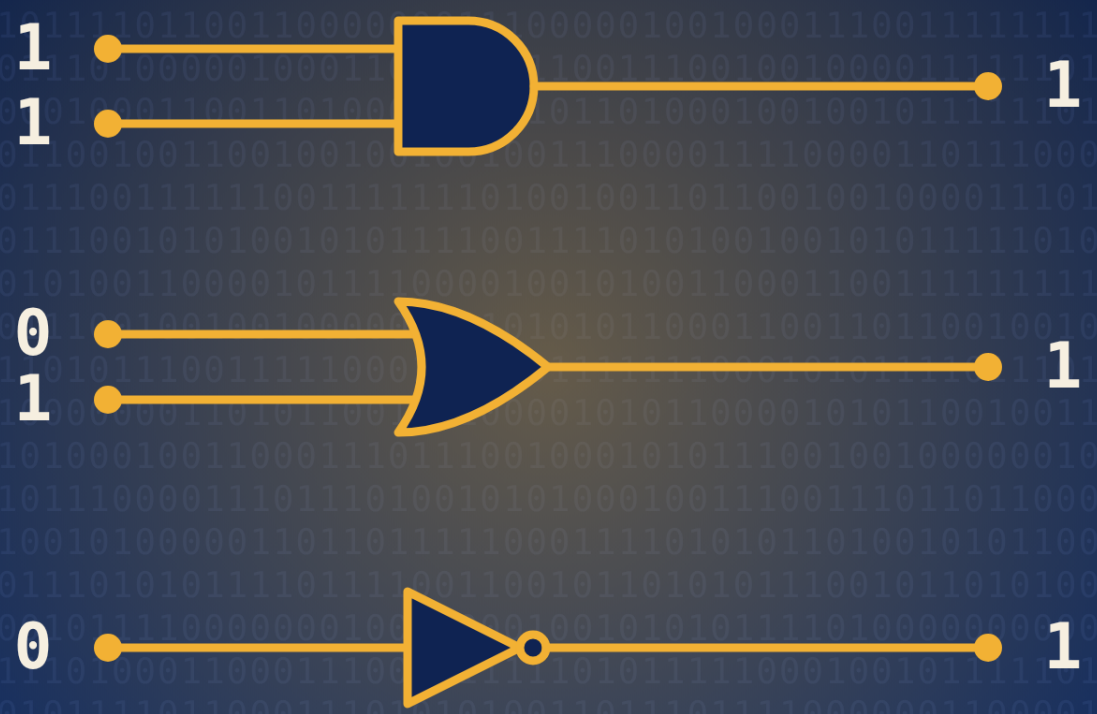
A LÓGICA QUE PENSA

Boole, a álgebra e o nascimento dos computadores



DE 1854 A 1971

Belisário Retto de Abreu

Sumário

Apresentação 1

Capítulo 1: Uma pergunta antiga, pensar pode ser calculado? 1

Capítulo 2: George Boole, o menino que ensinou a si mesmo 7

Capítulo 3: As leis do pensamento, a álgebra de 0 e 1 10

Capítulo 4: E, OU, NÃO, a álgebra booleana no dia a dia 12

Capítulo 5: De Morgan, Jevons, Venn e Peirce, a lógica ganha diagrama… 15

Capítulo 6: Babbage e Ada Lovelace, a máquina que calculava e a ideia… 18

Capítulo 7: Cartões perfurados, Jacquard, Hollerith e o censo 21

Capítulo 8: Claude Shannon, quando a lógica encontrou os relés 24

Capítulo 9: Alan Turing e a máquina universal 27

Capítulo 10: Zuse, Atanasoff e Aiken, os primeiros computadores 30

Capítulo 11: Guerra e computação, Colossus, ENIAC e as mulheres que… 33

Capítulo 12: Von Neumann e o programa armazenado 36

Capítulo 13: Do relé à válvula, ao transistor e ao chip: as portas lógicas 39

Capítulo 14: No Brasil, os primeiros computadores brasileiros 42

Capítulo 15: A lógica no bolso, a álgebra booleana hoje 45

Linha do tempo e glossário 47

Fontes 49

A Lógica que Pensa: Boole, a álgebra e o nascimento dos computadores

Apresentação

Este livro mostra, para o leitor curioso e sem formação técnica, como uma ideia publicada em 1854 se tornou a base de todo computador. A narrativa é histórica: gente de carne e osso, erros, rivalidades e acasos, com a matemática explicada só quando a história pede.

São 15 capítulos, do sonho de Leibniz de calcular o raciocínio até a lógica que hoje cabe no bolso, passando por Boole, Babbage, Ada Lovelace, Shannon, Turing e pelos primeiros computadores brasileiros. Cada capítulo traz uma cena de abertura, a história, figuras e tabelas, uma seção "Na prática" com um exercício de álgebra booleana para resolver com lápis, e uma linha do tempo curta no fim.

Capítulo 1: Uma pergunta antiga, pensar pode ser calculado?

Em algum momento de 1701, um envelope atravessou meio mundo até chegar a uma escrivaninha em Hanôver, na Alemanha. Quem o enviava era Joachim Bouvet, um padre jesuíta francês que vivia na China e frequentava a corte do imperador. Quem o recebia era Gottfried Wilhelm Leibniz, um homem que já tinha sido diplomata, bibliotecário, inventor e filósofo, e que se recusava a escolher uma só dessas vidas.

Dentro do envelope havia um desenho de figuras estranhas: pilhas de seis linhas, algumas inteiras, outras partidas ao meio. Eram os hexagramas do *I Ching*, o antigo livro chinês das mutações, organizados numa sequência que, segundo a tradição, remontava ao lendário Fu Xi. Para Bouvet, aquilo era uma curiosidade oriental. Para Leibniz, foi um choque. Ele olhou as linhas inteiras e as linhas partidas e viu o que ninguém ao redor via: um sistema de numeração com apenas dois símbolos, o mesmo sistema que ele próprio vinha imaginando havia mais de vinte anos.

Leibniz escreveu de volta, entusiasmado, e em 1703 publicou um pequeno texto sobre a aritmética binária. A ideia parecia inútil na época. Quem precisaria contar só com zeros e uns, se dez dedos e dez algarismos funcionavam tão bem? Mas Leibniz suspeitava de algo maior do que um truque de contagem. Para ele, o 1 e o 0 eram quase uma imagem do mundo: o ser e o nada, o sim e o não, a luz acesa e a luz apagada. Se o raciocínio humano também pudesse ser reduzido a escolhas desse tipo, talvez fosse possível fazer com ele o que se faz com números: calcular.

Este livro conta o que aconteceu com essa suspeita. Ela atravessou três séculos, passou pelas mãos de um professor irlandês de matemática, de uma condessa londrina, de um engenheiro de telefonia e de um jovem britânico que morreu injustiçado, até desembocar no objeto que você provavelmente tem a menos de um metro de distância agora. Mas, antes de chegar lá, precisamos entender por que a pergunta de Leibniz era tão antiga e tão ousada. Pensar pode ser calculado? Gente muito inteligente se fez essa pergunta bem antes dele.

O homem que classificou o raciocínio

A primeira pessoa a tratar o argumento como uma coisa que se pode examinar por fora, como quem examina uma ponte, foi Aristóteles. No século IV antes de Cristo, em Atenas, ele notou que certos raciocínios são convincentes por causa da forma e não do assunto. Veja este: todo humano é mortal; Sócrates é humano; logo, Sócrates é mortal. Agora troque as palavras: toda baleia é mamífera; Moby Dick é baleia; logo, Moby Dick é mamífera. O assunto mudou por completo, mas a estrutura é idêntica, e é ela que nos obriga a aceitar a conclusão.

Aristóteles chamou essa estrutura de silogismo e passou anos catalogando as formas válidas e as inválidas, com nomes que os estudantes medievais decoravam como quem decora cantiga. Foi o primeiro catálogo de maneiras corretas de pensar. Repare no que isso significa: se a validade depende só da forma, então dava, em princípio, para verificar um argumento sem entender do que ele fala. Bastava conferir o encaixe das peças.

A figura abaixo mostra o encaixe do exemplo de Sócrates. Cada grupo é um círculo. Se o círculo dos humanos cabe inteiro dentro do círculo dos mortais, e Sócrates está dentro do círculo dos humanos, não há como ele estar fora do círculo dos mortais. A conclusão não é uma opinião: é uma consequência geométrica. Dois mil anos depois, desenhos desse tipo ganhariam o nome de diagramas de Euler e de Venn, e voltarão neste livro no capítulo sobre Boole.

Um círculo dentro do outro: a conclusão não precisa de mais nada

Premissa 1
Todo humano é mortal.

Premissa 2
Sócrates é humano.

Conclusão
Logo, Sócrates é mortal.

Desenho original para este livro:
a lógica de Aristóteles em círculos.

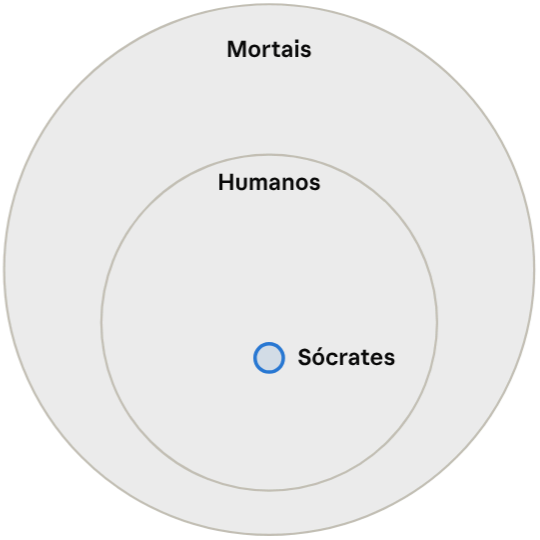


Figura 1.1 · o silogismo de Aristóteles em três círculos

Havia, porém, um limite sério. O sistema de Aristóteles tratava de poucas formas de frase, todas do tipo “todo A é B” ou “algum A é B”. Era elegante, mas estreito, e ficou praticamente inalterado por cerca de dois mil anos. Kant chegou a dizer, no século XVIII, que a lógica nascera pronta e acabada com Aristóteles. Estava enganado, e a história que vem a seguir é a prova disso.

A máquina de gerar ideias de um poeta de Maiorca

O passo seguinte veio de um lugar inesperado. Ramon Llull nasceu na ilha de Maiorca por volta de 1232, numa família rica, e passou a juventude como cortesão e poeta. Segundo a tradição, mudou de vida depois de uma série de visões, vendeu o que tinha e dedicou o resto dos dias a uma missão: converter muçulmanos e judeus ao cristianismo, mas pela razão, não pela espada. Para isso, achava ele, era preciso um método que fosse irrefutável para qualquer pessoa, de qualquer religião.

O método que ele inventou parece, à primeira vista, um brinquedo. Llull escolheu um pequeno conjunto de conceitos tidos como indiscutíveis, como bondade, grandeza, poder, sabedoria e verdade, e deu a cada um uma letra. Depois desenhou discos de papel concêntricos, um dentro do outro, com essas letras marcadas na borda. Ao girar um disco em relação ao outro, diferentes letras ficavam lado a lado, e cada alinhamento era uma afirmação possível: a bondade é grande, a grandeza é poderosa, o poder é verdadeiro.

A roda não sabia nada. Ela apenas cobria, de forma mecânica e sem esquecer nenhuma, todas as combinações possíveis. Cabia ao usuário decidir quais faziam sentido. Hoje reconhecemos nisso o primeiro esboço de uma ideia que os programadores usam todos os dias: gerar sistematicamente todas as possibilidades e depois filtrar. Llull não chegou perto de resolver os problemas que sonhava resolver, e foi muito criticado por isso. Quatro

séculos depois, o escritor Jonathan Swift fez troça das rodas de Llull em *As Viagens de Gulliver*, descrevendo uma máquina de escrever livros em que os professores giravam manivelas para formar frases ao acaso.

