. Considere o seguinte argumento:

Se João desligar o interruptor, então a lâmpada se apaga. Vamos separar as proposições P, Q, portanto:

P: João desliga o interruptor.

Q: A lâmpada apaga.

Agora vamos representar cada parte da fórmula de Modus Tollens:

I. $(P \to Q)$: Se João desligar o interruptor, então a lâmpada se apaga.

II. $\leftarrow Q$: A lâmpada não apagou.

Ao fazer a conjunção entre a primeira parte com a segunda, conseguimos inferir a conclusão. Pois, se João desligar o interruptor, a lâmpada se apaga **E** a lâmpada não se apagou, logo podemos inferir (concluir) que João não desligou o interruptor.

A regra de Modus Tollens também pode ser representada pelo esquema:

$$\begin{array}{l} P \to Q \\ \neg Q \\ \hline \quad \neg P \end{array}$$

. Aqui fica claro que quando se tem uma implicação e o consequente não é verdade, então a conclusão é que o antecedente também não aconteceu.

Resumindo esses dois métodos, no Modus Ponens, usamos a implicação para provar que a consequência é verdadeira ao demonstrar que a premissa é verdadeira. Já no Modus Tollens, usamos a implicação para provar que a premissa é falsa ao demonstrar que a consequência é falsa.

Outra importante regra é o **Silogismo Hipotético** (SH). Nessa regra, além de existirem implicações e conjunções nas hipóteses, a conclusão também é uma implicação. Sua estrutura é dada pela fbf: $(P \to Q) \wedge (Q \to R) \to (P \to R)$. Vejamos um exemplo. Considere o seguinte argumento: Se as árvores começam a florir, então começa a primavera. Se começa a primavera, então as árvores dão frutos. Vamos separar as proposições P, Q, R portanto:

P: As árvores começam a florir.

Q: A primavera começa.

R: As árvores dão frutos.

Agora vamos representar cada parte da fórmula de Silogismo Hipotético:

I. $(P \to Q)$: Se as árvores começam a florir, então começa a primavera.

II. $(Q \to R)$: Se começa a primavera então as árvores dão frutos.

Ao fazer a conjunção entre a primeira parte com a segunda, conseguimos inferir a conclusão. Pois, se as árvores começam a florir, então começa a primavera **E** se começa a primavera então as árvores dão frutos, logo podemos inferir (concluir) que se as árvores começam a florir, então darão frutos.

A regra de Silogismo Hipotético também pode ser representada pelo esquema: $\dfrac{\begin{array}{c} P \rightarrow Q \\ Q \rightarrow R \end{array}}{P \rightarrow R}$. Veja que a consequente de uma proposição é a antecedente na outra e, por isso, o resultado pode ser inferido do antecedente da primeira para a consequente da segunda proposição.

O Quadro 3.6 apresenta um resumo de algumas das principais regras de inferências. Além das regras já apresentadas, estão elencadas mais três regras à conjunção, na qual dadas duas proposições elas podem ser unidas por essa operação. A simplificação, que faz o contrário da conjunção, separa duas proposições. A adição, que adiciona uma nova proposição a uma já existente por meio de uma conjunção.

Quadro 3.6 | Regras de inferência

De (fbf)	Podemos deduzir (fbf)	Nome/Abreviação
$P \rightarrow Q$, P	Q	Modus Ponens/MP
$P \rightarrow Q$, $\leftarrow Q$	$\leftarrow P$	Modus Tollens/MT
$P \rightarrow Q$, $Q \rightarrow R$	$P \rightarrow R$	Silogismo Hipotético/SH
P, Q	$P \wedge Q$	Conjunção/conj
$P \wedge Q$	P, Q	Simplicação/simp
P	$P \wedge Q$	Adição/ad

Fonte: adaptada de Gersting (2017, p. 27).

Assimile

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. Em outras palavras, “Ao contrário das regras de equivalência, as regras de inferência não funcionam em ambas as direções” (GERSTING, 2017, p. 27).

Agora que já conhecemos as principais regras de dedução lógica, vamos utilizar esse método para mostrar que o argumento $(\neg A \vee B) \wedge (B \rightarrow C) \rightarrow (A \rightarrow C)$ é válido. O primeiro passo é identificar as hipóteses e a conclusão. A fórmula do argumento ($P_1 \wedge P_2 \wedge P_3 \wedge ... \wedge P_n \rightarrow C$) nos diz que as hipóteses são ligadas pela conjunção e a conclusão é ligada pela última implicação lógica. Outro detalhe importante é que as hipóteses podem ser fbf e não somente proposições simples. Portanto, para nosso caso temos:

Hipótese 1: $(\neg A \vee B)$

Hipótese 2: $(B \rightarrow C)$

Conclusão: $(A \rightarrow C)$

Veja que na conclusão temos uma implicação, isso já nos dá indícios de que conseguiremos usar o silogismo hipotético. Vamos à sequência de demonstração.

1. $\neg A \vee B$ (hip).
2. $B \rightarrow C$ (hip).
3. $A \rightarrow B$ (1, cond).
4. $A \rightarrow C$ (3, 4, SH).

Em quatro passos conseguimos demonstrar que o argumento é válido. Nos passos 1 e 2 elencamos as hipóteses. No passo 3, consultamos as regras de equivalência no Quadro 3.5, e usamos a regra de equivalência do condicional, ou seja, trocamos a hipótese $\neg A \vee B$ por $A \rightarrow B$ já que são equivalentes. Na linha 4, consultamos as regras de inferência no Quadro 3.6 e aplicamos o silogismo hipotético entre as linhas 3 e 2, ou seja, substituímos $(A \rightarrow B) \wedge (B \rightarrow C)$ por $A \rightarrow C$. Como chegamos exatamente a fbf da conclusão, provamos a validade do argumento. As regras de dedução lógica devem ser consultadas a todo momento no processo de demonstração. Não existe uma receita, somente a prática nos auxilia a desenvolvermos o raciocínio. Podemos fazer uma analogia do processo de demonstração com a atividade do programador. Na programação sequencial, temos uma entrada e uma saída desejável, portanto implementamos uma sequência lógica de passos para chegar a esse resultado. É trabalho do desenvolvedor descobrir qual a sequência de proposições (comandos) que farão a transformação dos dados de entrada, para a saída. Cada linha do programa tem que seguir as regras de sintaxe da linguagem de programação, bem como a lógica necessária para alcançar o resultado. Portanto, praticar esse raciocínio de sequência lógica é um passo importante na formação do profissional de tecnologia.

Sem medo de errar

Chegou a hora de resolver seu último desafio do processo seletivo para a vaga de trainee em uma grande empresa de tecnologia. Você recebeu como desafio, avaliar dois argumentos, mostrando se eles são válidos ou não. Para isso, você deve usar as regras de dedução lógica. Relembrando os dois argumentos que lhe foram passados:

a. Se o papel de tornassol ficar vermelho, então a solução é ácida. O papel de tornassol ficou vermelho. Portanto, a solução é ácida.

b. Se treino, eu venço o campeonato de xadrez. Se não jogo vôlei, então eu treino xadrez. Não venci o campeonato de xadrez. Portanto, joguei vôlei.

Nessa etapa do desafio, você foi autorizado a usar a internet para consultar as regras de equivalência e inferência lógica.

O primeiro passo para resolver esse problema é localizar uma fonte confiável para consultar as regras. Uma ótima opção é usar seu acesso à biblioteca virtual e acessar a obra: GERSTING, J. L. **Fundamentos matemáticos para a ciência da computação**: matemática discreta e suas aplicações. 7. ed. Rio de Janeiro: LTC, 2017. Nas páginas 26 e 27 você encontrará as Tabelas 1.11 e 1.12 contendo as regras de equivalência e inferência. Outra opção para consulta é você utilizar o ambiente virtual e acessar seu livro didático, pois nele você poderá consultar os Quadros 3.5 e 3.6 que também contêm as regras necessárias para resolver o desafio.

Localizada uma fonte segura de informações, agora é preciso traduzir os argumentos para a forma simbólica e fazer a demonstração lógica. Vamos começar pelo argumento (a):

a. Se o papel de tornassol ficar vermelho, então a solução é ácida. O papel de tornassol ficou vermelho. Portanto, a solução é ácida.

Vamos traduzir o argumento para proposições:

P: O papel de tornassol fica vermelho.

Q: A solução é ácida.

Agora é possível traduzir o argumento para forma simbólica: $(P \to Q) \wedge P \to Q$. Agora podemos começar a sequência de demonstração, iniciando pela enumeração das hipóteses, seguida da aplicação de regras de dedução:

1. $P \to Q$ (hip).

2. P (hip).

3. Q (1, 2 MP).

No item 1 tem-se a primeira hipótese $P \rightarrow Q$. No segundo item, a segunda hipótese, lembrando que cada hipótese é conectada pela conjunção e, que cada uma delas pode ser fbf. Após elencar as hipóteses, consultamos o Quadro 3.6 e vimos que era possível aplicar a regra de Modus Ponens, ao aplicá-la na linha 3, chegamos exatamente na conclusão do argumento, logo esse argumento é válido.

Agora vamos demonstrar o segundo argumento:

b. Se treino, eu venço o campeonato de xadrez. Se não jogo vôlei, então eu treino xadrez. Não venci o campeonato de xadrez. Portanto, joguei vôlei.

Proposições:

P: Eu treino.

Q: Eu venci o campeonato de xadrez.

R: Eu jogo vôlei.

Forma simbólica do argumento: $(P \rightarrow Q) \wedge (\neg R \rightarrow P) \wedge \neg Q \rightarrow R$.

Sequência de demonstração:

1. $P \rightarrow Q$ (hip).
2. $\neg R \rightarrow P$ (hip).
3. $\neg Q$ (hip).
4. $\neg P$ (1, 3, MT).
5. $\neg\neg R$ (2, 4, MT).
6. R (6, dn).

Para demonstrar esse argumento, foram necessários seis passos, sendo três deles as hipóteses. Após elencar as hipóteses, consultamos o Quadro 3.6 e vimos que era possível aplicar a regra de Modus Tollens entre os itens 1 e 3, com isso obtivemos o resultado $\neg P$ no item 4. Também vimos que podíamos aplicar a mesma regra entre os itens 2 e 4, com isso obtivemos o resultado $\neg\neg R$ no item 5. Por fim, consultado o Quadro 3.5 vimos que podíamos aplicar a dupla negação no item 5 e obtivemos R no item 6. Como R é a conclusão, demonstramos que o argumento é válido.

Com essas demonstrações, finalizamos o desafio aplicando a lógica proposicional com suas formas e regras para resolver os mais diversos problemas. Agora é só esperar o resultado positivo da empresa!

Faça valer a pena

1. “Uma proposição pode ser verdadeira ou falsa e não pode ser válida ou inválida; do mesmo modo, um argumento pode ser válido ou inválido e não pode ser verdadeiro ou falso” (BISPO; CASTANHEIRA, 2011, p. 36).

Escolha a alternativa correta em que a fbfs é um argumento:

a. $(A \to B) \wedge C \vee D$.

b. $A \wedge B \wedge \neg C$.

c. $(A \to B) \vee C$.

d. $A \wedge (B \vee C) \to D$.

e. $A \vee (B \vee C) \wedge D$.