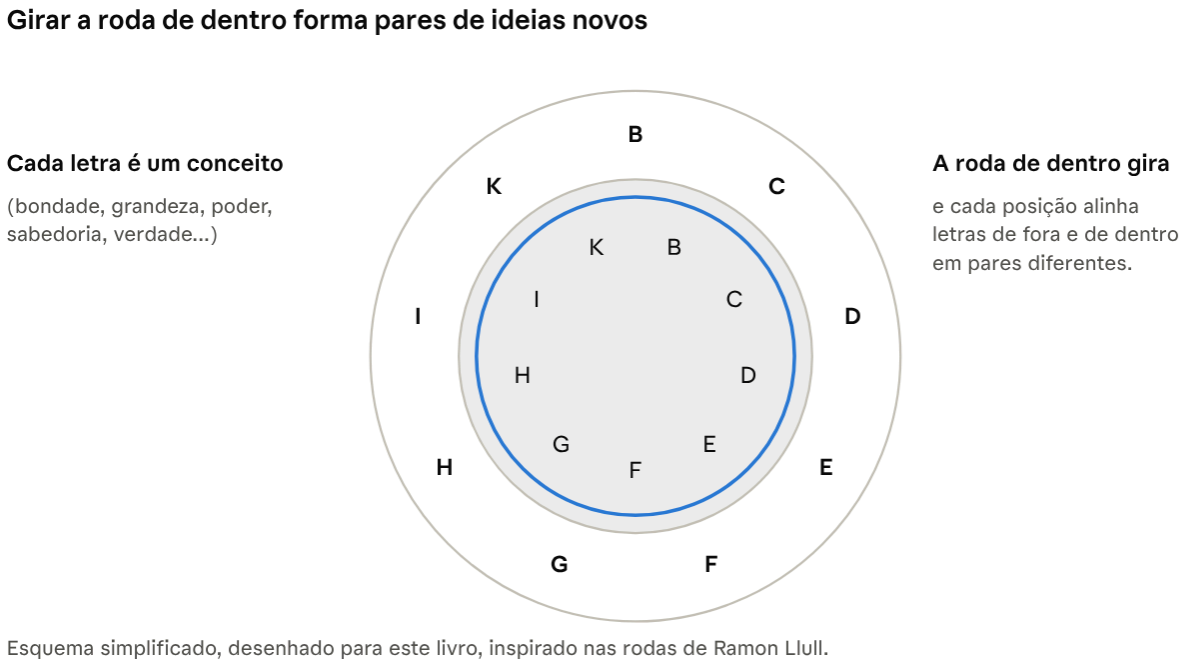


Figura 1.2 · esquema das rodas giratórias de Ramon Llull

A troça não diminui a importância da ideia. O que Llull fez, sem saber, foi dar o segundo passo da nossa história: depois de Aristóteles mostrar que o raciocínio tem forma, ele sugeriu que essa forma poderia ser manipulada por um aparelho. Faltava apenas um jeito de fazer o aparelho calcular de verdade, e foi nisso que um alemão muito persistente apostou três séculos mais tarde.

Leibniz e o sonho de calcular o raciocínio

Leibniz conhecia Llull e conhecia Aristóteles, e achava que ambos tinham chegado perto de algo sem alcançá-lo. Nascido em Leipzig em 1646, ele foi uma espécie de máquina de ideias em carne e osso: inventou o cálculo diferencial em paralelo com Isaac Newton, desenhou carruagens, projetou bombas para minas de prata e escreveu milhares de cartas para metade dos sabiões da Europa. Em meio a tudo isso, carregava uma obsessão que nunca largou.

A obsessão era uma linguagem universal do pensamento, a que chamava *characteristica universalis*. Imaginava uma escrita em que cada ideia simples teria um símbolo, e as ideias complexas seriam combinações desses símbolos, como as palavras são combinações de letras. Junto com ela viria um *calculus ratiocinator*, um cálculo do raciocínio: regras para manipular os símbolos de modo que os erros de argumentação aparecessem como erros de conta. Numa frase famosa, ele imaginou que, diante de uma disputa, duas pessoas não precisariam mais brigar. Bastaria pegar o lápis e dizer: vamos calcular.

Leibniz não ficou só na teoria. Por volta de 1672 ele construiu uma calculadora mecânica capaz de somar, subtrair, multiplicar e dividir, com um cilindro dentado que ficou conhecido como roda de Leibniz. A máquina era engenhosa, mas temperamental, e nunca funcionou tão bem quanto ele anunciava aos amigos. Ainda assim, a combinação é reveladora: uma mão buscava a linguagem lógica do pensamento, a outra buscava uma máquina que fizesse contas. Faltava juntar as duas.

O elo era o binário. Quando o envelope de Bouvet chegou, Leibniz percebeu que o sistema de zeros e uns tinha uma vantagem que o nosso sistema decimal não tem: cada dígito só admite duas possibilidades, ligado ou desligado, e duas possibilidades é exatamente o que uma peça mecânica simples consegue representar. Mais do que isso, duas possibilidades é também o que a lógica de Aristóteles usa: uma frase é verdadeira ou falsa. Leibniz intuiu que as duas coisas se encontrariam, mas não soube dizer como. Ele escreveu que o binário seria uma grande maravilha do mundo, e morreu em 1716 sem ter feito o encontro acontecer. Foi preciso esperar mais de um século para que alguém tivesse a ideia certa.

Contar com lâmpadas

O sistema binário parece estranho só porque estamos acostumados com o outro. Nosso sistema usa dez símbolos e valores que crescem de dez em dez: unidades, dezenas, centenas. No binário, cada posição só tem dois símbolos, e os valores crescem de dois em dois: 1, 2, 4, 8, 16, e assim por diante.

Para sentir isso na prática, imagine uma fileira de três lâmpadas. A da direita vale 1, a do meio vale 2, a da esquerda vale 4. Para formar um número, você acende as lâmpadas cujos valores somam o que quer. O número 5 pede a lâmpada do 4 e a do 1, então ficam acesas a primeira e a terceira. O 6 pede o 4 e o 2. O 3 pede o 2 e o 1. A figura mostra esses três casos, e a tabela abaixo traz todos os números de 0 a 7.

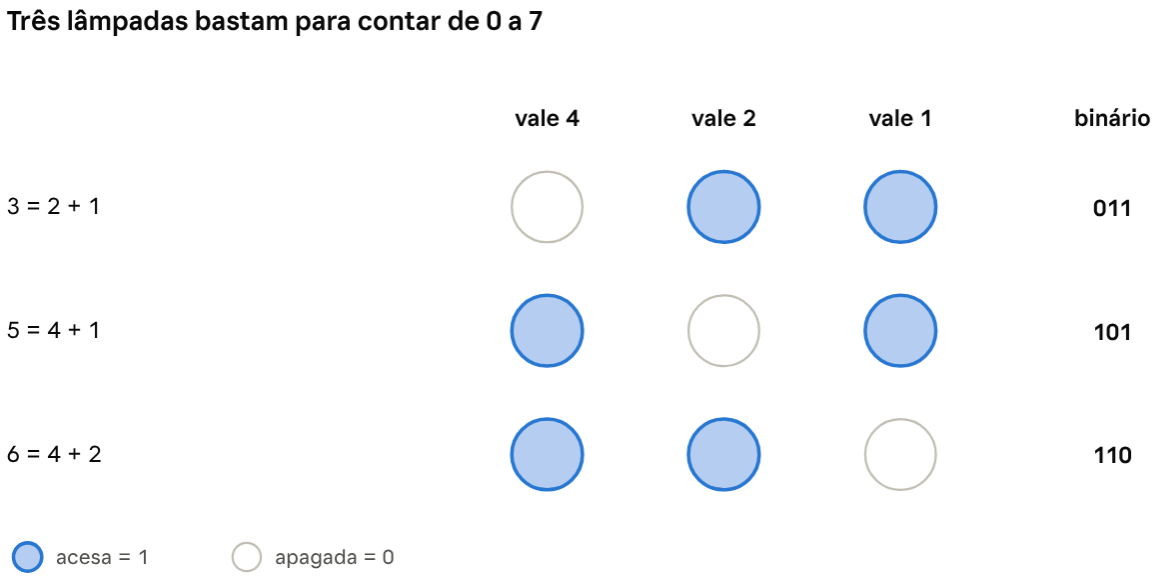


Figura 1.3 · três lâmpadas representando 3, 5 e 6

Número	Lâmpada 4	Lâmpada 2	Lâmpada 1	Binário
0	apagada	apagada	apagada	000
1	apagada	apagada	acesa	001
2	apagada	acesa	apagada	010
3	apagada	acesa	acesa	011
4	acesa	apagada	apagada	100
5	acesa	apagada	acesa	101
6	acesa	acesa	apagada	110
7	acesa	acesa	acesa	111

Repare em duas coisas. A primeira é que com três lâmpadas você tem exatamente oito combinações, de 000 a 111, e que cada lâmpada a mais dobra esse total: com quatro seriam dezesseis, com dez seriam 1.024. A segunda é que cada lâmpada é apenas um sim ou não, e é por isso que um interruptor, uma válvula ou um transistor, cada um com seus dois estados, consegue guardar um número inteiro quando está em fila com outros. Nenhuma peça precisa saber contar. Só precisa estar ligada ou desligada.

Na prática: seu primeiro número binário

Pegue três objetos que tenham dois estados, como três moedas (cara vale 1, coroa vale 0) ou três copos (virado para cima vale 1, para baixo vale 0). Coloque-os em fila, o da direita valendo 1, o do meio valendo 2 e o da esquerda valendo 4. Agora tente, sem olhar a tabela:

- Mostre o número 7. (Resposta: os três ligados, 111.)
- Mostre o número 4. (Resposta: só o da esquerda, 100.)
- Mostre o número 2 e depois some 1 a ele sem desmontar tudo. (Resposta: de 010 para 011.)
- Desafio: quanto dá 5 + 1? (Resposta: 101 mais 001 dá 110, ou seja, 6. Repare que o 1 da direita “vai” para a casa ao lado, exatamente como o “vai um” das contas que você aprendeu na escola.)

Se você fez isso, acabou de executar a operação básica de todo computador do planeta.

Linha do tempo do capítulo

Quando	O que aconteceu
século IV a.C.	Aristóteles cataloga as formas válidas de raciocínio (o silogismo).
por volta de 1300	Ramon Llull usa rodas de papel para combinar conceitos de modo sistemático.
por volta de 1672	Leibniz constrói uma calculadora mecânica com cilindro dentado.
1701 a 1703	Leibniz recebe os hexagramas do *I Ching* e publica sua aritmética binária.
1716	Leibniz morre sem unir lógica e binário.

Capítulo 2: George Boole, o menino que ensinou a si mesmo

Na cidade de Lincoln, na Inglaterra, o ano de 1815 começou a terminar em novembro, e foi nesse mês que nasceu o primeiro filho de um sapateiro. O pai, John Boole, tinha um ofício humilde e uma curiosidade desproporcional: gostava de instrumentos ópticos, de telescópios e de fazer lentes, e passava os domingos explicando aos vizinhos o que via no céu. O menino, George, cresceu entre sapatos e lentes e aprendeu cedo que um objeto qualquer podia ser um mistério.

A família não tinha dinheiro para uma boa escola, e a Inglaterra do começo do século XIX não oferecia muita coisa a quem não tinha. George estudou numa escola comercial simples, onde aprendeu o básico, e foi o pai quem lhe passou os primeiros fundamentos de matemática. O resto ele conseguiu sozinho. Aprendeu latim com a ajuda de um livreiro da cidade, depois grego, depois francês, alemão e italiano, porque queria ler os grandes matemáticos no idioma em que haviam escrito.

O poema que ninguém acreditou

Por volta dos catorze anos, George traduziu para o inglês, em versos, um poema grego de um poeta antigo chamado Meleágro. O pai ficou tão orgulhoso que o publicou no jornal local. A redenção durou pouco: alguns leitores acharam inconcebível que um menino pobre de catorze anos tivesse feito aquilo e insinuaram que o texto era de outra pessoa. Era a primeira vez que a sociedade lhe dizia, com todas as letras, que um filho de sapateiro não deveria ser tão bom. Ele respondeu com uma carta cortês e firme no jornal, defendendo a autoria, e voltou aos livros.

O episódio importa por duas razões. Primeiro, mostra um traço que acompanhou Boole pela vida toda: não havia obstáculo que o fizesse parar de estudar. Segundo, revela de onde vinha a sua maneira de pensar. Quem traduz poesia de uma língua para outra aprende que o sentido de uma frase pode ser preservado enquanto as palavras mudam por completo. Esse tipo de pensamento, o de olhar a estrutura por baixo das palavras, seria exatamente o que ele aplicaria mais tarde à lógica.

Professor aos dezesseis, diretor aos dezenove

Quando o negócio do pai começou a afundar, George, ainda adolescente, precisou sustentar a família. Aos dezesseis anos já dava aulas como ajudante numa escola de outra cidade. Aos dezenove, abriu a própria escola em Lincoln, onde ensinava crianças pela manhã e estudava a noite inteira. Foi nesse período, com a escola funcionando, que ele resolveu aprender matemática de verdade. Não havia universidade ao alcance, então escolheu o jeito mais difícil: ler direto as obras dos mestres, como as de Laplace e Lagrange, e refazer as contas até entender cada passo.

Logo começou a ter ideias suas. Escreveu artigos para uma revista de matemática de Cambridge, e um deles, sobre uma maneira de tratar operações como se fossem números, chamou a atenção de leitores importantes. Em 1844 a Royal Society, a mais respeitada associação científica da Grã-Bretanha, lhe deu uma medalha. Pense no que isso significa: um professor primário de cidade pequena, sem diploma, sendo premiado por quem tinha todos os diplomas.

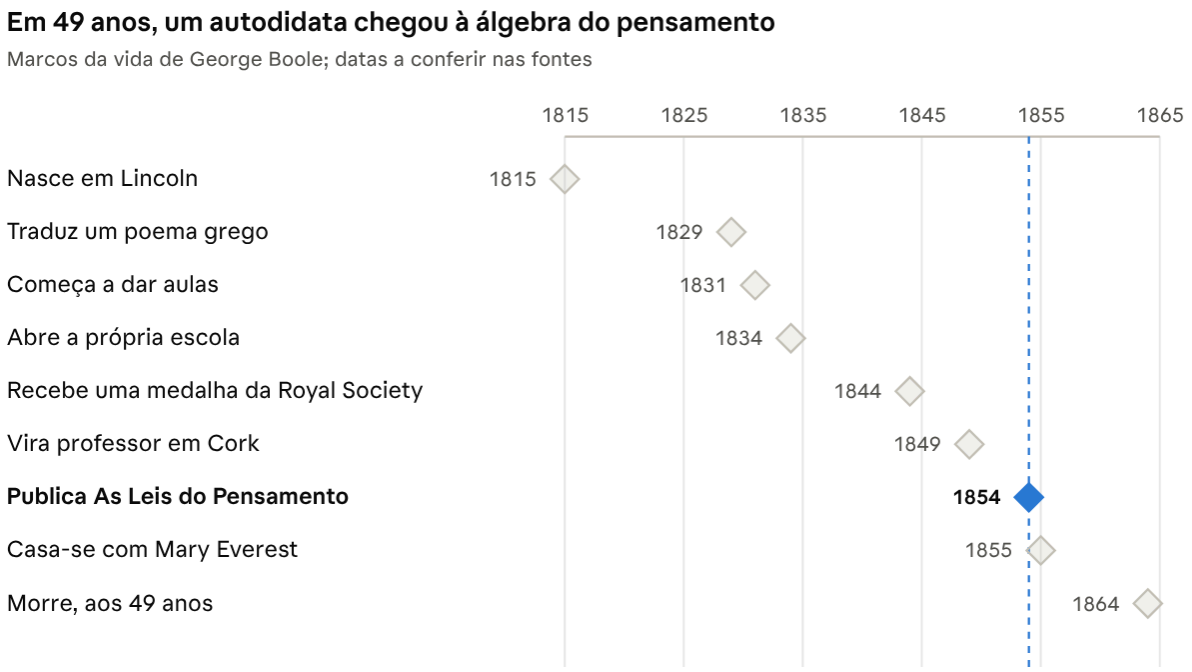


Figura 2.1 · a vida de Boole em nove marcos

Cork, a família e o fim prematuro

Em 1849 a vida de Boole mudou de vez. A Irlanda tinha acabado de abrir universidades novas, e uma delas, em Cork, procurava um professor de matemática. Apesar de não ter nenhum título universitário, Boole foi escolhido pelo que já havia publicado. Mudou-se para a Irlanda, onde passaria o resto da vida. Cinco anos depois, em 1854, publicou a obra que o tornaria imortal, de que trataremos no próximo capítulo. Em 1855 casou-se com Mary Everest, sobrinha de um famoso agrimensor que dá nome à maior montanha do mundo e, ela própria, uma pensadora de ideias originais sobre educação matemática.

O casal teve cinco filhas. Algumas delas deixaram marca: Alicia, que desenvolveu uma intuição geométrica rara sobre figuras de quatro dimensões, e Ethel, que se tornou romancista de sucesso na Europa e nos Estados Unidos. A família viveu feliz, e tudo parecia promissor.

O fim veio cedo. Em dezembro de 1864, aos 49 anos, Boole morreu de pneumonia. A tradição conta que ele atravessou a pé uma chuva forte para dar aula e passou o dia com a roupa molhada, mas essa história é repetida mais do que provada. O que se sabe é que o homem que inventou a álgebra do pensamento não viveu para ver nenhuma das suas aplicações.

Na prática: agrupando por propriedades

Trabalhe com uma cesta de frutas de verdade, ou imagine uma: morango, cereja, maçã vermelha, banana, laranja, uva roxa. Separe o grupo das frutas vermelhas. Depois separe o grupo das frutas redondas. Agora responda: quais frutas pertencem ao mesmo tempo aos dois grupos? (Resposta: maçã vermelha e cereja, se você considerou o morango não redondo; o importante é notar que a palavra “e” virou uma regra para escolher.) Boole percebeu que dava para tratar cada grupo como se fosse uma quantidade e o “e” como se fosse uma multiplicação. Essa ideia estranha, que você acaba de usar sem perceber, é o ponto de partida do próximo capítulo.

Capítulo 3: As leis do pensamento, a álgebra de 0 e 1

Em 1854, numa gráfica de Londres, os exemplares de um livro de cerca de quatrocentas páginas foram empilhados e encaminhados aos livreiros. O título era comprido e solene: *Uma Investigação das Leis do Pensamento, sobre as quais se Fundam as Teorias Matemáticas da Lógica e das Probabilidades*. O autor, o professor George Boole, de Cork, não esperava que o livro vendesse bem. Esperava, no máximo, que alguns leitores atentos entendessem o que ele tinha feito. Nenhum deles poderia imaginar que aquela ideia, passando por mãos muito diferentes, acabaria dentro de aparelhos que cabem no bolso.

Uma briga que empurrou Boole

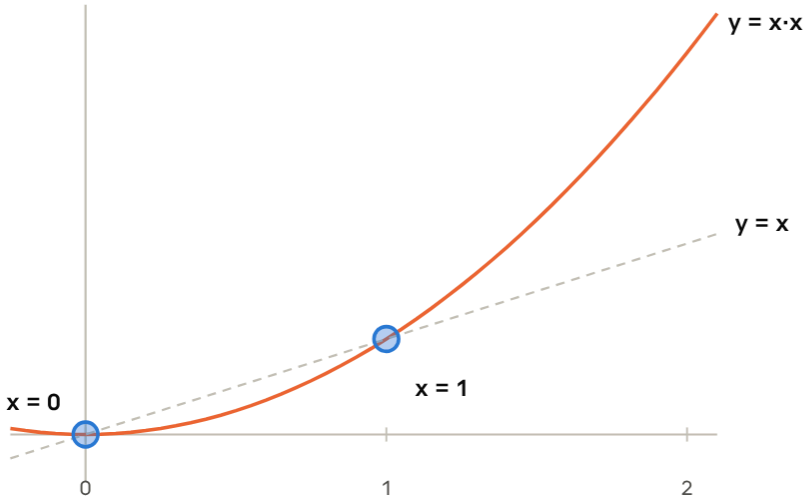
A história do livro começa com uma discussão pública. Em 1846 e 1847, dois acadêmicos britânicos, Augustus De Morgan, de Londres, e William Hamilton, de Edimburgo, trocaram acusações sobre quem tinha tido primeiro uma certa ideia de lógica. A briga saiu em cartas, panfletos e jornais, e deixou a lógica, por alguns meses, na boca de todo mundo que ligava para essas coisas. Boole, então um professor de província, acompanhou tudo de longe e percebeu que tinha algo diferente para dizer. Em 1847 escreveu depressa um pequeno livro, *A Análise Matemática da Lógica*, que é o primeiro esboço do que viria. O livro grande de 1854 é a versão madura da mesma ideia.

A ideia: transformar palavras em contas

O ponto de partida de Boole é simples de enunciar e difícil de ter. Em vez de tratar uma frase lógica como uma cadeia de palavras, ele sugeriu tratá-la como uma conta. Pegue a palavra “animais”. Ela nomeia um grupo de coisas. Chame esse grupo de x. Agora pegue a palavra “vermelhos” e chame o grupo das coisas vermelhas de y. A expressão “animais vermelhos” nomeia o grupo das coisas que estão ao mesmo tempo nos dois, e Boole escreveu isso como o produto x · y, a mesma notação da multiplicação.

O truque tem um efeito curioso. Se você diz “animais animais”, não está dizendo nada além de “animais”, e portanto x · x = x. Essa equação, que seria absurda com números quaisquer, é a lei fundamental do pensamento segundo Boole. E ela tem uma consequência deliciosa: se você resolve x · x = x como equação, só encontra duas respostas, 0 e 1. A figura 3.1 mostra isso: a curva y = x · x e a reta y = x só se encontram em dois pontos.

A equação x · x = x só tem duas soluções: 0 e 1



Só em 0 e em 1 a curva encontra a reta. Em qualquer outro número, x · x é diferente de x.

Figura 3.1 · y = x·x cruza y = x em 0 e em 1

Boole tirou daí uma conclusão ousada. Se a única coisa que importa é que a equação só admite 0 e 1, então podemos dar a esses números um sentido lógico. O 1 passa a significar “tudo”, o universo de que estamos falando. O 0 passa a significar “nada”, o grupo vazio. E qualquer grupo x fica em algum lugar entre esses dois extremos. O que não é x vale 1 − x, ou seja, tudo menos x.

Para Boole, “tudo” vale 1 e “nada” vale 0

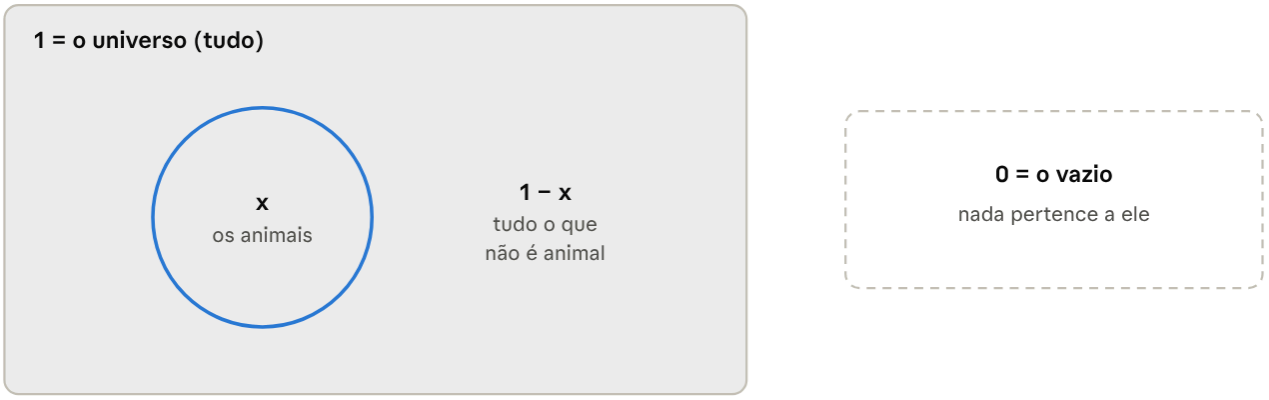


Figura 3.2 · o universo (1) e o vazio (0)

Com essas peças, as regras da álgebra comum passam a falar de lógica. Veja um exemplo que o próprio Boole destacou. Se escrevermos que $x \cdot x = x$ e passarmos tudo para um lado, obtemos $x - x \cdot x = 0$, que pode ser reescrita como $x \cdot (1 - x) = 0$. Em palavras: o que é ao mesmo tempo animal e não animal não existe. Esse é o velho princípio da não contradição de Aristóteles, que ninguém tinha precisado provar por dois mil anos, e que de repente saía de uma conta de colegial.