2. “Uma sequência de demonstração em um sistema de lógica formal é uma sequência de fbfs que são hipóteses ou que podem ser deduzidas de fbfs anteriores na sequência pelas regras de dedução do sistema” (GERSTING, 2017, p. 33).

Dado o argumento: “Se Marina for a autora, então o livro será de ficção. Mas o livro não é de ficção. Portanto, Marina não é a autora”, escolha a alternativa que representa a regra de dedução lógica que pode ser usada para representá-lo:

a. Comutatividade.

b. Leis de De Morgan.

c. Modus Tollens.

d. Modus Ponens.

e. Silogismo Hipotético.

3. O processo de dedução lógica é composto por regras de equivalência e de inferência lógica. “O sistema de lógica proposicional é completo e correto; argumentos válidos, e apenas esses, são demonstráveis” (GERSTING, 2017, p. 33). Dado o seguinte argumento: $[A \to (A \to B)] \wedge A \to B$, qual opção representa, sequencialmente, as regras usadas para demonstrá-lo?

a. hip, hip, MP, MT.

b. hip, hip, MP, MP.

c. hip, hip, DN, MP.

d. hip, hip, cond, MT.

e. hip, hip, que, MT.

Referências

BISPO, F.; CASTANHEIRA, L. B. **Introdução a lógica matemática.** São Paulo: Cengage Learning, 2011.

GERSTING, J. L. **Fundamentos matemáticos para a ciência da computação**: matemática discreta e suas aplicações. 7. ed. Rio de Janeiro: LTC, 2017.

MACHADO, N. J; CUNHA, M. O. da. **Lógica e linguagem cotidiana**: verdade, coerência, comunicação, argumentação. 2. ed. Belo Horizonte: Autêntica, 2008.

MANZANO, J. A. N. G.; OLIVEIRA, J. F. de. **Algoritmos**: lógica para desenvolvimento de programação de computadores. 29. ed. São Paulo: Érica, 2019.

SOUZA, J. A. L. de S. (org.). **Lógica matemática**. São Paulo: Pearson Education do Brasil, 2016.

Unidade 4

Vanessa Cadan Scheffer

Tabela Verdade

Convite ao estudo

Caro(a) estudante, seja bem-vindo e/ou bem-vinda! Reparou no modo como escrevi as boas-vindas (e/ou)? O que essa notação quer dizer? O que eu quis dizer ao escrever "e"? Será que a resposta seria a mesma se eu optasse pelo "ou"? Por exemplo: "vamos convidar somente a família *e* os amigos próximos para a festa". Qual seria o impacto se trocássemos E por OU nessa frase?

Como você pode ver, a lógica faz parte do nosso cotidiano, da nossa forma de comunicação.

Embora tenhamos esse conhecimento comum, precisamos ir além e compreender, com clareza, como esses conectores se comportam no mundo binário, pois da mesma forma que nos comunicamos uns com os outros por meio da linguagem natural, os componentes do computador se comunicam entre si usando a linguagem binária, e acredite, tudo é uma combinação de 0s, 1s e os conectores lógicos. Para auxiliar na compreensão e clareza dos resultados dos conectores, existe uma "ferramenta" chamada tabela verdade, que é o assunto desta unidade de ensino. A todo momento, nesta unidade, vamos desenvolver e praticar o raciocínio lógico utilizando a tabela verdade para que, ao final da unidade, você conheça e saiba aplicar os seus conceitos e fundamentos a partir de cases e aplicações computacionais.

Como funcionário *trainee* na área de *analytics* de uma empresa de varejo, você deve ser capaz de resolver os problemas propostos pela equipe a fim de ajudá-la e, em breve, ser promovido a júnior. Seu primeiro desafio consiste em construir uma tabela verdade com os resultados dos conectores de conjunção, disjunção e negação para *insights* de vendas. Seu segundo desafio é completar a tabela verdade com novos *insights*, baseados no conector de implicação, a fim de direcionar uma campanha promocional. Após completar as duas primeiras etapas, você deverá utilizar os novos dados recebidos para extrair *insights* para uma campanha promocional personalizada. Animado?!

Para solucionar os desafios propostos, você aprenderá a construir tabelas verdades com os conectores lógicos, analisar os resultados obtidos e solucionar fórmulas mais complexas.

Portanto, chegou a hora de se tornar um profissional mais completo e se apropriar dessa nova ferramenta! Vamos lá!!!

Construção da Tabela Verdade

Diálogo aberto

Caro estudante, seja bem-vindo! Iniciaremos agora uma jornada que lhe capacitará compreender e aplicar a lógica por meio da construção de tabelas verdade. Quem tem irmão já deve ter ouvido a mãe gritar: "*Você E seu irmão estão de castigo*!", mas esse "*E*" tem a ver com o conectivo lógico *AND*? A resposta é sim, o resultado desse E é o mesmo, seja na linguagem natural ou na binária, por isso aprendemos lógica proposicional e a aplicamos no universo computacional.

Você foi recentemente contratado como um funcionário *trainee* na área de *analytics* e almeja se tornar júnior em breve, mas para isso deve cumprir seus desafios e ajudar a equipe. Você recebeu uma planilha com os dados de compras de clientes, conforme ilustrado na Tabela 4.1. Dadas as seguintes proposições: **p: o cliente é do sexo feminino, q: o cliente tem idade entre 20 e 30 anos**, o seu desafio é construir uma Tabela Verdade que generalize a solução fazendo a conjunção e a disjunção para as proposições p e q, além de criar os resultados para a negação de ambas as fórmulas. Após criar a tabela verdade, você poderá analisar cada registro informando se o resultado é verdadeiro ou falso para cada um dos conectores lógicos propostos na Tabela 4.1. Tal resultado ajudará a equipe de vendas a criar rotinas para tomada decisões.

Tabela 4.1 | Dados de compra dos clientes

codigo_cli	nome_cli	genero_cli	idade_cli	valor_compra	E	OU
53682	Karly Dillon	F	40	74,84	?	?
58246	Channing Vazquez	M	49	98,04	?	?
27022	Adria Key	F	47	65,93	?	?
82075	Ella Nelson	F	34	94,01	?	?
90657	Arden Battle	M	48	21,73	?	?
80330	Brittany Ramirez	F	38	42,23	?	?
53989	Moses Graham	M	42	37,20	?	?
61370	Jin Fuller	M	49	65,60	?	?
41807	Phelan Blair	M	46	77,40	?	?
94269	Porter West	M	22	67,19	?	?

Fonte: elaborada pela autora.

Para cumprir seu desafio, nesta seção você aprenderá o mecanismo de construção de uma tabela verdade, bem como utilizar os conectores lógicos de conjunção, disjunção e negação. Então, mãos à obra!

Não pode faltar

Caro estudante, já deve ser do seu conhecimento que um computador é dividido em duas partes: o hardware (componentes físicos) e o software (os programas). Muitos pesquisadores contribuíram para que chegássemos ao nível de evolução computacional que vivenciamos. O avanço do hardware se deu por meio das pesquisas na área da eletrônica digital, que visa construir circuitos que representam grandezas por meio de valores discretos (DACHI; HAUPT, 2018). Os componentes eletrônicos que compõem um computador são formados por pequenos elementos, chamados de transistores, que são capazes de lidar com dois estados, aberto ou fechado. Um conjunto de transistores pode ser usado para construir uma porta lógica, que ao receber sinais digitais de entrada produz uma determinada saída, que depende do tipo de operação lógica para o qual foi construído. Por exemplo, uma porta lógica pode receber um sinal de ligado (1) em uma entrada e um sinal de desligado (0) em outra, qual seria o resultado? Depende da operação lógica dessa porta; se for uma operação AND, o resultado seria "desligado", mas se fosse um OR, o resultado seria "ligado".

Assim como no hardware, o software também possui operações lógicas. Por exemplo, podemos escrever um programa que irá somar dois valores se, e somente se, ambos forem positivos. Nesse caso, teremos que construir o algoritmo utilizando o operador AND.

Como você pode ver, tanto o hardware como o software computacional dependem da Lógica Formal. Sabemos que os fundamentos da lógica computacional estão baseados nas proposições e nos conectivos (ou operadores) lógicos, mas como podemos organizar os resultados das operações lógicas para facilitar nosso trabalho? Podemos seguir a sugestão de Silva, Finger e Melo (2017) e construir matrizes de conectivos, conforme mostra o Quadro 4.1. No canto superior esquerdo, temos a operação lógica a ser feita, no caso AND (E). Nas linhas abaixo da operação, temos a proposição "P" e os possíveis valores que ela pode assumir, ou seja, verdadeira / falsa. Nas colunas ao lado da operação, temos os valores da proposição "Q", ou seja, também verdadeira / falsa. No centro da matriz estão os possíveis resultados lógicos para a operação AND. Veja que, quando ***P* E *Q*** são verdadeiras, o resultado é V. Para todos os demais casos, o resultado é falso (F).

Quadro 4.1 | Matriz do conectivo AND

P AND Q	Q = V	Q = F
P = V	V	F
P = F	F	F

Fonte: adaptado de Silva, Finger e Melo (2017).

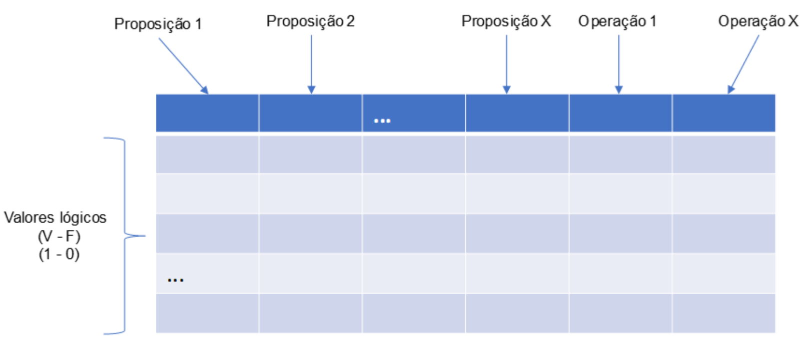
A representação dos resultados lógicos por meio de matrizes de conectores ajuda na organização, porém, limita uma operação por matriz. Como meio de organizar os resultados e facilitar a operação entre vários conectores em uma mesma estrutura, podemos utilizar a Tabela Verdade.

Construção da tabela verdade

Dada a necessidade de se obter resultados lógicos da combinação de proposições e conectores, um dos métodos mais utilizados é o método da Tabela Verdade. Por definição, a Tabela da Verdade é um método exaustivo de geração de valorações para uma dada fórmula (SILVA; FINGER; MELO, 2017). Entendemos por fórmula a composição de proposições e conectores lógicos, por exemplo, $P \vee Q$. Observe o esquema geral da Tabela Verdade na Figura 4.1. Nas colunas, colocaremos primeiro as proposições (quantas forem necessárias testar) e, em seguida, as operações lógicas das quais queremos obter os resultados. Já nas linhas, colocaremos os valores lógicos (V – F) tanto para as proposições quanto para os resultados das fórmulas que obteremos, lembrando que nosso objetivo com a tabela verdade é analisar TODOS os resultados possíveis, e podemos compará-la com um mapa de resultados.

Obtida a tabela uma vez, basta consultá-la.

Figura 4.1 | Esquema geral de uma tabela verdade



Fonte: elaborada pela autora.