Por que o livro importou, e por que demorou

Boole acreditava estar descrevendo as leis reais do pensamento humano, e escreveu os últimos capítulos do livro tratando de filosofia e até de religião. Hoje, ninguém acha que o cérebro funcione exatamente assim. O que sobreviveu não foi a psicologia de Boole, e sim a sua álgebra, que virou uma ferramenta prática.

O livro teve boa recepção entre os poucos que o leram, mas ficou longe do grande público. Outros pensadores, como Jevons e Peirce, que vamos encontrar no capítulo 5, ajustaram detalhes e melhoraram as notações. Faltava uma razão prática para usar a álgebra de Boole, e essa razão só apareceria no século seguinte, quando os engenheiros precisassem projetar circuitos com milhares de interruptores. Até lá, a álgebra de 0 e 1 dormiu nos livros de matemática.

Na prática: por que $x \cdot x = x$ só aceita 0 e 1

Pegue lápis e papel e teste três valores na equação $x \cdot x = x$. Com $x = 0$, temos $0 \cdot 0 = 0$, e funciona. Com $x = 1$, temos $1 \cdot 1 = 1$, e também funciona. Com $x = 2$, temos $2 \cdot 2 = 4$, que não é 2, e portanto não serve. Tente $x = 3$, depois $x = 0{,}5$. Você vai ver que só dois valores sobrevivem. Boole usou exatamente esse fato como alicerce: se nosso pensamento lida com “sim ou não”, os números certos para descrevê-lo são 1 e 0.

Capítulo 4: E, OU, NÃO, a álgebra booleana no dia a dia

Você já usa a álgebra de Boole, e provavelmente a usou hoje pela manhã. Quando disse “quero café E pão”, quando decidiu levar o guarda-chuva porque “vai chover OU o céu está fechado”, quando resolveu sair de casa porque “NÃO está chovendo”. Este capítulo não ensina nada que você não faça. Ele apenas dá nome ao que já está na sua cabeça e mostra por que três palavras tão pequenas são suficientes para fazer um computador funcionar.

Três palavras e uma regra

A regra é que toda afirmação pode ser verdadeira ou falsa. Chamamos o verdadeiro de 1 e o falso de 0, como no capítulo anterior. As três operações combinam afirmações assim.

O E exige que as duas afirmações sejam verdadeiras. “Hoje chove E faz frio” só é verdadeira nos dias frios e chuvosos. Basta uma das duas falhar para a frase inteira falhar.

O OU exige que pelo menos uma seja verdadeira. “Quero morango OU banana” está satisfeita se vier morango, se vier banana, ou se vierem as duas. Aqui a língua portuguesa atrapalha, porque “ou” às vezes quer dizer “uma só das duas”, como em “ou você vem ou eu vou”. A álgebra de Boole usa o OU que aceita as duas, e deixa o outro como uma operação à parte, chamada OU exclusivo.

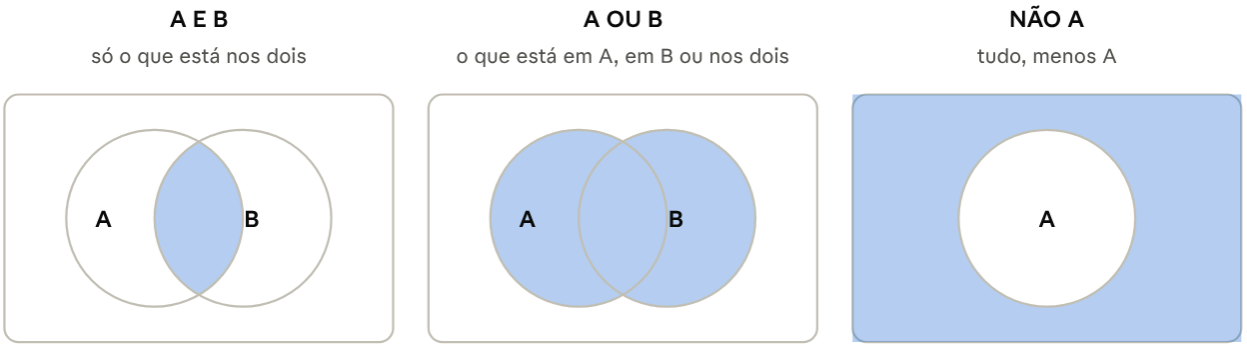
O NÃO inverte: o que era verdadeiro vira falso e o que era falso vira verdadeiro. “NÃO está chovendo” é o oposto de “está chovendo”.

A tabela-verdade resume tudo isso, listando o resultado de cada operação para cada combinação de valores. Ela é aquela que aparece no roteiro deste capítulo, mais acima. Vale a pena olhá-la com calma, linha por linha, porque o resto do livro depende dela.

Desenhando as três operações

Uma forma de enxergar o E, o OU e o NÃO é por meio de círculos. Imagine que A e B são dois grupos de pessoas, cada um desenhado como um círculo. O E pega só quem está nos dois grupos ao mesmo tempo, a região em que os círculos se sobrepõem. O OU pega todo mundo que está em pelo menos um dos grupos. O NÃO pega todo mundo que está fora do grupo. Esses desenhos tomam o nome de John Venn, que os popularizou em 1880, e voltaremos a ele no capítulo 5.

Três operações, três regiões pintadas



Desenho original para este livro. A área pintada é a resposta de cada operação.

Figura 4.1 · diagramas de Venn de E, OU e NÃO

Interruptores: onde a lógica vira eletricidade

Há uma segunda forma de enxergar o E e o OU, e é a que vai mudar o mundo. Imagine uma pilha, uma lâmpada e dois interruptores. Se você liga os interruptores um depois do outro, na mesma fila, a corrente só chega à lâmpada quando os dois estão ligados. Isso é um E. Se

você os liga lado a lado, em dois caminhos separados, basta um deles estar ligado para a corrente passar. Isso é um OU.

Esse paralelo entre a lógica e a eletricidade parece ingenuamente simples, e é o que Claude Shannon vai usar, em 1937, no capítulo 8. A figura abaixo mostra os dois circuitos com a mesma pilha e a mesma lâmpada.

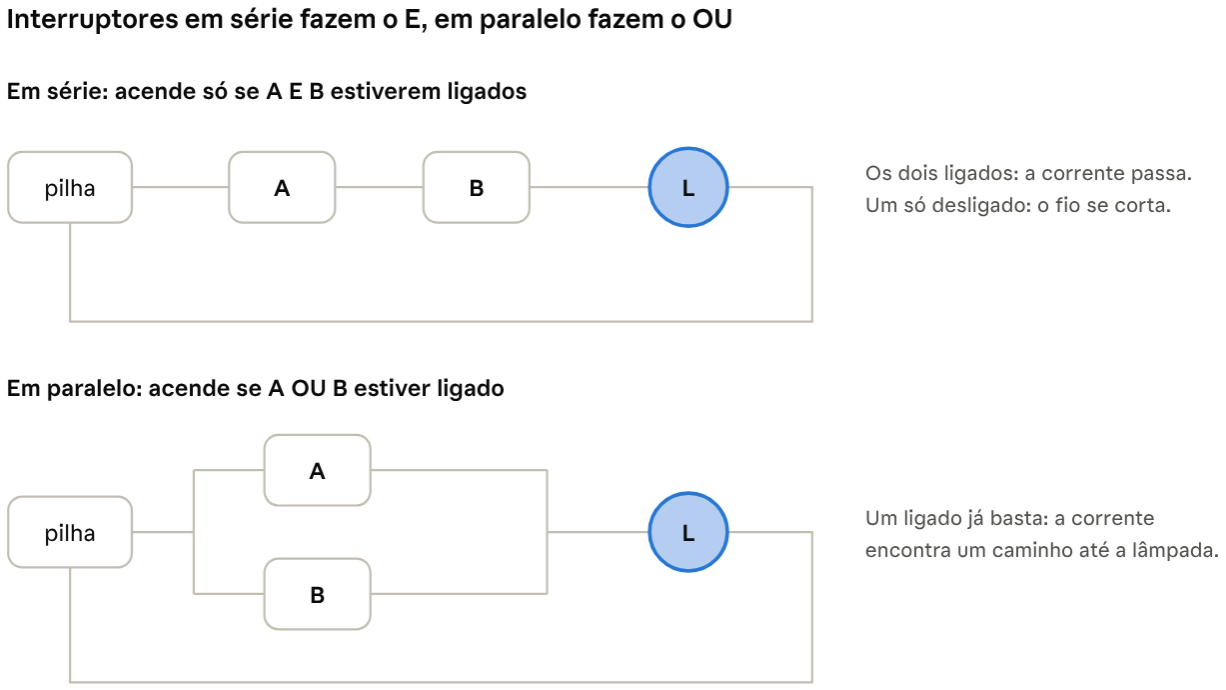


Figura 4.2 · interruptores em série e em paralelo

As leis de De Morgan

Havia uma regra que Augustus De Morgan descreveu e que economiza muito trabalho. Ela diz como negar uma frase inteira. Pense em “NÃO é verdade que choveu E fez frio”. Para isso ser verdade, basta que uma das duas coisas não tenha acontecido: ou não choveu, ou não fez frio. Ou seja, negar um E transforma-o num OU de negativas. A recíproca também vale: negar um OU transforma-o num E de negativas. Dizer que “NÃO é verdade que quero morango OU banana” é o mesmo que dizer que não quero morango E não quero banana.

Os engenheiros usam essas duas leis o tempo todo para simplificar circuitos. Trocar um E por um OU, com negativas nos dois lados, pode economizar peças numa placa que será produzida aos milhões.

Na prática: o filtro de uma busca

Imagine uma pequena biblioteca digital e um filtro que pede “livros infantis E NÃO de terror”. Confira, linha por linha, quais títulos passam.

Título	É infantil?	É de terror?	Passa no filtro?
O Morango Mais Vermelho	sim (1)	não (0)	sim
A Casa Assombrada	sim (1)	sim (1)	não
Guia do Astrônomo	não (0)	não (0)	não
Contos de Sombra	não (0)	sim (1)	não

Só o primeiro passa, porque é o único em que a primeira condição é verdadeira (infantil) e o NÃO da segunda também (não é de terror). Parece pouca coisa, mas é exatamente a conta que um site de livros faz, milhões de vezes por dia, quando você marca as caixinhas de um filtro.

Capítulo 5: De Morgan, Jevons, Venn e Peirce, a lógica ganha diagramas e máquinas

Imagine uma sala vitoriana, iluminada a gesso, numa tarde londrina de 1870. Cavalheiros de casaca e senhoras de vestido comprido se aglomeram em volta de um móvel estranho: parece um piano pequeno, mas não tem cordas nem pedais. Em vez disso, tem teclas marcadas com letras e sinais, e acima delas uma pequena parede de vidro com tiras que sobem e descem. O dono do móvel, um economista magro chamado William Stanley Jevons, aperta três teclas com calma e espera. Uma por uma, as tiras se movem. Num painel, aparece a resposta de um problema de lógica. A plateia aplaude. Nunca se havia visto uma máquina que raciocinasse, ainda que de maneira muito simples.

Este capítulo trata das décadas que vieram depois de Boole, entre sua morte, em 1864, e o final do século XIX. Foi um período de ajustes, de nomes e de máquinas curiosas. Nenhuma delas virou o computador que conhecemos, mas cada uma foi uma peça do caminho.

De Morgan, o amigo de Boole

Augustus De Morgan nasceu na Índia, em 1806, e foi o primeiro professor de matemática da Universidade de Londres, uma instituição nova que aceitava estudantes de qualquer religião. Era um homem espirituoso, que adorava curiosidades numéricas e escreveu cartas para metade da Europa. É dele a pergunta engraçada de qual era a sua idade, que ele respondia dizendo que tinha x anos no ano x ao quadrado.

De Morgan trocou muitas cartas com Boole e foi um dos primeiros a perceber a importância do seu trabalho. A contribuição que ficou com seu nome é a dupla de regras que vimos no capítulo anterior: negar um E vira um OU de negativas, e negar um OU vira

um E de negativas. Elas parecem pequenas, mas hoje estão em cada manual de eletrônica digital. De Morgan foi também professor da jovem Ada Lovelace, personagem do próximo capítulo.

Jevons e o piano lógico

William Stanley Jevons, nascido em 1835, é mais conhecido como economista, mas tinha uma segunda paixão: a lógica de Boole. Ele achava a álgebra do mestre brilhante, mas pesada. Uma das mudanças que propôs foi simplificar a operação OU, de modo a aceitar também o caso em que as duas partes valem, que é exatamente o OU que usamos até hoje. A outra foi construir uma máquina.

O piano lógico, apresentado ao público por volta de 1870, funcionava com a ideia que a figura abaixo explica. Você digitava uma premissa nas teclas, como “todo A é B”. A máquina mantinha uma lista com todas as combinações possíveis das letras. Cada premissa digitada eliminava as combinações proibidas. No fim, o painel mostrava só o que sobrava, e daí se tirava a conclusão. Não era uma máquina de pensar, era uma máquina de eliminar, e fazia isso muito mais rápido do que uma pessoa, sem esquecer nenhuma combinação.

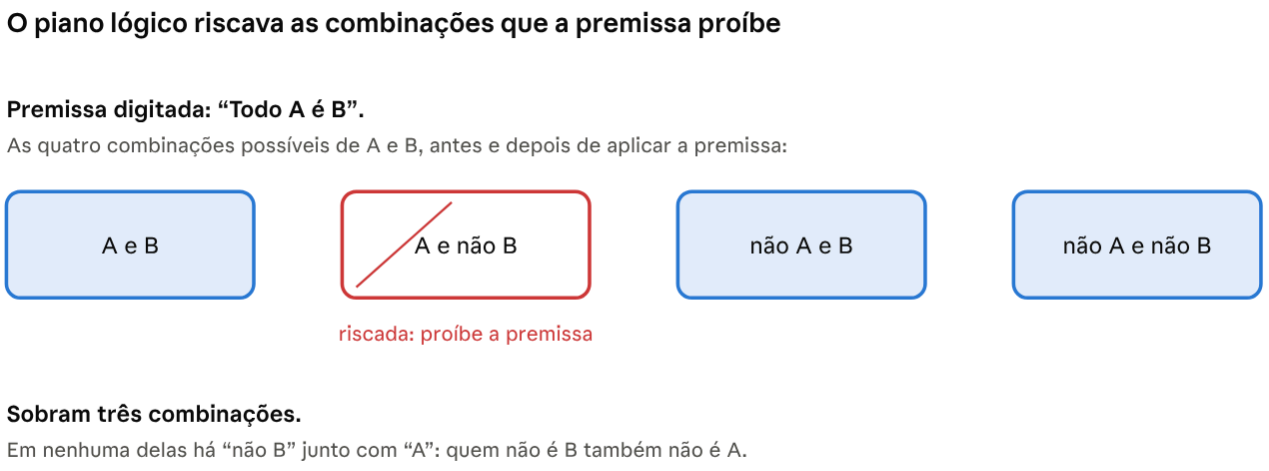


Figura 5.1 · como o piano lógico eliminava combinações

O piano de Jevons tinha limites sérios: tratava de poucas letras ao mesmo tempo e se confundia com facilidade. Mas é historicamente importante porque mostrou, diante de uma plateia, que a lógica de Boole podia ser executada por um aparelho mecânico sem nenhum humano pensando no meio do caminho.

Venn e seus círculos

John Venn, nascido em 1834, era um clérigo e professor de Cambridge que se interessava por lógica. Em 1880 publicou um artigo propondo uma forma de desenhar as ideias de Boole: dois ou três círculos que se sobrepõem, cada um representando um grupo, com as

regiões de encontro pintadas conforme a pergunta. Desenhos parecidos já tinham sido feitos antes, por Euler, mas Venn os tornou sistemáticos, e por isso levam o seu nome.

A grande vantagem dos diagramas é que eles tornam visual o que a álgebra deixa abstrato. Olhe a figura 5.2, que mostra duas frutas e as pessoas que gostam delas. A região do meio é o E, as três regiões juntas são o OU, e o espaço de fora é quem não gosta de nenhuma. Quem entendeu o diagrama entendeu as duas operações.

O E é a região do meio; o OU são as três regiões dos círculos

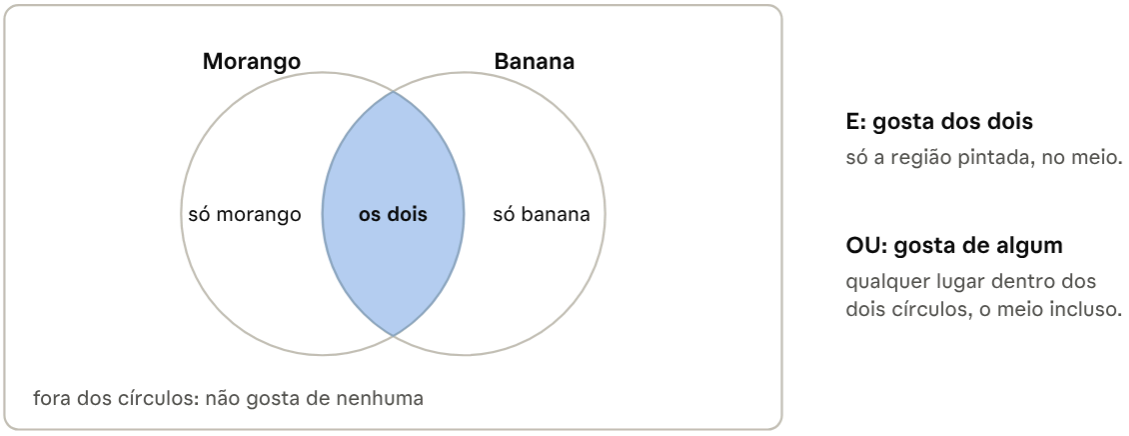


Figura 5.2 · Venn de duas frutas: E e OU

Peirce e a ideia esquecida

Charles Sanders Peirce foi um filósofo e cientista americano, e talvez a mente mais original desta história. Estudou a álgebra de Boole com cuidado, aperfeiçoou sua notação e descobriu coisas que ninguém tinha visto. Mas o que mais importa para nós é uma carta que ele escreveu em 1886 a um ex-aluno, Allan Marquand, que tinha construído uma máquina lógica parecida com a de Jevons. Peirce sugeriu que a máquina fosse refeita com circuitos elétricos em vez de peças mecânicas, e explicou que interruptores em série e em paralelo poderiam fazer o trabalho da lógica.

Era exatamente a ideia que Shannon teria meio século depois. A carta de Peirce, porém, ficou nos arquivos, e nenhum engenheiro da época a levou adiante. Ao longo do livro, vamos ver com frequência essa lei do esquecimento: uma ideia boa demais para o seu tempo pode ficar dormindo por décadas até que alguém a reencontre, sem saber que ela já existia.

Na prática: pinte as regiões

Desenhe dois círculos que se sobrepõem, um para quem gosta de morango e outro para quem gosta de banana. Escreva “só morango”, “os dois” e “só banana” nas três regiões. Agora pinte de azul a região do E e de vermelho a região do OU. Você vai perceber que o

azul está dentro do vermelho, e isso quer dizer, em lógica, que quem satisfaz o E também satisfaz o OU, mas não o contrário.

Capítulo 6: Babbage e Ada Lovelace, a máquina que calculava e a ideia do programa

Em junho de 1833, numa casa elegante de Londres, uma festa de salão reuniu políticos, cientistas e escritores em torno de uma curiosidade: um pequeno pedaço de metal polido, do tamanho de uma caixa de sapatos, que o dono da casa guardava numa vitrine. Quando o anfitrião, um homem de olhar vivo chamado Charles Babbage, girava uma manivela, as engrenagens se moviam e uma coluna de números aparecia nas rodas. Era um pedaço da Máquina Diferencial, que ele sonhava construir em tamanho real. Entre os convidados, uma menina de dezessete anos olhava com uma atenção fora do comum. Chamava-se Augusta Ada Byron, e tudo que ela perguntou naquela noite revelou que tinha entendido como a máquina funcionava. Babbage ficou impressionado, e os dois passaram a trocar cartas por quase vinte anos.

Babbage e a obsessão com os erros

Charles Babbage, nascido em 1791, era matemático, inventor e, segundo muitos, uma pessoa difícil. O que o movia era um ódio quase pessoal aos erros de cálculo. Naquela época, as tabelas de logaritmos, usadas por navegadores e engenheiros, eram calculadas à mão por equipes de pessoas e continham erros que podiam fazer um navio errar o caminho. Babbage queria uma máquina que calculasse e imprimisse as tabelas sem nenhum erro de transcrição.

A solução foi a Máquina Diferencial, que ele começou a construir por volta de 1822 com dinheiro do governo britânico. Ela usava um método curioso, chamado diferenças finitas, que permite calcular uma tabela inteira usando só somas. A tabela abaixo mostra isso com os quadrados dos números: a segunda coluna é a diferença entre quadrados vizinhos, e a terceira é a diferença entre essas diferenças, que fica sempre igual a 2. Se você sabe que a terceira coluna é constante, pode gerar todos os quadrados com somas, sem nunca multiplicar.

Número	Quadrado	1ª diferença	2ª diferença
1	1		
2	4	3	
3	9	5	2
4	16	7	2

Número	Quadrado	1ª diferença	2ª diferença
5	25	9	2
6	36	11	2

O projeto, porém, emperrou. As peças exigiam uma precisão de usinagem que não existia na época, os custos explodiram e as brigas com o engenheiro responsável pela fabricação azedaram o clima. Por volta de 1833 e 1834, o trabalho parou, e o governo desistiu de financiá-lo. A máquina nunca foi terminada nos anos de Babbage.

A Máquina Analítica: o projeto que mudou tudo

Babbage, porém, não parou. A partir de 1834 e, de forma mais clara, em 1837, ele pensou numa máquina muito mais ambiciosa. A Diferencial fazia um único tipo de conta. A nova máquina, a Analítica, faria qualquer conta, na ordem que o operador mandasse. Para isso ele imaginou quatro peças, que a figura abaixo compara com as de um computador moderno: um armazém, onde os números ficam guardados; um moinho, que faz as contas; cartões perfurados, que dizem quais operações fazer; e uma impressora, que mostra o resultado. Os cartões, como veremos no capítulo 7, vieram dos teares de Jacquard.