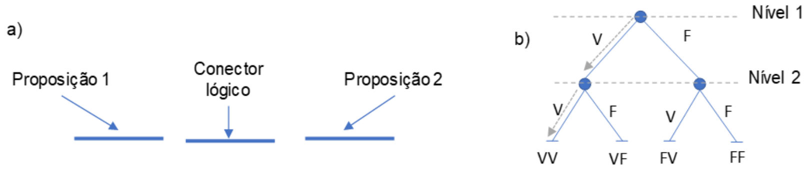
Alguns pontos sobre a lógica têm que estar bem claros para que possamos construir nossas Tabelas Verdade.

Toda proposição é binária, ou seja, só pode assumir um dos seguintes valores: Verdadeiro (V) ou Falso (F). Você pode optar por utilizar 1 para V e 0 para F.

Ao realizar uma operação lógica com duas proposições, temos que testar todas as combinações de respostas, o que influenciará diretamente a quantidade de linhas necessárias na Tabela Verdade.

O segundo ponto é muito importante e precisa ficar bem claro. Observe a Figura 4.2 (a). Criamos um esquema que mostra uma fórmula simples (operação binária); nele, temos um espaço reservado para uma proposição, um conector e outra proposição. Pois bem, sabemos que cada proposição pode assumir somente os valores V ou F (1 ou 0 se preferir). Com esses valores de entrada, essa fórmula genérica pode gerar **<u>quatro</u>** resultados distintos. Mas como isso é possível? Na verdade, estamos diante de um problema de análise combinatória, mas para facilitar nossa vida, em vez de ficarmos decorando fórmulas, vamos entender por meio da árvore de possiblidades (Figura 4.2 b). Observe a Figura 4.2 (b), cada proposição gera um nível na árvore, e como temos duas proposições, temos dois níveis. Os ramos da árvore representam as possíveis respostas para cada proposição (somente V ou F) e suas combinações. Para encontrar todas as combinações possíveis é preciso percorrer os ramos passando pelos níveis, ou seja, para esse caso, existem 4 caminhos diferentes VV, VF, FV, FF. Em outras palavras, podemos dizer que a entrada V da primeira proposição pode se combinar com as entradas V e F da segunda proposição, gerando, assim, dois resultados (o resultado depende do conector utilizado). A entrada F da primeira proposição também pode se combinar com as entradas da segunda proposição, gerando dois novos resultados. Portanto, ao final, temos quatro resultados.

Figura 4.2 | Árvore de possibilidades



Fonte: elaborada pela autora.

Assimile

É importante que você entenda a lógica da combinação dos resultados, pois a quantidade de combinações será a quantidade de linhas da Tabela Verdade. 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 $2^2 = 4$; 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 conector 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. Utilizaremos o símbolo $\wedge$ para representar esse conector lógico. Para construir a tabela verdade da conjunção, vamos considerar como entradas as proposições A e B. Queremos avaliar os resultados para a fórmula $A \wedge B$. Veja na Figura 4.3 que na primeira coluna (C1) colocamos a proposição A, na segunda coluna (C2) a proposição B e na terceira coluna (C3) a fórmula que queremos avaliar. Sobre os resultados, vamos analisar linha a linha.

Na linha 1 (L1), colocamos as proposições A e B com entrada V. Veja na coluna 3 (C3) que a saída para essas entradas também é V.

Na linha 2 (L2), colocamos a proposição A com entrada V e a B com entrada F. Veja na coluna 3 (C3) que a saída para essas entradas é F.

Na linha 3 (L3), colocamos a proposição A com entrada F e a B com entrada V. Veja na coluna 3 (C3) que a saída para essas entradas é F.

Na linha 4 (L4), colocamos as proposições A e B com entrada F. Veja na coluna 3 (C3) que a saída para essas entradas é F.

Figura 4.3 | Tabela Verdade da conjunção

	C1	C2	C3
	A	B	$A \wedge B$
L1	V	V	V
L2	V	F	F
L3	F	V	F
L4	F	F	F

Fonte: elaborada pela autora.

Reflita

Como usamos duas proposições, foram necessárias 4 linhas para gerar a valoração da fórmula $A \wedge B$. Repare na distribuição das entradas. Para a proposição A, primeiro foram dispostas as entradas V combinando com as entradas V e F da proposição B; em seguida, foram dispostas as entradas F combinando com as entradas V e F de B. Portanto, a primeira coluna ficou com entradas VVFF e a segunda coluna VFVF. Será que foi só uma coincidência ou existe uma lógica que ajuda na organização? Como ficariam dispostas as entradas da Tabela Verdade da conjunção para três proposições?

Quando utilizamos a lógica formal na construção de algoritmos, o conector lógico AND é um recurso muito valioso na tomada de decisões. O resultado dessa operação em uma linguagem de programação pode ser descrito por meio de uma Tabela Verdade. Por isso, entender esse mecanismo é fundamental para sua vida profissional.

Exemplificando

Vejamos um exemplo prático para a conjunção. Imagine que estamos criando uma aplicação que precisa informar se uma determinada pessoa irá pagar imposto ou não, a depender da sua renda, de acordo com a seguinte regra: Se o salário for superior a 5 mil e a idade menor que 40 anos, a pessoa pagará de imposto 10% do seu salário. Considere as seguintes proposições:

A: o salário é maior que R$ 5 mil.

B: a idade é menor que 40 anos.

Com base na tabela verdade da conjunção, vamos analisar qual seria o resultado da fórmula $A \wedge B$ para uma pessoa que recebe um salário de R$ 4 mil e possui 32 anos.

Avaliando a proposição A para o caso, temos um resultado F (pois não ganha salário de 5 mil). Já a proposição B possui resultado V (a idade é menor que 40 anos). Ao consultarmos a terceira linha da Figura 4.3, vemos que o resultado de $A \wedge B$ para tais entradas é falso. Portanto, para o caso analisado, o resultado da fórmula é F.

Tabela Verdade da disjunção (OR - OU)

O conector 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. Utilizaremos o símbolo $\vee$ para representar esse conector lógico. Para construir a Tabela Verdade da disjunção, vamos considerar como entradas as proposições A e B. Queremos avaliar os resultados para a fórmula $A \vee B$. Veja no Figura 4.4 que na primeira coluna colocamos a proposição A, na segunda coluna a proposição B e na terceira coluna a fórmula que queremos avaliar. Sobre os resultados, vamos analisar linha a linha.

Na linha 1 (L1), colocamos as proposições A e B com entrada V. Veja na coluna 3 (C3) que a saída para essas entradas também é V.

Na linha 2 (L2), colocamos a proposição A com entrada V e a B com entrada F. Veja na coluna 3 (C3) que a saída para essas entradas é V.

Na linha 3 (L3), colocamos a proposição A com entrada F e a B com entrada V. Veja na coluna 3 (C3) que a saída para essas entradas é V.

Na linha 4 (L4), colocamos as proposições A e B com entrada F. Veja na coluna 3 (C3) que a saída para essas entradas é F.

Figura 4.4 | Tabela Verdade com operador de negação

	C1	C2	C3
	A	B	$A \vee B$
L1	V	V	V
L2	V	F	V
L3	F	V	V
L4	F	F	F

Fonte: elaborada pela autora.

Como mostra a Tabela Verdade da disjunção, basta que uma entrada seja verdadeira para obtermos um resultado verdadeiro.

Tabela Verdade para Negação

O operador lógico de negação tem a função de inverter, seja uma entrada ou o resultado de uma operação. Utilizaremos o símbolo ← para representar esse conector lógico. Na Figura 4.5 (a), temos uma Tabela Verdade com negação para as proposições A, B. Veja que $\leftarrow A$ inverte o valor de A, ou seja, onde é V fica F e vice-versa. O mesmo acontece para $\leftarrow B$, que inverte o valor de B. Na Figura 4.5 (b), temos nas colunas 3 (C3) e 5 (C5) os resultados já conhecidos da conjunção e disjunção, e nas colunas 4 (C4) e 6 (C6) os novos resultados, sendo na coluna 4 (C4) os resultados para a negação da conjunção e na coluna 6 (C¨) a negação da disjunção. Repare nos parênteses, que são obrigatórios, indicando que a negação é para toda a operação e não somente para uma proposição.

Figura 4.5 | Tabela Verdade com operador de negação

a)

A	B	$\neg A$	$\neg B$
V	V	F	F
F	F	V	V

b)

	C1	C2	C3	C4	C5	C6
	A	B	$A \wedge B$	$\neg(A \wedge B)$	$A \vee B$	$\neg(A \vee B)$
L1	V	V	V	F	V	F
L2	V	F	F	V	V	F
L3	F	V	F	V	V	F
L4	F	F	F	V	F	V

Fonte: elaborada pela autora.

Para ficar claro como os parênteses podem afetar o resultado da operação lógica envolvendo a negação, observe a Figura 4.6. Na coluna 4 (C4), temos os resultados para a fórmula $\neg(A \wedge B)$, já na coluna 5 (C5) temos o resultado para a fórmula $(A \wedge \neg B)$. Veja como é diferente. No primeiro caso estamos invertendo o resultado da operação toda, já no segundo caso, estamos invertendo apenas o valor de B e fazendo a conjunção com A.

Figura 4.6 | Tabela Verdade com operador de negação

	C1	C2	C3	C4	C5
	A	B	$\neg B$	$\neg(A \wedge B)$	$(A \wedge \neg B)$
L1	V	V	F	F	F
L2	V	F	V	V	V
L3	F	V	F	V	F
L4	F	F	V	V	F

Fonte: elaborada pela autora.

Como você já deve ter percebido, não adianta tentar decorar os resultados, você precisa entender os operadores básicos e, a partir daí, ir resolvendo as fórmulas, parte por parte, com auxílio da Tabela Verdade.

Sem medo de errar

Caro estudante, chegou o momento de solucionarmos o desafio.

Como membro da equipe de *analytics* de uma empresa de varejo, dadas as seguintes proposições: p: o cliente é do sexo feminino e q: o cliente tem idade entre 20 e 30 anos, você foi encarregado de construir uma Tabela Verdade para as operações de conjunção e disjunção, além de criar a negação para as fórmulas. Com a Tabela Verdade criada, você deve avaliar os registros de clientes que foi lhe passado na Tabela 4.1, completando as colunas E/OU com V ou F.

A Tabela Verdade é um mecanismo que permite valorar fórmulas de forma genérica a partir de entradas binárias e conectores lógicos. Pois bem, como o problema proposto apresenta duas proposições, serão necessárias 4 linhas para contemplar todas as combinações possíveis das entradas. Além disso, serão necessárias 6 colunas, sendo 2 para as proposições (p, q), uma para a fórmula da conjunção, outra para a disjunção, outra para a negação da conjunção e uma última com a negação da disjunção. O resultado da Tabela Verdade deve estar conforme o Quadro 4.2.

Quadro 4.2 | Tabela Verdade para time de *analytics*

p	q	$p \wedge q$	$p \vee q$	$\neg(p \wedge q)$	$\neg(p \vee q)$
V	V	V	V	F	F
V	F	F	V	V	F
F	V	F	V	V	F
F	F	F	F	V	V

Fonte: elaborado pela autora.

A Tabela Verdade pode ser usada como um gabarito para as operações lógicas, pois contempla todas as entradas possíveis e suas combinações para as fórmulas em estudo. Com esse gabarito em mãos, podemos passar para a segunda etapa do desafio, que é fazer a valoração das fórmulas $p \wedge q$ e $p \vee q$ para cada registro da base de clientes. Pois bem, vejamos na Tabela 4.2, como ficaram os resultados.