Em 1837, Babbage já desenhava as quatro peças de um computador

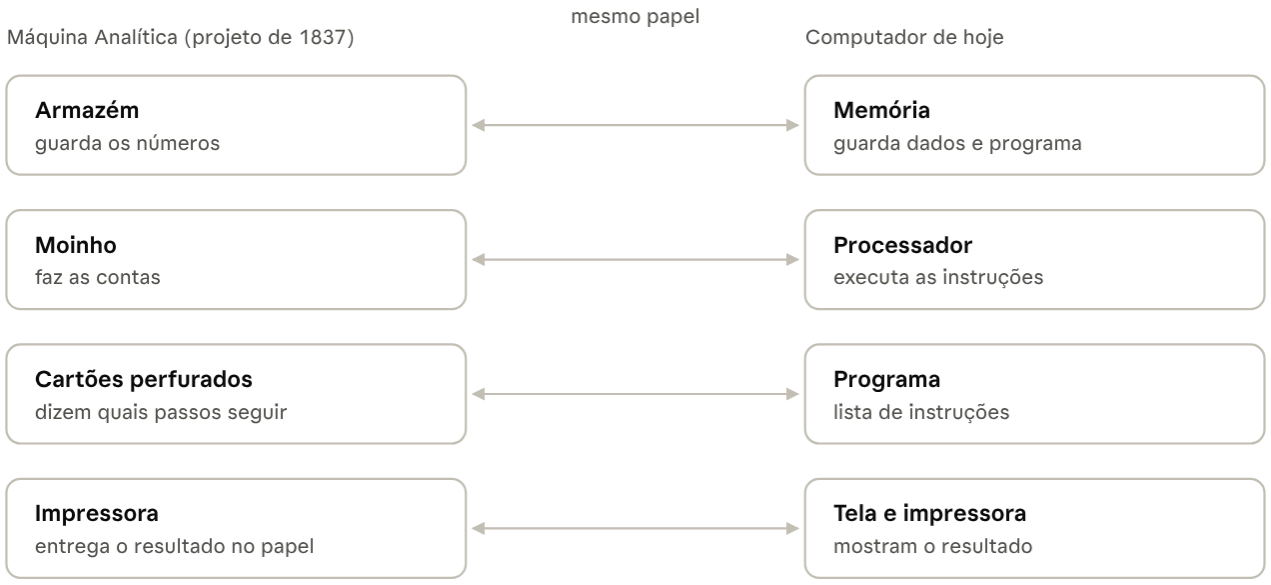


Figura 6.1 · as quatro partes da Máquina Analítica e as do computador de hoje

A Máquina Analítica nunca foi construída. Seria enorme, movida a vapor e com milhares de engrenagens. Mas o projeto está todo desenhado, e muitos historiadores acham que, se tivesse sido feita, funcionaria. O fato é que Babbage, sozinho, antecipou em mais de cem anos a organização de todo computador moderno.

Ada, a condessa que viu mais longe

Ada Byron nasceu em dezembro de 1815, filha do poeta Lord Byron, que deixou a família quando ela tinha poucas semanas de vida. A mãe, Annabella, temendo que a filha herdasse o temperamento do pai, insistiu que ela estudasse matemática e ciência. O plano deu certo de um modo que a mãe talvez não previsse: Ada ficou fascinada, teve professores excelentes, entre eles Augustus De Morgan, e na vida adulta se tornou a condessa de Lovelace.

Em 1842, um engenheiro italiano chamado Luigi Menabrea publicou, em francês, um artigo descrevendo a Máquina Analítica, a partir de uma palestra de Babbage. Ada traduziu o texto para o inglês e, por sugestão de Babbage, acrescentou notas próprias. Quando o trabalho saiu, em 1843, as notas eram bem mais longas que o artigo original, e continham ideias que nem Babbage tinha expressado.

A mais famosa, chamada Nota G, descreve passo a passo como a máquina calcularia uma sequência de números conhecidos como números de Bernoulli. Muita gente a considera o primeiro programa de computador publicado. Mas a contribuição mais profunda de Ada está numa observação geral. Ela percebeu que a máquina trabalhava com símbolos, e que esses símbolos não precisavam representar números. Se fosse possível descrever, por exemplo, as relações entre sons de uma melodia, a máquina poderia compor música. Ela escreveu isso em 1843, mais de cento e vinte anos antes de existir um computador pessoal.

Ada morreu em 1852, aos 36 anos, de câncer. Babbage viveu até 1871. Nenhum dos dois viu a máquina funcionar. A prova de que o projeto estava certo só veio em 1991, quando o Science Museum de Londres, seguindo os desenhos de Babbage, construiu a Diferencial nº 2, que calcula corretamente até hoje.

Dois caminhos que ainda não se encontravam

Repare numa coisa estranha. Babbage e Ada tinham a máquina de calcular, mas não tinham a lógica de Boole. Boole tinha a lógica, mas não tinha a máquina. Eles viveram na mesma época, na mesma ilha, e não há registro de que tenham trabalhado juntos. As duas histórias correm em paralelo por um século, até que um jovem engenheiro do outro lado do Atlântico perceba que as duas conversam.

Na prática: escreva uma receita para uma máquina que só obedece

Imagine que você é Ada e precisa dar instruções a uma máquina que só sabe somar, guardar e imprimir. Escreva, em passos numerados e sem nenhuma ambiguidade, a receita para calcular 3 vezes 4, usando só somas. Uma resposta: (1) guarde 0 no lugar do

resultado; (2) some 4 ao resultado; (3) some 4 ao resultado; (4) some 4 ao resultado; (5) imprima o resultado. Você acabou de escrever um programa, e a lista de passos acima é o que Ada chamaria de um conjunto de operações.

Capítulo 7: Cartões perfurados, Jacquard, Hollerith e o censo

Num escritório de Washington, no verão de 1890, centenas de funcionários, quase todas mulheres jovens, estavam sentadas diante de pequenas máquinas de ferro. Cada uma tinha ao lado uma pilha de cartões de papel rígido, do tamanho de uma nota de dólar da época, e um furador a que chamavam pantográfico. Com os olhos num formulário, elas apertavam o furador até abrir, no cartão, os furos que diziam a idade, o sexo, a profissão e o lugar de nascimento de uma pessoa. Um cartão, um habitante. Noutra sala, uma máquina com mostradores lia esses cartões: pinos metálicos desciam sobre cada um, atravessavam os furos, tocavam pequenas cubas de mercúrio e fechavam um circuito elétrico, e um ponteiro avançava um número. Era a primeira vez que um país inteiro seria contado por uma máquina.

Um tear que guardava desenhos

Para entender como os cartões chegaram até ali, é preciso voltar quase um século no tempo, à cidade francesa de Lyon, famosa pela seda. Os tecidos de desenho complicado, como os brocados, exigiam que os tecelões levantassem os fios da urdidura numa ordem precisa, e isso era trabalho de duas pessoas: o tecelão e um ajudante, geralmente uma criança, que puxava cordinhas. Um erro estragava metros de pano.

Em 1804, Joseph-Marie Jacquard patenteou um tear que resolvia o problema com cartões de papel rígido, ligados uns aos outros numa longa fita. Cada fileira de furos de um cartão correspondia a uma linha do tecido. Onde havia um furo, uma agulha passava e um gancho levantava o fio. Onde não havia, a agulha batia no papel e o fio ficava. A figura abaixo mostra a cadeia. O desenho inteiro do tecido ficava guardado na fita, e para trocar de desenho bastava trocar a fita.

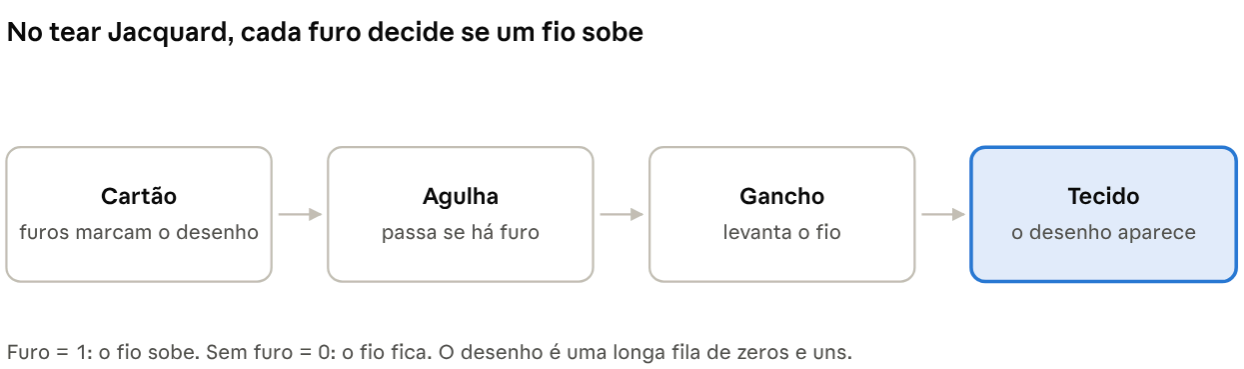


Figura 7.1 · do cartão ao tecido no tear Jacquard

Os tecelões tradicionais se revoltaram, temendo perder o emprego, e houve quebra-quebra. Mesmo assim, em poucas décadas, milhares de teares Jacquard estavam em uso pela Europa. E aqui está o fato que nos interessa: aquela fita de cartões era uma sequência de zeros e uns, furo ou não furo, que mandava numa máquina o que fazer. O programa e o aparelho estavam separados. Como vimos no capítulo anterior, Babbage viu um retrato de Jacquard tecido nessa técnica e copiou a ideia dos cartões para a sua Máquina Analítica.

O censo que não terminava

A segunda parte da história passa-se do outro lado do Atlântico. Nos Estados Unidos, a Constituição manda contar a população a cada dez anos, e a população crescia rápido, com a imigração. O censo de 1880, apurado à mão, levou cerca de sete anos para ficar pronto. Ou seja, quando os resultados saiam, já estavam velhos, e estava claro que o censo de 1890 corria o risco de não terminar antes do de 1900.

Herman Hollerith, um engenheiro alemão-americano nascido em 1860, tinha trabalhado no censo de 1880 como jovem auxiliar e fixou na cabeça a ideia de que aquilo deveria ser feito por máquinas. Segundo uma anedota famosa, a inspiração veio de um condutor de trem que furava, no bilhete, uma descrição do passageiro, como cor do cabelo e altura. Hollerith pensou: por que não furar, num cartão, as características de cada pessoa?

O sistema que ele desenhou tinha três partes: o furador, a leitora de cartões com seus pinos e cubas de mercúrio, e uma caixa de triagem que separava os cartões por categoria. Em 1890, o governo americano o adotou. O resultado foi espetacular. A tabela abaixo compara os dois censos, com números aproximados que precisam ser conferidos.

Censo	População (aprox.)	Como foi apurado	Tempo aproximado
1880	50 milhões	à mão	cerca de 7 anos
1890	63 milhões	cartões perfurados	de 2 a 3 anos

O sucesso de Hollerith criou uma indústria. Ele fundou a sua empresa, que depois de várias fusões, em 1924, passou a se chamar International Business Machines, a IBM. Por décadas, cartões perfurados foram a memória e a entrada de dados dos computadores, até os anos 1970.

Um cartão é uma fila de bits

Pense no cartão como uma tabela. Cada posição pode ter furo ou não ter furo, e portanto guarda um bit. Escolha um código, como atribuir a cada letra do alfabeto o seu número de ordem, de 1 a 26, escrito em binário com cinco posições, e você guarda texto num pedaço de papel. A figura abaixo mostra a palavra BOOLE perfurada assim. Repare como cada letra ocupa cinco posições, com pesos 16, 8, 4, 2 e 1, o mesmo esquema das lâmpadas do capítulo 1.