Na linha 1, o cliente Karly é do sexo feminino, portanto, p é verdadeiro, e tem 40 anos, logo, a proposição q é falsa para esse cliente. Nesse caso, ao consultar a Tabela Verdade, a conjunção com entradas $V \wedge F$ tem como resultado F, mas a disjunção tem resultado V, pois basta que uma proposição seja V.

Na linha 2, o cliente é do sexo masculino e possui 49 anos; nesse caso, tanto p quanto q são falsas, logo, ambas fórmulas são valoradas como F.

Termine de analisar suas respostas comparando os resultados com a Tabela Verdade.

Tabela 4.2 | Valoração das fórmulas $p \wedge q$ e $p \vee q$

linha	codi-go_cli	nome_cli	gene-ro_cli	idade_cli	valor_compra	E	OU
1	53682	Karly Dillon	F	40	74,84	F	V
2	58246	Channing Vazquez	M	49	98,04	F	F
3	27022	Adria Key	F	47	65,93	F	V
4	82075	Ella Nelson	F	34	94,01	F	V
5	90657	Arden Battle	M	48	21,73	F	F
6	80330	Brittany Ramirez	F	38	42,23	F	V
7	53989	Moses Graham	M	42	37,20	F	F
8	61370	Jin Fuller	M	49	65,60	F	F
9	41807	Phelan Blair	M	46	77,40	F	F
10	94269	Porter West	M	22	67,19	F	V

Fonte: elaborada pela autora.

Nossa maior lição com esse desafio é entender como uma questão de lógica formal pode ser utilizada em um algoritmo, e isso, no mercado de trabalho, fornece resultados para as mais diversas áreas de uma empresa. Se fornecêssemos para a equipe de vendas o resultado da coluna da conjunção, a fim de comunicá-los sobre o lançamento de uma nova promoção, nenhum desses 10 receberia tal comunicado, por outro lado, se nosso resultado fosse baseado na disjunção, vários deles receberiam o comunicado.

É isso aí, você como um profissional deve ter consciência das suas responsabilidades e tomar as melhores decisões para a empresa!

Faça valer a pena

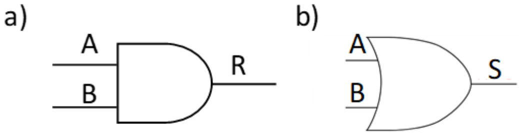
1.

Um circuito digital combinacional pode ser desenvolvido com portas lógicas. Portas lógicas são circuitos eletrônicos que representam as funções lógicas por meio de entradas e saídas. As portas lógicas têm sua representação simbó-

> lica para facilitar o desenvolvimento de projetos digitais e seguem a mesma definição da lógica formal. (DACHI; HAUPT, 2018, p. 35)

A figura seguinte apresenta duas portas lógicas, sendo, respectivamente, as portas AND e OR. Ambas as portas ilustradas possuem duas entradas A e B e uma única saída para cada R e S. As entradas A e B podem receber os valores 1 ou 0 (verdadeiro ou falso) e as saídas R e S também serão 1 ou 0.

Portas lógicas AND e OR



Fonte: elaborada pela autora.

Escolha a seguir a opção que representa a sequência correta dos possíveis resultados a serem obtidos pelas portas, por meio da Tabela Verdade, para as saídas R e S considerando, respectivamente, as seguintes entradas:

1. A = V; B = V.
2. A = V; B = F.
3. A = F; B = V.
4. A = F; B = F.

a. R = V F F F; S = V V V F.

b. R = V V F F; S = F V V F.

c. R = F F F V; S = F V V F.

d. R = F V V F; S = V F F F.

e. R = V F V V; S = V V F F.

2. Segundo Gersting (2017), os conectivos lógicos podem ser classificados em binários e unários. Os conectivos binários juntam duas expressões por meio de um conectivo lógico, produzindo uma terceira expressão. Já os conectivos unários agem em uma única expressão para produzir uma segunda expressão. A conjunção e a disjunção são binárias, já o conectivo de negação é do tipo unário.

Considerando a Tabela Verdade a seguir, complete a coluna com a fórmula $\neg(\neg A \wedge B)$:

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

Assinale a alternativa que apresenta a sequência correta.

a. F – F – V – F.

b. V – F – F – F.

c. V – V – F – V.

d. F – F – F – V.

e. V – F – V – F.

3.

> 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 da satisfatibilidade e validade de fórmulas é o método da Tabela da Verdade. (SILVA; FINGER; MELO, 2017, p. 13)

Com base em seu conhecimento de Tabela Verdade, analise as afirmativas a seguir:

I. A quantidade de linhas necessárias para valorar as possíveis respostas em uma Tabela Verdade aumenta exponencialmente com a quantidade de proposições de acordo com a regra n^2 .

II. As respostas obtidas pela fórmula $\neg(A \wedge B)$ são iguais às respostas obtidas pela fórmula $(\neg A \vee \neg B)$.

III. As respostas obtidas pela fórmula $\neg A \wedge B$ são iguais às respostas obtidas pela fórmula $A \wedge \neg B$.

É correto o que se afirma apenas em:

a. I.

b. II.

c. III.

d. I e II.

e. II e III.

Resultados na Tabela Verdade

Diálogo aberto

Caro estudante, profissionais das mais diversas áreas tomam decisões o tempo todo em seu ambiente de trabalho. Todas as decisões precisam ser bem avaliadas, pois toda ação gera uma consequência. Por exemplo, em uma campanha promocional, se optar por desconto a um determinado grupo, poderá não vender tanto para um outro grupo potencial. No mundo da lógica computacional, as decisões são tomadas por meio de estruturas de decisão, que têm sua origem em um conector lógico chamado de implicação, o qual veremos sua tabela verdade nesta seção.

Como funcionário *trainee* na área de *analytics* de uma empresa de varejo, você deve ajudar a equipe de marketing em uma campanha para o dia internacional da mulher.

Dadas as proposições:

A: o cliente é do sexo feminino.

B: o cliente fez um compra com valor superior a R$ 50,00.

C: ganhar cupom com 10% de desconto.

Seu desafio consiste primeiro em avaliar a fórmula $A \wedge B$ para cada um dos registros da Tabela 4.3. Essa avaliação lhe permitirá classificar a proposição C para cada um dos clientes, ou seja, se o cliente ganhará ou não o cupom de 10% de desconto.

Após a classificação, você deverá generalizar, por meio de uma Tabela Verdade, as possíveis respostas para a fórmula $P \rightarrow Q$, sendo P e Q duas proposições genéricas.

Tabela 4.3 | Dados de compra dos clientes

codigo_cli	nome_cli	genero_cli	idade_cli	valor_compra	cupom_10
53682	Karly Dillon	F	40	74,84	?
58246	Channing Vazquez	M	49	98,04	?
27022	Adria Key	F	47	65,93	?
82075	Ella Nelson	F	34	94,01	?
90657	Arden Battle	M	48	21,73	?

80330	Brittany Ramirez	F	38	42,23	?
53989	Moses Graham	M	42	37,20	?
61370	Jin Fuller	M	49	65,60	?
41807	Phelan Blair	M	46	77,40	?
94269	Porter West	M	22	67,19	?
56516	Zena Skinner	F	54	73,98	?
38904	Teagan Rios	M	34	61,57	?

Fonte: elaborada pela autora.

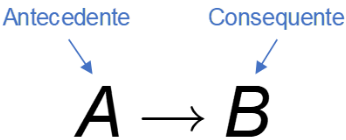
Para cumprir seu desafio, nesta seção veremos a Tabela Verdade do conector de implicação, bem como outros importantes resultados da Tabela Verdade.

Pronto para o desafio?!

Não pode faltar

A Tabela Verdade é utilizada como um método exaustivo de extração de resultados (SILVA; FINGER; MELO, 2017). Em outras palavras, construímos uma Tabela Verdade para testarmos todos os resultados possíveis para todas as combinações possíveis de entradas em uma determinada fórmula. Uma fórmula é composta por proposições e operadores lógicos, como, 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 $\rightarrow$, e significa que se a proposição 1 é verdadeira, implicará na 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. A primeira parte, antes do conector, é chamada de antecedente, e a segunda de consequente conforme ilustra a Figura 4.7.

Figura 4.7 | Implicação lógica



Fonte: elaborada pela autora.

Para entendermos como funciona a implicação, vejamos alguns exemplos.

Considere as proposições *A e B*:

A: há uma falha na rede elétrica.

B: a chave central irá desligar.

A fórmula $A \rightarrow B$, que significa que B está condicionado a ***A***, deve ser lida como: "se houver uma falha na rede elétrica, então, a chave central irá desligar."

Considere as proposições P, Q, R.

P: a nota mínima necessária para ser aprovado é 6,0.

Q: João tirou 8,0 na prova.

R: João será aprovado.

Nesse exemplo, temos o resultado de uma conjunção implicando uma terceira proposição. Simbolicamente, escrevemos $(P \wedge Q) \rightarrow R$ e deve ser lida como: "Se a nota mínima necessária para ser aprovado é 6,0 e João tirou 8,0 na prova, então, ele será aprovado."

Os possíveis resultados do operador condicional estão representados na Tabela Verdade apresentada na Figura 4.8.

Figura 4.8 | Tabela Verdade para o condicional

	C1	C2	C3
	A	B	$A \rightarrow B$
L1	V	V	V
L2	V	F	F
L3	F	V	V
L4	F	F	V

Fonte: elaborada pela autora.

Para entendermos os resultados da implicação, vamos utilizar o exemplo presente 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 $A \rightarrow B$ 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 da Figura 4.8. Na primeira linha (L1), temos como entrada a verdade das proposições A e B, em nosso exemplo, quer dizer que a pedra foi solta (proposição A é verdadeira) e caiu (proposição B é verdadeira), portanto, a condição era verdadeira e o resultado é **V** (linha 1 e coluna 3). Na linha dois (L2), temos como entrada que a proposição A é verdadeira e a proposição B é falsa, isso quer dizer que a pedra foi solta, mas não caiu. Nesse caso, a condicional não é verdadeira e o resultado é F (linha 2 e coluna 3). Nas demais linhas, terceira e quarta (L3 e L4), temos como entrada que a proposição A é falsa (que é o antecedente), nesse caso, não há como avaliar a condicional e o resultado é tomado como verdadeiro.

Assimile

Os resultados lógicos das linhas 3 e 4 (L3 e L4) da Tabela Verdade da condicional não são tão fáceis de identificar. Como se trata de uma dependência, caso o antecedente seja falso, o resultado lógico será sempre verdadeiro.

“Por convenção, $A \rightarrow B$ será considerada verdadeira se A for falsa, independentemente do valor lógico de B” (GERSTING, 2017, p. 2).

Tautologia, contradição e contingência

Considerando a proposição A como:

A: hoje está chovendo.

Vamos construir a Tabela Verdade para a fórmula $A \vee \neg A$, que, traduzindo, quer dizer, “hoje está chovendo ou hoje não está chovendo”. Veja o resultado no Quadro 4.3.

Quadro 4.3 | Tabela Verdade da fórmula $A \vee \neg A$

A	$\leftarrow A$	$A \vee \neg A$
V	F	V
F	V	V

Fonte: elaborado pela autora.