Furo ou não furo: a palavra BOOLE em 25 bits

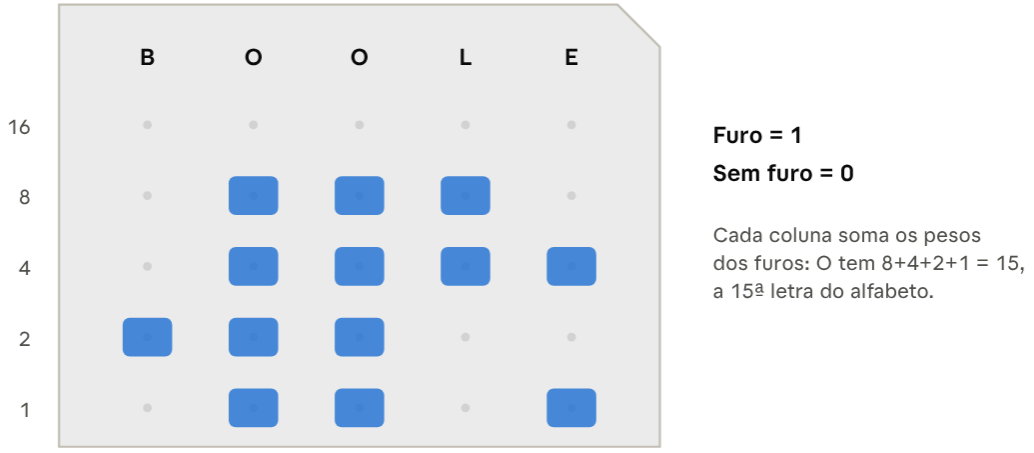


Figura 7.2 · a palavra BOOLE em um cartão de furos

Na prática: seu nome em furos

Use a tabela de letras abaixo e escreva o seu primeiro nome em cinco posições por letra. Num pedacinho de papel, faça um furo onde houver 1 e deixe inteiro onde houver 0. Troque de papel com um amigo e peça que ele decifre. O código: A=00001, B=00010, C=00011, D=00100, E=00101, F=00110, G=00111, H=01000, I=01001, J=01010, K=01011, L=01100, M=01101, N=01110, O=01111, P=10000, Q=10001, R=10010, S=10011, T=10100, U=10101, V=10110, W=10111, X=11000, Y=11001, Z=11010.

Capítulo 8: Claude Shannon, quando a lógica encontrou os relés

No outono de 1936, um jovem de vinte anos entrou todos os dias, durante semanas, numa sala do Instituto de Tecnologia de Massachusetts, o MIT, onde um aparelho do tamanho de um quarto zumbia, estalava e cheirava a óleo. Era o analisador diferencial, a maior máquina de calcular da época, projetada pelo engenheiro Vannevar Bush. Parte do seu cérebro era feita de relés: pequenas chaves eletromagnéticas que estalavam a cada decisão. O rapaz olhava para aquela parede de estalos, e na cabeça dele ressoava um livro que lera numa disciplina de filosofia da universidade, um livro de 1854, escrito por um professor de Cork, que falava de uma álgebra do pensamento. Oitenta e três anos separavam o livro dos relés. O rapaz chamava-se Claude Shannon, e ele foi o primeiro a perceber que os dois falavam da mesma coisa.

Um jovem de Michigan que gostava de construir

Claude Elwood Shannon nasceu em 1916, numa cidadezinha de Michigan, nos Estados Unidos. Cresceu mexendo em rádios, montando pequenos aparelhos e até um telegráfico entre a casa dele e a de um amigo, usando as cercas de arame do pasto como fios. Formou-se, em 1936, em duas áreas ao mesmo tempo, engenharia elétrica e matemática, e foi para o MIT trabalhar como assistente de pesquisa de Bush. Ele nunca foi o tipo que escolhia entre o abstrato e o prático, e essa mistura foi decisiva.

Relés: o interruptor que se liga sozinho

Para entender a descoberta de Shannon, é preciso saber o que é um relé. A figura abaixo mostra: é um interruptor que não é apertado por um dedo, mas por um ímã elétrico. Quando passa corrente numa bobina de fio, ela vira um ímã, puxa uma lâmina de metal e fecha o contato de outro circuito. Resultado: uma corrente fraca controla uma outra, e a saída de um relé pode ser a entrada de outro. Redes telefônicas inteiras eram feitas assim, com milhares de relés ligados por projetistas que tentavam descobrir, no tentativa e erro, o menor jeito de montar cada circuito.

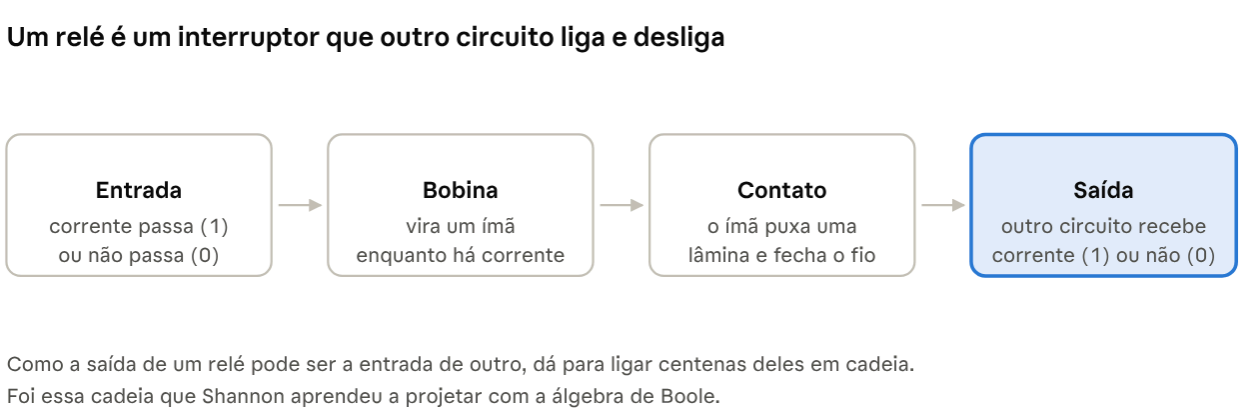


Figura 8.1 · o que faz um relé

A tese: o encontro de duas ideias

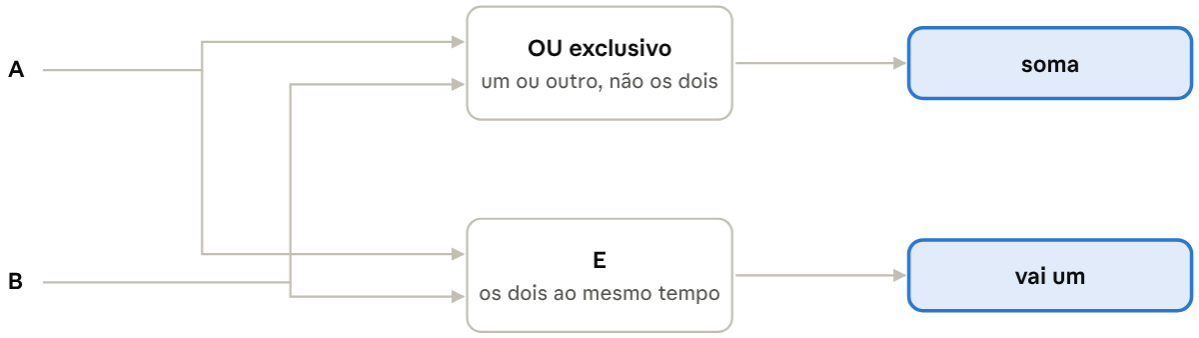
Shannon notou que um circuito de relés pode ser descrito por uma frase de Boole. Interruptores em série, ligados um depois do outro, deixam passar corrente apenas se todos estiverem ligados: um E. Interruptores em paralelo deixam passar se pelo menos um estiver ligado: um OU. Um relé que desliga quando recebe corrente faz o NÃO. Com isso, qualquer circuito virava uma expressão de álgebra, e qualquer expressão de álgebra virava um circuito.

A consequência prática era enorme. Se um projetista tem uma expressão complicada, pode simplificá-la com as regras de Boole e De Morgan e obter um circuito com menos relés, que faz a mesma coisa. Em vez de tentativa e erro, o projeto passava a ser uma conta. Em 1937, Shannon escreveu isso na sua dissertação de mestrado, intitulada, em tradução livre, *Análise Simbólica de Circuitos de Relés e de Chaveamento*. Muita gente a considera a dissertação de mestrado mais importante já escrita.

Como somar com portas

O exemplo mais bonito da tese é a soma. A figura abaixo mostra como somar dois bits usando só duas portas. Uma porta OU exclusivo dá o bit da soma: ela vale 1 quando um dos dois, e só um, vale 1. Uma porta E dá o “vai um”: ela vale 1 só quando os dois valem 1. A tabela mostra os quatro casos.

Um circuito soma dois bits com apenas duas portas



Desenho original para este livro. Dois bits de entrada, dois bits de saída: 1 + 1 = 10 em binário.

Figura 8.2 · somador de um bit com duas portas

A	B	Soma	Vai um	Em binário
0	0	0	0	0 + 0 = 00
0	1	1	0	0 + 1 = 01
1	0	1	0	1 + 0 = 01
1	1	0	1	1 + 1 = 10

O último caso é o mais importante: um mais um dá dois, que em binário é 10, com o zero na casa da soma e o um na casa do vai-um. Encadeando muitos desses somadores, um circuito soma números de quantos bits quiser. Todo computador faz isso bilhões de vezes por segundo.

Depois da tese: a informação e o bit

Shannon não parou na tese. Trabalhou nos Laboratórios Bell e, em 1948, publicou um artigo sobre a matemática da comunicação, em que mostrou que qualquer mensagem, seja texto, som ou imagem, pode ser medida em unidades de informação. A unidade mais simples, a resposta a uma pergunta de sim ou não, ficou conhecida como bit, palavra cuja autoria costuma ser atribuída ao estatístico John Tukey. Com isso, Shannon deu uma conta para o que antes era vago: quanta informação cabe num cabo, num disco ou numa mensagem.

Fora do trabalho, Shannon era uma figura excêntrica e divertida. Andava de monociclo pelos corredores, fazia malabarismo e construiu, nos anos 1950, um pequeno rato mecânico movido a relés que aprendia a sair de labirintos. Morreu em 2001, depois de anos afetado pela doença de Alzheimer.

Na prática: dois interruptores e uma lâmpada

Desenhe, num papel, uma pilha, uma lâmpada e dois interruptores, A e B. Primeiro ligue os interruptores em série, na mesma linha. Pergunte: em que casos a lâmpada acende? (Resposta: só quando A e B estão ligados.) Agora refaça o desenho com os interruptores em paralelo, em duas linhas lado a lado. Em que casos acende? (Resposta: quando A, ou B, ou os dois estão ligados.) Você acabou de reproduzir a descoberta de Shannon, e a figura 4.2 do capítulo 4 mostra a resposta.

Capítulo 9: Alan Turing e a máquina universal

Segundo uma história contada por um amigo, num dia de verão de 1935, um jovem de Cambridge saiu para correr e, cansado, deitou-se na grama de um campo perto da cidade. Tinha vinte e poucos anos, era magro, de cabelo emaranhado, e carregava uma pergunta que um matemático alemão, David Hilbert, lançara anos antes. Ali, olhando o céu, ele se perguntou como funciona, de verdade, uma pessoa que faz uma conta. Em vez de pensar no resultado, pensou nos movimentos: olhar um número, anotar outro, trocar de página, lembrar em que passo está. Naquele dia, ou perto dele, Alan Turing teve a ideia de uma máquina que faria exatamente isso e nada além.

A pergunta de Hilbert

Para entender o que Turing fez, é preciso saber o desafio. No começo do século XX, David Hilbert, o matemático mais influente da época, sonhava que toda a matemática pudesse ser colocada em ordem, e que existisse um método mecânico, uma receita, capaz de dizer, para qualquer afirmação matemática, se ela é verdadeira ou falsa. Esse desafio ficou conhecido pelo nome alemão *Entscheidungsproblem*, o problema da decisão.