Como podemos observar no Quadro 4.3, a coluna dos resultados (última coluna) para a fórmula obteve como resposta somente verdadeiro. Quando o resultado de uma fórmula obtém somente V como resposta, a fórmula é denominada **tautologia** (GERSTING, 2017).

Agora, vamos considerar a seguinte proposição B:

B: hoje é segunda-feira.

Vamos construir a Tabela Verdade para a fórmula $B \wedge \neg B$, que quer dizer, “Hoje é segunda-feira e hoje não é segunda-feira”. Veja o resultado na Quadro 4.4.

Quadro 4.4 | Tabela Verdade da fórmula $B \wedge \neg B$

B	$\leftarrow B$	$B \wedge \neg B$
V	F	F
F	V	F

Fonte: elaborado pela autora.

Como podemos observar no Quadro 4.4, a coluna dos resultados (última coluna da tabela verdade) para a fórmula obteve como resposta somente falso. Quando o resultado de uma fórmula obtém somente F como resposta, a fórmula é denominada **contradição** (GERSTING, 2017).

Quando uma tabela verdade não é uma tautologia e não é uma contradição, então, ela é uma **contingência**.

Reflita

A tautologia é uma proposição em que, independentemente das entradas, todas as respostas são verdadeiras. Já a contradição é o resultado quando todas as possíveis respostas são falsas.

Podemos afirmar que a tautologia acontecerá somente para as fórmulas com conectores lógicos de disjunção (OR) e a contradição para os conectores de conjunção (AND)?

Equivalência aplicada na tabela verdade

Considere as seguintes proposições:

A: o cliente tem 35 anos.

B: o cliente gastou mais do que R$ 100,00 na última compra.

Dadas as proposições A e B, vamos construir a Tabela Verdade (Quadro 4.5) para as seguintes fórmulas: $A \vee B$ e $B \vee A$.

Quadro 4.5 | Tabela Verdade das fórmulas $A \vee B$ e $B \vee A$

A	B	$A \vee B$	$B \vee A$	$A \vee B \Leftrightarrow B \vee A$
V	V	V	V	V
V	F	V	V	V
F	V	V	V	V
F	F	F	F	V

Fonte: elaborado pela autora.

As colunas 3 e 4 do Quadro 4.5 apresentam os possíveis resultados para as fórmulas $A \vee B$ e $B \vee A$. Na quinta coluna, temos mais um teste lógico chamado de **equivalência**, no qual testamos se os resultados obtidos para a fórmula $A \vee B$ são iguais (equivalentes) aos obtidos por $B \vee A$. Veja que usamos o símbolo $\Leftrightarrow$ para denotar essa operação e que o resultado foi uma tautologia, o que nos permite concluir que as fórmulas $A \vee B$ e $B \vee A$ são equivalentes.

Exemplificando

Para saber se duas fórmulas são equivalentes, é necessário construir a tabela verdade e verificar se a equivalência é uma tautologia. Por exemplo, vamos construir uma tabela verdade (Quadro 4.6) para testar se as fórmulas $A \wedge B$ e $B \wedge A$ são equivalentes.

Quadro 4.6 | Tabela verdade das fórmulas $A \wedge B$ e $B \wedge A$

A	B	$A \wedge B$	$B \wedge A$	$A \wedge B \Leftrightarrow B \wedge A$
V	V	V	V	V
V	F	F	F	V
F	V	F	F	V
F	F	F	F	V

Fonte: elaborado pela autora.

Como podemos observar no Quadro 4.6, as fórmulas $A \wedge B$ e $B \wedge A$ também são equivalentes.

Os resultados obtidos nos quadros 4.5 e 4.6 não são uma coincidência, pois estamos diante de uma forte propriedade, a comutativa. Lembra das aulas de matemática quando aprendeu que o resultado de 2 + 3 é igual a 3 + 2? Pois bem, aqui na lógica também temos essa mesma propriedade: a ordem dos fatores não altera o resultado. Veja no Quadro 4.7 mais algumas propriedades.

Quadro 4.7 | Equivalências tautológicas

1	$A \vee B \Leftrightarrow B \vee A$	$A \wedge B \Leftrightarrow B \wedge A$	Comutatividade
2	$(A \vee B) \vee C \Leftrightarrow A \vee (B \vee C)$	$(A \vee B) \vee C \Leftrightarrow A \vee (B \vee C)$	Associatividade
3	$A \vee (B \wedge C) \Leftrightarrow (A \vee B) \wedge (A \vee C)$	$A \wedge (B \vee C) \Leftrightarrow (A \wedge B) \vee (A \wedge C)$	Distributividade

Fonte: adaptado de Gersting (2017).

Já vimos as tabelas verdade para a propriedade da comutatividade, vejamos agora para as demais, começando pela associativa $(A \vee B) \vee C \Leftrightarrow A \vee (B \vee C)$. O resultado está no Quadro 4.8. O primeiro detalhe é que, como temos três proposições envolvidas, vamos precisar de 8 linhas para representar todas as combinações possíveis (lembra, linhas = 2^n). Outro detalhe é que para facilitar, resolvemos a fórmula em partes. Veja: na coluna C4, extraímos os resultados para $A \vee B$, e na coluna C5 utilizamos o resultado de C4 para fazer a disjunção com a proposição C, obtendo o resultado do lado esquerdo da fórmula. O mesmo fizemos para o lado direito, na coluna C6, fizemos $B \vee C$ (como estava entre parênteses, fizemos primeiro) e depois, na coluna C7, utilizamos o resultado de C6 para fazer a disjunção com a proposição A. Por fim, na coluna C8, fizemos a equivalência verificando se os resultados obtidos em C5 e C7 eram iguais, como temos uma tautologia, então, podemos afirmar que elas de fato são equivalentes.

Quadro 4.8 | Tabela Verdade para propriedade associativa $(A \vee B) \vee C \Leftrightarrow A \vee (B \vee C)$

C1	C2	C3	C4	C5	C6	C7	C8
A	B	C	$A \vee B$	$(A \vee B) \vee C$	$B \vee C$	$A \vee (B \vee C)$	$\Leftrightarrow$
V	V	V	V	V	V	V	V
V	V	F	V	V	V	V	V
V	F	V	V	V	V	V	V
V	F	F	V	V	F	V	V
F	V	V	V	V	V	V	V
F	V	F	V	V	V	V	V
F	F	V	F	V	V	V	V
F	F	F	F	F	F	F	V

Fonte: elaborada pela autora.

Vamos agora verificar uma equivalência da propriedade da distributividade para a fórmula $A \wedge (B \vee C) \Leftrightarrow (A \wedge B) \vee (A \wedge C)$. O Quadro 4.9 apresenta o resultado. Resolvemos primeiro o lado esquerdo da equivalência, ou seja, a fórmula $A \wedge (B \vee C)$. Na coluna C4 fizemos a disjunção entre parênteses e na coluna C5 utilizamos o resultado obtido em C4 para fazer a conjunção com a proposição A. Nas colunas C6 e C7, resolvemos os parênteses da fórmula à direita da equivalência, e na C8, usamos os resultados de C6 e C7 para a disjunção. Por fim, na C9 comparamos os resultados obtidos em C5 e C8, provando que se trata de uma equivalência.

Quadro 4.9 | Tabela verdade para propriedade da distributividade $A \wedge (B \vee C) \Leftrightarrow (A \wedge B) \vee (A \wedge C)$

C1	C2	C3	C4	C5	C6	C7	C8	C9
A	B	C	$B \vee C$	$A \wedge (B \vee C)$	$A \wedge B$	$A \wedge C$	$(A \wedge B) \vee (A \wedge C)$	$\Leftrightarrow$
V	V	V	V	V	V	V	V	V
V	V	F	V	V	V	F	V	V
V	F	V	V	V	F	V	V	V
V	F	F	F	F	F	F	F	V
F	V	V	V	F	F	F	F	V
F	V	F	V	F	F	F	F	V
F	F	V	V	F	F	F	F	V
F	F	F	F	F	F	F	F	V

Fonte: elaborada pela autora.

Não poderíamos encerrar esta seção sem mencionar, além das propriedades, outros dois importantes resultados da equivalência, usados para fazer a negação de uma proposição composta. Refiro-me às leis de De Morgan, "[...] *assim nomeadas em honra ao matemático inglês do século XIX Augustus De Morgan, o primeiro a enunciá-las*" (GERSTING, 2017, p. 9). As duas equivalências são:

I - $\neg(A \vee B) \Leftrightarrow \neg A \wedge \neg B$

II - $\neg(A \wedge B) \Leftrightarrow \neg A \vee \neg B$

Observe que a primeira equivalência (I) trata da equivalência envolvendo a negação em uma disjunção e a segunda equivalência (II) corresponde a uma negação em uma conjunção. O Quadro 4.10 apresenta os resultados para a fórmula I e o Quadro 4.11 para a fórmula II.

Quadro 4.10 | Tabela verdade para a lei de De Morgan $\neg(A \vee B) \Leftrightarrow \neg A \wedge \neg B$

A	B	$\neg(A \vee B)$	$\neg A \wedge \neg B$	$\Leftrightarrow$
V	V	F	F	V
V	F	F	F	V
F	V	F	F	V
F	F	V	V	V

Fonte: elaborado pela autora.

Quadro 4.11 | Tabela verdade para a lei de De Morgan $\neg(A \wedge B) \Leftrightarrow \neg A \vee \neg B$

A	B	$\neg(A \wedge B)$	$\neg A \vee \neg B$	$\Leftrightarrow$
V	V	F	F	V
V	F	V	V	V
F	V	V	V	V
F	F	V	V	V

Fonte: elaborado pela autora.

O que podemos concluir das leis de De Morgan é que a negação de uma disjunção é equivalente à negação de cada uma das proposições em uma conjunção (fórmula I) e que a negação de uma conjunção é equivalente à negação de cada uma das proposições em uma disjunção (fórmula II).

E com esses importantes resultados, finalizamos nossa seção!

Até a próxima!

Sem medo de errar

Você foi encarregado da missão de direcionar a equipe de marketing em uma campanha para o dia internacional da mulher.

Dada as proposições:

A: o cliente é do sexo feminino.

B: o cliente fez um compra com valor superior a R$ 50,00.

C: ganhar cupom com 10% de desconto.

Você deve primeiro avaliar a fórmula $A \wedge B$ para cada um dos registros da Tabela 4.3 classificando a proposição C, como V ou F, para cada um dos clientes, ou seja, se o cliente ganhará ou não o cupom de 10% de desconto.

Pois bem, vamos analisar o primeiro registro:

A: O cliente é do sexo feminino. (SIM – V)

B: O cliente fez um compra com valor superior a R$ 50,00. (SIM – V)

Portanto, para o primeiro registro a fórmula $A \wedge B$ resulta em V, pois $V \wedge V = V$, então a proposição C é V.

Já para o segundo registro, temos $F \wedge V = F$, pois o cliente é do sexo masculino. Então a proposição C é falsa para esse caso.

Ao analisar todos os registros, você deve chegar ao resultado da Tabela 4.4.

Tabela 4.4 | Resultado para equipe de marketing

linha	codigo_cli	nome_cli	genero_cli	idade_cli	valor_ compra	cupom_10
1	53682	Karly Dillon	F	40	74,84	V
2	58246	Channing Vazquez	M	49	98,04	F
3	27022	Adria Key	F	47	65,93	V
4	82075	Ella Nelson	F	34	94,01	V
5	90657	Arden Battle	M	48	21,73	F
6	80330	Brittany Ra-mirez	F	38	42,23	F
7	53989	Moses Graham	M	42	37,20	F
8	61370	Jin Fuller	M	49	65,60	F
9	41807	Phelan Blair	M	46	77,40	F
10	94269	Porter West	M	22	67,19	F
11	56516	Zena Skinner	F	54	73,98	V
12	38904	Teagan Rios	M	34	61,57	F

Fonte: elaborada pela autora.

Agora, vamos generalizar as repostas construindo a Tabela verdade para a fórmula $P \rightarrow Q$, como mostra o Quadro 4.12.

Quadro 4.12 | Tabela verdade para a fórmula $P \rightarrow Q$

P	Q	$P \rightarrow Q$
V	V	V
V	F	F
F	V	V
F	F	V

Fonte: elaborado pela autora.

Para finalizar, vamos fazer a conexão entre o resultado obtido na análise dos registros com o Quadro 4.12.

A primeira observação importante é que a proposição genérica P no desafio é o resultado da conjunção $A \wedge B$. Vamos fazer a análise com os dois primeiros registros.

No primeiro registro, a fórmula $A \wedge B$ resultado em V e a proposição C também foi classificada como V. Temos aqui o caso $V \rightarrow V$, que, consultado a tabela verdade da implicação tem como resultado também V, isso quer dizer que a proposição "O cliente é do sexo feminino e fez uma compra acima de R$ 50,00, então, ele ganhará desconto" é uma verdade.

Já para o segundo registro, temos $A \wedge B = F$, consequentemente a proposição C também é falsa. Temos aqui o caso $F \rightarrow F$, que consultado a tabela verdade da implicação, tem como resultado V, mas o que esse resultado quer dizer? Quer dizer que a proposição "O cliente não é do sexo feminino e nem fez uma compra acima de R$ 50,00, então, ele não ganhará desconto" é uma verdade.

Termine de fazer as avaliações, caso a caso, pois quanto mais praticar, mais apto estará para avaliar as implicações lógicas e contribuir ainda mais com seu time!

Faça valer a pena

1. "Dizemos que uma fórmula B é consequência lógica de outra fórmula A, se toda valoração v que satisfaz A também satisfaz B. Note que esta definição permite que B seja satisfeito por valorações que não satisfazem A. Neste caso, também dizemos que A implica logicamente B" (SILVA; FINGER; MELO, 2017, p. 19).

Dada as proposições:

A: o computador desliga abruptamente.

B: os processos finalizam.

Escolha a fórmula que representa, simbolicamente, a sentença: “se o computador desligar abruptamente, então os processos finalizarão.” e a correta sequência de resultados para as entradas: VF, FV, VV, FF.

a. $A \to B$; F V V V.

b. $A \wedge B$; V F V V.

c. $A \to B$; V F V V.

d. $A \vee B$; F V V V.

e. $A \to B$; V F V F.

2. A tabela verdade é utilizada como um método exaustivo de extração de resultados (SILVA; FINGER; MELO, 2017). Em outras palavras, uma tabela verdade deve ser construída para testar todos os resultados possíveis dada uma combinação de entradas e uma fórmula.

Dada a fórmula $A \to B$ e as proposições:

A: o cliente fez uma compra acima de R$ 100,00.

B: o cliente ganhará um cupom de desconto.

Escolha a sentença que traduz corretamente, em linguagem natural, o resultado falso da fórmula.

a. Se o cliente fizer uma compra acima de R$ 50,00, ele ganhará um cupom de desconto.

b. Se o cliente fizer uma compra acima de R$ 50,00, ele não ganhará um cupom de desconto.

c. Se o cliente não fizer uma compra acima de R$ 50,00, ele ganhará um cupom de desconto.

d. Se o cliente não fizer uma compra acima de R$ 50,00, ele não ganhará um cupom de desconto.

e. Se o cliente fizer uma compra acima de R$ 50,00 e ganhar um cupom, ele terá um desconto.

3. "Uma tautologia é "intrinsecamente verdadeira" pela sua própria estrutura; ela é verdadeira independentemente dos valores lógicos atribuídos às suas letras de proposição" (GERSTING, 2017, p. 17). A tautologia é um resultado usado para verificar a equivalência de duas fórmulas.

Considerando a equivalência lógica, analise as afirmativas a seguir.

I. As fórmulas $A \vee B$ e $B \vee A$ são equivalentes, pois se trata da propriedade da comutatividade.

II. As fórmulas $A \vee (B \wedge C)$ e $(A \wedge B) \vee (A \wedge C)$ são equivalentes, pois se trata da propriedade da distributividade.

III. As fórmulas $\neg(A \vee B)$ e $\neg A \vee \neg B$ são equivalentes, pois se trata de uma das leis de De Morgan.

É correto o que se afirma em:

a. I, apenas.

b. II e III, apenas.

c. I e III, apenas.

d. II, apenas.

e. I, II e III.

Aplicações Tabela Verdade

Diálogo aberto

Caro estudante, até você chegar ao ensino superior foram anos de estudos, não é mesmo? Tudo começou com o processo de alfabetização, tanto de leitura quanto das operações básicas de matemática (soma, subtração, divisão e multiplicação). Primeiro você aprendeu a fazer operações matemáticas simples, por exemplo, 2 + 3, depois apareceram os parênteses, por exemplo, $(2+3)\times 4$. Mas será que esses parênteses são só de enfeite? Certamente que não! Eles influenciam a ordem que as operações devem ser resolvidas e, como você já deve saber, não seguir as regras certamente implica em chegar em resultados errôneos. Pois bem, da mesma forma que a matemática tem suas fórmulas e suas regras de resolução, a lógica formal também as possui, e nesta seção vamos aprender mais sobre elas.

Como funcionário trainee na área de *analytics* de uma empresa de varejo, você deve dar continuidade em seu trabalho, fornecendo novos insights para a equipe de marketing realizar sua campanha promocional. Para esse novo desafio foi enviada a você uma base com novas informações, conforme ilustra a Tabela 4.5. Nessa base é possível encontrar o valor gasto na última compra do cliente, o total de compras já feito por ele e o ticket médio (valor médio gasto em cada compra). A partir desses dados você deve usar as regras da lógica para classificar se o cliente tem potencial para comprar na nova campanha e, se tiver, então ele ganhará um cupom com desconto de 10%. Caso não seja um cliente com potencial então ele ganhará somente um cupom com 5%. Vamos às regras: para ser classificado como um cliente com potencial de compra, não importa o gênero (pode ser feminino ou masculino), o cliente deve ter idade entre 30 e 45 anos, ter feito acima de 10 compras e ter um ticket médio acima de R$ 50,00. Seu desafio é montar uma fórmula que traduza essa regra e, então preencher a coluna "cliente_potencial" com o resultado da fórmula para cada registro. Dada a classificação, você deve escrever uma nova fórmula que traduza "Se o cliente tem potencial de compra, então ele deve ganhar um cupom com 10% de desconto", e outra fórmula que traduza "Se é falso que o cliente tem potencial de compra, então ele deve ganhar um cupom com 5% de desconto". Por fim, use a lógica de programação para preencher as colunas "cupom_10" e "cupom_5" valorando as condicionais.

Tabela 4.5 | Dados de compra dos clientes

codigo_cli	nome_cli	genero_cli	idade_cli	valor_ultima_compra	total_compras	ticket_medio	cliente_potencial	cupom _10	cupom _5
53682	Karly Dillon	F	40	74,84	5	45,00	?	?	?
58246	Channing Vazquez	M	49	98,04	20	200,00	?	?	?
27022	Adria Key	F	47	65,93	12	34,00	?	?	?
82075	Ella Nelson	F	34	94,01	16	150,00	?	?	?
90657	Arden Battle	M	48	21,73	4	23,00	?	?	?
80330	Brittany Ramirez	F	38	42,23	1	42,23	?	?	?
53989	Moses Graham	M	42	37,20	29	45,00	?	?	?
61370	Jin Fuller	M	31	86,00	35	123,00	?	?	?
41807	Phelan Blair	M	45	77,40	23	95,00	?	?	?
94269	Porter West	M	22	67,19	6	35,00	?	?	?
56516	Zena Skinner	F	54	73,98	15	60,00	?	?	?
38904	Teagan Rios	M	34	61,57	17	71,00	?	?	?

Fonte: elaborada pela autora.

Para cumprir esse desafio, nesta seção você aprenderá a importância dos parênteses nas fórmulas, bem como seguir a regra de precedência dos operadores. Você também verá um caso de como fazer a ligação entre a lógica formal e a lógica na computação, assim já vai se preparando para em breve começar a programar.

Então vamos lá!

Não pode faltar

Ao longo de nossos estudos aprendemos diversas fórmulas; quem não se lembra da famosa fórmula de Bhaskara usada para encontrar as raízes de uma equação do segundo grau? Ou então da segunda lei Newton, que diz que a força é sempre diretamente proporcional ao produto da aceleração de um corpo pela sua massa? Para conseguimos resolver esses tipos de equação, primeiro foi necessário aprender as operações matemáticas básicas (soma, subtração, multiplicação e divisão); depois essas operações passaram a se combinar em fórmulas mais elaboradas, e nesse momento foi necessário aprender a ordem de precedência dos operadores. Por exemplo: as fórmulas: (i) $2+3\times4$ e (ii) $(2+3)\times4$ apresentam o mesmo resultado? A resposta é não, pois a primeira tem como resultado 14 e a segunda, 20. Isso acontece porque, em uma fórmula, a multiplicação tem precedência sobre adição, então ao não usar parênteses na fórmula (i) a multiplicação foi feita antes da soma. Outro detalhe importante é que as fórmulas apresentam uma sintaxe correta, por exemplo: não podemos escrever 2 ++ 3, pois essa fórmula é inválida.

Assim como as fórmulas matemáticas, podemos construir expressões lógicas mais complexas a partir da combinação das proposições, dos conectivos e dos parênteses (GERSTING, 2017). Da mesma forma que as operações matemáticas têm ordem de precedência, os conectivos lógicos também a possui.

Assimile

Para resolver uma expressão lógica que combina várias proposições com conectivos lógicos é preciso obedecer a seguinte regra de precedência:

1. Para expressões que tenham parênteses, primeiro efetuam-se as operações lógicas dentro dos parênteses mais internos.
2. $\leftarrow$ (Negação) (maior precedência).
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 $A \vee \neg B$, uma vez que, de acordo com a ordem de precedência, a negação será realizada primeiro, mesmo sem parênteses.

Exemplificando

Para exemplificar como a ordem de precedência dos operadores lógicos pode influenciar o resultado, vamos construir uma Tabela Verdade para as fórmulas $A \wedge B \rightarrow A$ e $A \vee (B \rightarrow A)$. Veja no Quadro 4.13 que os resultados das colunas C4 e C6 são diferentes. Em C4, como o operador de conjunção tem precedência sobre a implicação, chegamos em uma tautologia, o que não ocorreu em C6 quando forçamos, por meio de parênteses, a implicação ser efetuada primeiro.

Quadro 4.13 | Tabela Verdade das fórmulas $A \wedge B \rightarrow A$ e $A \vee (B \rightarrow A)$

C1	C2	C3	C4	C5	C6
A	B	$A \wedge B$	$A \wedge B \rightarrow A$	$B \rightarrow A$	$A \wedge (B \rightarrow A)$
V	V	V	V	V	V
V	F	F	V	V	V
F	V	F	V	F	F
F	F	F	V	V	F

Fonte: elaborado pela autora.

Ainda no Quadro 4.13, observe que usamos a coluna C3 para efetuar a primeira operação da fórmula $A \wedge B \rightarrow A$, e na coluna C4 usamos o resultado obtido em C3 para fazer a implicação. O mesmo acontece na coluna C5, em que fazemos a primeira operação da fórmula $A \vee (B \rightarrow A)$, (considerando as regras da ordem de precedência) e depois usamos o resultado de C5 para fazer a conjunção final em C6. Esse processo de criar uma coluna para cada operação facilita o trabalho e nos auxilia a não cometer erros na construção da Tabela Verdade. Esses resultados intermediários podem ser representados por novas letras, por exemplo: poderíamos chamar o resultado da coluna C3 de P, então na coluna C4 teríamos $P \rightarrow A$. Da mesma forma, poderíamos chamar o resultado da coluna C5 de R, então na coluna C6 teríamos a fórmula $A \wedge R$. O Quadro 4.14 mostra essa alternativa.

Quadro 4.14 | Tabela Verdade com proposições intermediárias

		P		R	
A	B	$A \wedge B$	$P \rightarrow A$	$B \rightarrow A$	$A \wedge R$
V	V	V	V	V	V
V	F	F	V	V	V
F	V	F	V	F	F
F	F	F	V	V	F

Fonte: elaborado pela autora.

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 externo)

3 - $((A \vee B) \rightarrow C) \wedge A$ (operação fora dos parênteses).

O Quadro 4.15 mostra o resultado para essa fórmula. Veja que usamos proposições intermediárias para nomear os resultados. Primeiro obtemos P ($A \vee B$) que é o resultado 1, depois usamos para obter Q ($P \rightarrow C$), que é o resultado 2, e por fim, o usamos para obter o resultado final da fórmula, o qual também nomear, por exemplo R ($Q \wedge A$).

Quadro 4.15 | Tabela Verdade para a fórmula $((A \vee B) \rightarrow C) \wedge A$

			P	Q	R
A	B	C	$A \vee B$	$P \rightarrow C$	$Q \wedge A$
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	V	F

Fonte: elaborado pela autora.

Reflita

Por praxe – mas não obrigatoriamente – construímos a Tabela Verdade colocando nas colunas mais à esquerda as proposições, em seguida, as colunas com as fórmulas. Em uma fórmula do tipo $(A \vee B) \rightarrow A$, precisamos primeiro obter o resultado de $A \vee B$, para depois fazer a implicação. Quando o resultado de uma operação é usado para entrada em outra operação faz diferença a ordem das operações lógicas que serão consideradas? Em outras palavras, nessa operação $(A \vee B) \rightarrow A$, obteremos o mesmo resultado se fizermos $A \rightarrow (A \vee B)$?

Uma dúvida que pode surgir é sobre como vamos aplicar todas essas operações e regras no universo da programação. A resposta é simples: vamos utilizar para construir uma sequência de instruções, chamada de algoritmo.

Um algoritmo é uma sequência de passos que soluciona algum problema de diversas áreas do mundo real. 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 (SOFFNER, 2013). Por exemplo, imagine que acabou de se mudar e esteja realizando a busca de um apartamento em um site de aluguel de imóveis. O site oferece uma interface na qual você clica nas opções que deseja, conforme ilustra a Figura 4.9.

Figura 4.9 | Interface de site de busca de imóvel

Fonte: elaborada pela autora.

Vamos fazer algumas simulações para avaliarmos a lógica por trás da nossa seleção.

Simulação 1: Selecionar Apartamento; 1 dormitório; 1 banheiro; sem vaga de garagem.

A expressão lógica que será construída com base nessa simulação é:

Apartamento E 1 quarto E 1 banheiro E sem garagem.

Serão exibidos na tela somente os imóveis que satisfazem todas essas alternativas; os demais serão ignorados, pois no algoritmo a instrução é composta pelo conector de conjunção.

Simulação 2: Selecionar Apartamento; 1, 2 dormitórios; 1, 2 banheiros; 0, 1 vaga de garagem.

Nesse caso a expressão lógica será:

Apartamento E (1 quarto OU 2 quartos) E (1 banheiro OU 2 banheiros) E (0 OU 1 vaga de garagem).

Veja que nesse caso a expressão é mais complexa e envolve a resolução de parênteses internos. Como resultado da busca apareceriam tanto opções com 1 quanto com 2 quartos e banheiros e também sem garagem ou com 1 vaga.

Todas essas opções devem estar implementadas no algoritmo por meio das estruturas condicionais.

Esse tipo de estrutura é amplamente usado em algoritmos computacionais. Quem nunca fez uma compra em um site? Ao finalizar a compra, o algoritmo faz uma conjunção dos itens que colocamos no carrinho de compras, para calcular o preço final.

Para finalizar nossa seção, vamos utilizar uma base com dados reais e, a partir de uma Tabela Verdade, veremos como classificar os registros. Os dados usados são de domínio público e estão disponíveis no Portal Brasileiro de Dados Abertos (BRASIL, 2019).

Trata-se de uma base com os preços de combustíveis em algumas cidades do Brasil no ano de 2019, segregados por estado, cidade, estabelecimento de revenda, produto, data da pesquisa, valor de venda, valor de compra e bandeira do posto, conforme amostra dos dados na Tabela 4.6.

Tabela 4.6 | Amostra dos dados de combustíveis automotivos

	Estado	Município	Revenda	Produto	Data da Coleta	Valor de Venda	Valor de Compra	Bandeira
1	DF	BRASÍLIA	ABRITTA POSTOS DE SERVIÇOS LTDA.	DIESEL S10	23/01/2019	3,899		xx
2	DF	BRASÍLIA	ABRITTA POSTOS DE SERVIÇOS LTDA.	ETANOL	16/01/2019	3,299	2,9999	xx
3	RJ	RIO DE JANEIRO	3POSTO JULIO DE CAS-TILHO LIMITA-DA.	GASOLI-NA	07/03/2019	5,096	4,305	xx
4	DF	BRASÍLIA	ABRITTA POSTOS DE SERVIÇOS LTDA.	GASOLI-NA	23/01/2019	4,299	3,6745	xx
5	RJ	RIO DE JANEIRO	ABASTE-CIMEN-TO DE COMBUS-TIVEIS RIO DO A LTDA.	ETANOL	03/01/2019	3,499	2,9711	xx

6	DF	BRASÍLIA	AM COMERCIAL DE COMBUSTIVEIS LTDA.	ETANOL	23/01/2019	3,197	2,9416	xx
7	DF	BRASÍLIA	AM COMERCIAL DE COMBUSTIVEIS LTDA.	GASOLINA	08/01/2019	4,187	3,735	xx
8	SC	BLUMENAU	AUTO POSTO 7 LTDA.	DIESEL S10	02/01/2019	3,597		xx
9	SC	BLUMENAU	AUTO POSTO 7 LTDA.	GASOLINA	12/02/2019	3,767		xx

Fonte: Brasil (2019).

Dadas as seguintes proposições:

A: O município é Brasília.

B: O combustível é gasolina.

C: A bandeira é Petrobras.

Vamos construir a Tabela Verdade para a regra **"Se a cidade não for Brasília e o combustível for gasolina então a bandeira é Petrobras."**. Em seguida, vamos fazer a conexão entre a lógica formal com a lógica computacional, mais especificamente com a lógica de programação, para avaliar a regra para cada um dos registros. O primeiro passo é traduzir a regra para uma fórmula lógica, o que nos resulta em $(\neg A \wedge B) \rightarrow C$. O resultado da Tabela Verdade para a fórmula está no Quadro 4.16. Veja que a primeira operação que fizemos foi a negação, que está dentro dos parênteses (seguindo a ordem de precedência dos operadores), depois fizemos a conjunção e chamamos o resultado de R. Usamos esse resultado para fazer a implicação.

Quadro 4.16 | Tabela Verdade para a fórmula $(\neg A \wedge B) \rightarrow C$

				R	
A	B	C	$\leftarrow A$	$\neg A \wedge B$	$R \rightarrow C$
V	V	V	F	F	V
V	V	F	F	F	V
V	F	V	F	F	V

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

Fonte: elaborado pela autora.

Lembrando que a Tabela Verdade nos permite avaliar a veracidade da expressão testando todas as combinações possíveis. Vamos interpretar alguns resultados do Quadro 4.16. Na linha 1, temos a seguinte expressão "Se é falso que a cidade não é Brasília e o combustível é gasolina, então a bandeira é Petrobras.". Tal expressão tem como resultado V, que se deve à falsidade no antecedente, impossibilitando avaliar o resultado que, portanto, é tomado como verdadeiro. Na linha 2, a expressão é "Se é falso que a cidade não é Brasília e o combustível é gasolina, então é falso que a bandeira é Petrobras". Dada a falsidade no antecedente e consequente, a expressão tem como resultado V. O único caso em que a implicação tem como resultado F, é quando o antecedente é verdadeiro e o consequente é falso.

Os operadores lógicos são usados na lógica de programação para dizer se uma condição é verdadeira ou falsa. Nesse exemplo, queremos criar uma lógica de programação para avaliar regra "Se a cidade não for Brasília e o combustível for gasolina então a bandeira é Petrobras." Pois bem, o Quadro 4.17 apresenta o resultado da análise das proposições A e B para os registros do Quadro 4.17, bem como da conjunção $\neg A \wedge B$.

Quadro 4.17 | Valoração de A e B

	Município	Produto		Resultado
	A	B	$\leftarrow A$	$\neg A \wedge B$
1	V	F	F	F
2	V	F	F	F
3	F	V	V	V
4	V	V	F	F
5	F	F	V	F
6	V	F	F	F
7	V	V	F	F
8	F	F	V	F
9	F	V	V	V

Fonte: elaborado pela autora.

Dentro de um algoritmo computacional, o resultado da conjunção, dentro de uma estrutura condicional, pode ser usado para classificar a proposição C, pois onde a condição for satisfeita (V) a proposição C também será. Na programação, o operador de implicação é feito por meio do comando se… então… No nosso exemplo podemos dizer se $\neg A \wedge B$ então C, ou seja, onde $\neg A \wedge B$ for V, C também será. Para matar um pouco da nossa vontade de começar a programar, veja no Quadro 4.18 como ficaria esse comando dentro de três importantes linguagens de programação.

Quadro 4.18 | Estrutura condicional em linguagens de programação

Linguagem	Sintaxe

C	if(A != "BRASILIA" && B == "GASOLINA")
	{
	printf("Petrobras");
	}

Java	if(A != "BRASILIA" && B == "GASOLINA")
	{
	System.out.println("Petrobras");
	}

Python	if A != "BRASILIA" and B == "GASOLINA": print("Petrobras")

Fonte: elaborado pela autora.

Como você pode observar no Quadro 4.18, o comando **se** é escrito em inglês (if) e o **então**, nas linguagens C e Java é a chave ({) e em python é o dois pontos (:). Outro detalhe é a escrita da conjunção; em C e Java é feito pelo **&&** e em python pelo comando **and**. Além desses conectivos lógicos, também foram usados operadores relacionais. A sintaxe != significa "diferente" e a sintaxe == significa "igual". Como você pode ver, a solução geral de um problema está na lógica, que não muda de uma linguagem para outra, o que muda é a sintaxe. Por isso, compreender os operadores lógicos é essencial para sua carreira. Não deixe de aprofundar seus estudos e lembre-se: quanto mais treinar, mais desenvolverá sua lógica e sua capacidade analítica.

Sem medo de errar

Caro estudante, chegou o momento de resolvermos mais um desafio. Como funcionário na área de *analytics* de uma empresa de varejo, você recebeu uma nova base de dados e precisa identificar os clientes que têm potencial de comprar na nova campanha. A regra para dizer se o cliente tem ou não potencial para comprar é dada pelas seguintes condições:

- Não importa o gênero (pode ser feminino ou masculino).
- Ele ou ela deve ter idade entre 30 e 45 anos.
- Ele ou ela deve ter feito acima de 10 compras.
- Ele ou ela deve ter um ticket médio acima de R$ 50,00.

A primeira parte do desafio consiste em escrever uma fórmula que traduza essas regras e, então classificar o resultado da fórmula para cada registro da base da dados.

Primeiro ponto importante para montar a fórmula é entender que todas as condições precisam ser satisfeitas, ou seja, estamos diante de conjunções. Como não importa o gênero – pode ser F ou M –, usaremos a disjunção. Agora é montar essa disjunção com as várias conjunções, utilizando os parênteses para indicar a ordem da valoração. Vamos começar escrevendo os itens já em fórmulas:

- (feminino OU masculino).
- (idade >=30 E idade <=45).
- (compra >= 10).
- (ticket médio >= 50).

Agora é só juntar os itens com a conjunção:

(feminino **OU** masculino) **E** (idade >=30 **E** idade <=45) **E** (compra >= 10) **E** (ticket médio >= 50). Veja que temos conectores que não são tão evidentes, como no caso da idade, em que precisamos usar a conjunção para delimitar a idade procurada. Agora vamos avaliar a fórmula para os dados. Observe os resultados na coluna "cliente_potencial" na Tabela 4.7.

Vamos analisar juntos alguns registros. Na linha 1, o gênero, a idade e o valor da última compra são satisfeitos, porém, o total de compras e o ticket médio não são, o que resulta em falso para a coluna que indica se o cliente é ou não potencial. Já na linha 4, todos os itens são atendidos, logo o cliente é classificado como V, ou seja, é um cliente com potencial de compra na campanha.

Tabela 4.7 | Resultado da valoração da fórmula

linha	codigo_cli	nome_cli	genero_cli	idade_cli	valor_ultima_compra	total_compras	ticket_medio	cliente_potencial	cupom_10	cupom_5
1	53682	Karly Dillon	F	40	74,84	5	45,00	F	?	?
2	58246	Channing Vazquez	M	49	98,04	20	200,00	F	?	?
3	27022	Adria Key	F	47	65,93	12	34,00	F	?	?
4	82075	Ella Nelson	F	34	94,01	16	150,00	V	?	?
5	90657	Arden Battle	M	48	21,73	4	23,00	F	?	?
6	80330	Brittany Ramirez	F	38	42,23	1	42,23	F	?	?
7	53989	Moses Graham	M	42	37,20	29	45,00	F	?	?
8	61370	Jin Fuller	M	31	86,00	35	123,00	V	?	?
9	41807	Phelan Blair	M	45	77,40	23	95,00	V	?	?
10	94269	Porter West	M	22	67,19	6	35,00	F	?	?
11	56516	Zena Skinner	F	54	73,98	15	60,00	F	?	?
12	38904	Teagan Rios	M	34	61,57	17	71,00	V	?	?

Fonte: elaborada pela autora.

Agora que já sabemos quais clientes têm potencial para comprar, vamos ao próximo passo: marcar o cupom de desconto. Devemos escrever uma nova fórmula que traduza "Se o cliente tem potencial de compra, então ele deve ganhar um cupom com 10% de desconto". Vamos chamar de P a proposição "o cliente tem potencial de compra" e R a proposição "Ganhar um cupom com 10% de desconto". Então a fórmula pode ser escrita como $P \rightarrow R$. Na lógica de programação, a implicação é utilizada em estruturas condicionais do tipo se... então... E como resultado temos que se P for verdadeira então R acontecerá.

Também foi dado a você o desafio de escrever a fórmula que traduza "Se é falso que o cliente tem potencial de compra, então ele deve ganhar um cupom com 5% de desconto". Considerando a proposição P anterior e a proposição S como “Ganhar um cupom com 5% de desconto”, obtemos a fórmula $\neg P \rightarrow S$. Agora já podemos finalizar o relatório completando as colunas que indicam a qual cupom o cliente tem direito. Veja na Tabela 4.8 o resultado completo. Vamos analisar a linha 1. Temos que o cliente não é potencial, ou seja, $\neg P$ é uma verdade nesse caso, logo S acontecerá.

Tabela 4.8 | Relatório final

linha	codigo_cli	nome_cli	genero_cli	idade_cli	valor_ultima_compra	total_compras	ticket_medio	cliente_potencial	cupom_10	cupom_5
1	53682	Karly Dillon	F	40	74,84	5	45,00	F	F	V
2	58246	Channing Vazquez	M	49	98,04	20	200,00	F	F	V
3	27022	Adria Key	F	47	65,93	12	34,00	F	F	V
4	82075	Ella Nelson	F	34	94,01	16	150,00	V	V	F
5	90657	Arden Battle	M	48	21,73	4	23,00	F	F	V
6	80330	Brittany Ramirez	F	38	42,23	1	42,23	F	F	V
7	53989	Moses Graham	M	42	37,20	29	45,00	F	F	V
8	61370	Jin Fuller	M	31	86,00	35	123,00	V	V	F
9	41807	Phelan Blair	M	45	77,40	23	95,00	V	V	F
10	94269	Porter West	M	22	67,19	6	35,00	F	F	V
11	56516	Zena Skinner	F	54	73,98	15	60,00	F	F	V
12	38904	Teagan Rios	M	34	61,57	17	71,00	V	V	F

Fonte: elaborada pela autora.

Com esses resultados finalizamos nosso desafio, contribuindo com êxito com a área de marketing da empresa. Esperamos que venham novos desafios!

Faça valer a pena

1. Segundo Gersting (2017), é possível construir expressões lógicas mais complexas a partir da combinação das proposições, dos conectivos e dos parênteses. Da mesma forma que as operações matemáticas possuem ordem de precedência, os conectivos lógicos também as possuem e, para se chegar à correta valoração de uma fórmula, é preciso seguir com rigor tais regras.

Dada a fórmula $A \vee C \rightarrow (B \wedge C)$, escolha a alternativa que representa a correta sequência de valoração da fórmula.

a. $A \vee C$; $B \wedge C$; $A \vee C \rightarrow (B \wedge C)$.

b. $A \vee C \rightarrow (B \wedge C)$; $B \wedge C$; $A \vee C$.

c. $A \vee C$; $A \vee C \rightarrow (B \wedge C)$; $B \wedge C$.

d. $B \wedge C$; $A \vee C$; $A \vee C \rightarrow (B \wedge C)$.

e. $A \vee C \rightarrow (B \wedge C)$; $A \vee C$; $B \wedge C$.

2. A Tabela Verdade deve ser usada como ferramenta para extração dos resultados de todas as combinações possíveis. As proposições devem ser dispostas nas colunas mais à esquerda e depois as fórmulas que se deseja valorar. O resultado depende da combinação de entradas aplicada a cada fórmula.

Dada a sequência de entradas VF – VV – FV – FF para a fórmula $P \rightarrow R \wedge P$, escolha a opção que representa a sequência correta de respostas que serão obtidas.

a. V – F – V – F.

b. F – V – V – V.

c. V – V – F – F.

d. F – F – V – V.

e. F – V – F – V.

3. A resolução de expressão lógica deve seguir os mesmos conceitos da resolução de expressões matemáticas. Ou seja, além de ter uma correta sintaxe, também é preciso se atentar à ordem de valoração de cada operador. O resultado de cada uma das operações deve ir se combinando até que se obtenha o resultado final.

Dada a fórmula $A \wedge \neg B \rightarrow (\neg A \vee B)$ escolha a Tabela Verdade que apresenta a correta valoração da fórmula proposta.

a.

				P	R	
A	B	$\leftarrow A$	$\leftarrow B$	$A \wedge \neg B$	$\neg A \vee B$	$R \rightarrow P$
V	V	F	F	F	V	F
V	F	F	V	V	F	V
F	V	V	F	F	V	F
F	F	V	V	F	V	F

b.

				P	R	
A	B	$\leftarrow A$	$\leftarrow B$	$\neg A \vee B$	$A \wedge \neg B$	$R \rightarrow P$
V	V	F	F	V	V	V
V	F	F	V	F	V	F
F	V	V	F	V	F	F
F	F	V	V	V	V	V

c.

				P	R	
A	B	$\leftarrow A$	$\leftarrow B$	$\neg A \vee B$	$A \wedge \neg B$	$P \rightarrow R$
V	V	F	F	V	F	F
V	F	F	V	F	V	V
F	V	V	F	V	F	F
F	F	V	V	V	F	F

d.

				P	R	
A	B	$\leftarrow A$	$\leftarrow B$	$A \wedge \neg B$	$\neg A \vee B$	$P \rightarrow R$
V	V	F	F	F	V	V
V	F	F	V	F	F	V
F	V	V	F	F	V	V
F	F	V	V	F	V	V

e.

				P	R	
A	B	$\leftarrow A$	$\leftarrow B$	$\neg A \vee B$	$A \wedge \neg B$	$R \rightarrow P$
V	V	F	F	V	F	V
V	F	F	V	F	V	F
F	V	V	F	V	F	V
F	F	V	V	V	F	V

Referências

BRASIL. Portal Brasileiro de dados abertos. **1º Sem 2019**. Brasília, 2019. Disponível em: http://dados.gov.br/dataset/serie-historica-de-precos-de-combustiveis-por-revenda/resource/927fe124-0648-4b17-8f05-e1ccf39ab358. Acesso em: 12 dez. 2019.

CHIBENI, S. S. **Notas sobre lógica:** o condicional. Disponível em: https://www.unicamp.br/~chibeni/textosdidaticos/condicional.pdf. Acesso em: 2 dez. 2019.

DACHI, E. P.; HAUPT, A. G. **Eletrônica digital.** São Paulo: Blucher, 2018.

ESTRUTURAS Condicionais I – Curso de Algoritmos #07 – Gustavo Guanabara. Curso em Vídeo. YouTube, [s.d.]. Disponível em: https://www.youtube.com/watch?v=_g05aHdBAEY&list=PLHz_AreHm4dmSj0MHol_aoNYCSGFqvfXV&index=8. Acesso em: 15 dez. 2019.

GERSTING, J. L. **Fundamentos matemáticos para a ciência da computação:** matemática discreta e suas aplicações. 7. ed. Rio de Janeiro: LTC, 2017.

SILVA, F. S. C. da; FINGER, M.; MELO, A. C. V. de. **Lógica para computação.** 2. ed. São Paulo: Cengage Learning, 2017.

SOFFNER, R. Algoritmos e programação em linguagem C. São Paulo: Saraiva, 2013.

ISBN 978-85-522-1688-9

9 788552 216889 >