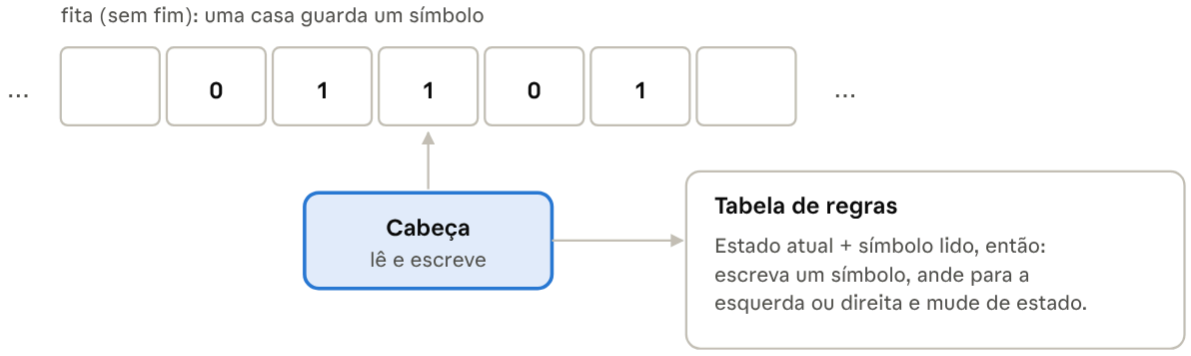
Em 1931, o lógico austríaco Kurt Gödel já tinha abalado o sonho de Hilbert mostrando que, em qualquer sistema de regras razoavelmente rico, há afirmações verdadeiras que o sistema não consegue provar. Mas ainda restava a pergunta sobre a receita: dada uma afirmação, existe um método mecânico que decida se ela é provável? Para responder, faltava uma definição precisa do que seja um “método mecânico”. E foi isso que Turing inventou.

A máquina de papel

Na época, a palavra “computador” (em inglês, *computer*) designava uma pessoa, um profissional que fazia contas à mão. Turing imaginou essa pessoa reduzida ao essencial. Ela tem uma fita de papel dividida em casas, cada uma com um símbolo. Olha uma casa de cada vez, e, de acordo com o que lê e com o estado em que está, escreve um símbolo, anda

uma casa para a esquerda ou para a direita e muda de estado. Só isso. A figura abaixo mostra as três peças: a fita, que se estende para os dois lados sem limite, a cabeça que lê e escreve, e uma tabela de regras.

Uma fita, uma cabeça e uma tabela de regras bastam



Desenho original para este livro, inspirado na máquina descrita por Turing em 1936.

Figura 9.1 · as peças de uma máquina de Turing

Parece pouco, e é pouco. Mas Turing mostrou que qualquer cálculo que um humano consiga fazer com lápis e papel, seguindo regras, pode ser feito por uma máquina assim. Soma, multiplicação, raiz quadrada, qualquer coisa. Para ver na prática, acompanhe a máquina que soma 1 a um número binário. Suas regras cabem em três linhas:

Se lê	Escreve	Anda	Próximo passo
1	0	uma casa para a esquerda	continua
0	1	fica parada	para
casa vazia	1	fica parada	para

A figura a seguir aplica essas regras a 0111, que é o número sete. A cabeça começa no último dígito. Cada 1 vira 0 e a cabeça anda para a esquerda, como o “vai um” da soma, até achar um 0, que vira 1. O resultado, 1000, é o oito.

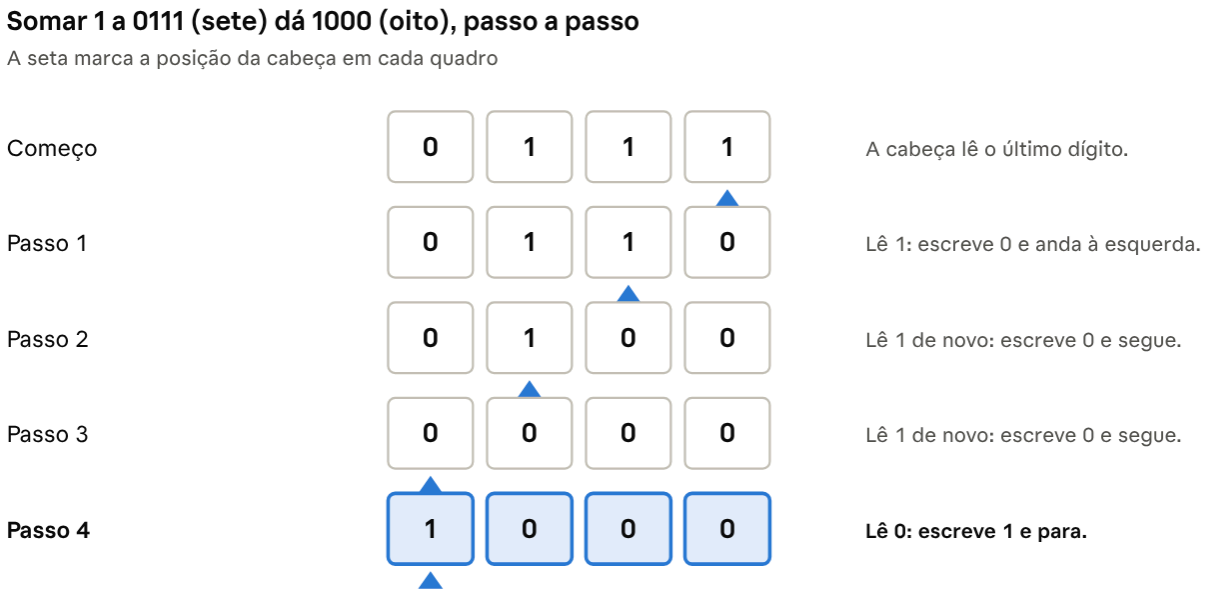


Figura 9.2 · a máquina somando 1 a 0111

A máquina universal

Até aqui, cada máquina de Turing faz uma coisa só: somar, multiplicar, ordenar. Mas o artigo dele, publicado em 1936, tem um segundo golpe, mais audacioso. Turing percebeu que a tabela de regras de uma máquina pode ser escrita como uma sequência de símbolos, e que sequências de símbolos cabem numa fita. Logo, existe uma máquina que lê, na própria fita, a descrição de outra máquina qualquer e passa a se comportar exatamente como ela. Ele a chamou de máquina universal.

Pense no que isso quer dizer. Em vez de uma máquina para cada tarefa, uma única máquina, que muda de papel conforme o programa que recebe. O celular que você tem na mão é isso: o mesmo aparelho é telefone, câmera, calculadora e tocador de música, dependendo do programa que estiver rodando. Turing escreveu esse conceito oito décadas antes dele existir, e sem nunca ter visto um computador eletrônico.

O que nenhuma máquina resolve

O artigo tem ainda um resultado inesperado, e talvez o mais profundo. Turing provou que existem perguntas que nenhuma máquina, por mais poderosa que seja, consegue responder. A mais famosa é saber, antes de rodar, se um programa qualquer vai terminar algum dia ou ficar rodando para sempre. Hoje isso se chama o problema da parada. Com isso, ele respondeu à pergunta de Hilbert: não, não existe a receita universal que o matemático sonhava.

Outro lógico, o americano Alonzo Church, chegou ao mesmo resultado quase ao mesmo tempo, por um caminho totalmente diferente, o cálculo lambda. Quando Church viu o trabalho do jovem inglês, convidou-o para estudar com ele em Princeton, onde Turing ficou entre 1936 e 1938 e fez seu doutorado.

E a lógica de Boole?

O leitor pode perguntar onde está Boole nisto tudo. Repare que a máquina de Turing usa símbolos, regras do tipo “se isso, então aquilo” e dois estados óbvios, ligado ou desligado. As peças de uma máquina real, como veremos, serão as portas lógicas de Shannon, e o que Turing mostrou foi o que é possível calcular com elas. Shannon explicou como construir; Turing explicou o que pode ser construído e, sobretudo, o que não pode.

Na prática: rode a máquina com um lápis

Desenhe numa tira de papel os quatro quadrados 0 1 1 1 (ou escolha outro número binário qualquer). Coloque o dedo no último quadrado. Siga a tabela de regras: se lê 1, apague e escreva 0 e ande para a esquerda; se lê 0, apague e escreva 1 e pare. Tente com 0101 (cinco) e com 0011 (três). Você deve chegar a 0110 e a 0100. Confira se está certo contando em binário como no capítulo 1.

Capítulo 10: Zuse, Atanasoff e Aiken, os primeiros computadores

Numa noite de novembro de 1937, um engenheiro dos Laboratórios Bell chamado George Stibitz voltou do trabalho com um punhado de relés que tinha pegado de uma caixa de peças descartadas. Em casa, na mesa da cozinha, com uma lata de tabaco, algumas lanternas de bolso e um pouco de fio, ele montou um circuito. Quando terminou, apertou dois botões, que representavam dois números binários, e uma pequena lâmpada acendeu ou ficou apagada, mostrando a soma. A mulher dele, Evelyn, olhou aquela geringonça e a chamou de Modelo K, de “kitchen”, cozinha em inglês. Era um somador binário feito de relés: o circuito do capítulo anterior, construído de verdade.

Quem foi o primeiro?

A pergunta “quem inventou o computador?” parece simples e não tem resposta única. A razão é que um computador moderno tem várias características, e cada pioneiro acertou em algumas e errou em outras. Ele precisa calcular em binário, ser eletrônico, ser programável e guardar o programa na própria memória. Nenhuma das máquinas deste capítulo tem tudo isso. A figura abaixo as coloca numa linha do tempo, e a tabela, mais acima, resume o que cada uma tinha.

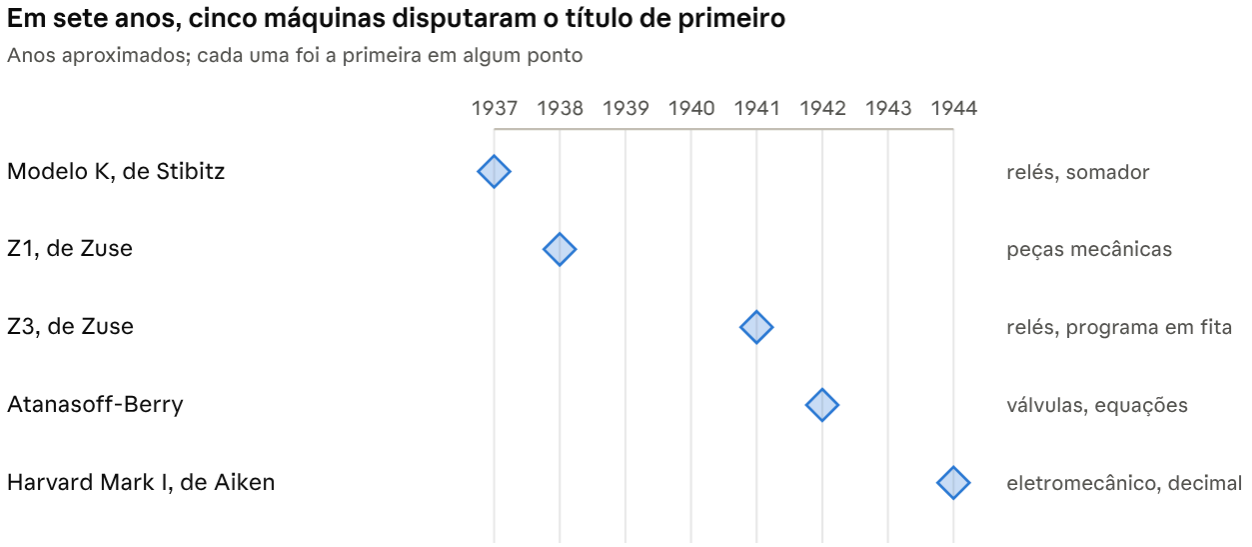


Figura 10.1 · cinco máquinas pioneiras, de 1937 a 1944

Konrad Zuse, o engenheiro de Berlim

Konrad Zuse nasceu em 1910 e estudava engenharia civil em Berlim. O trabalho dele consistia em calcular estruturas, e esse cálculo repetitivo o entediava a tal ponto que ele resolveu construir uma máquina que o fizesse por ele. Sem conhecer Babbage nem Boole, chegou às mesmas conclusões: usar binário, separar a memória do cálculo e controlar tudo por uma fita de instruções.

O Z1, construído de 1936 a 1938 na sala de estar dos pais dele, era puramente mecânico, feito de finas chapas de metal que deslizavam umas sobre as outras para representar zeros e uns. Era temperamental e travava. O Z3, concluído em 1941, usava relés e lia o programa de uma fita perfurada feita de filme fotográfico antigo. Muitos historiadores o consideram o primeiro computador programável funcional do mundo. O Z3 foi destruído num bombardeio a Berlim, em 1943, e só se conhece por uma réplica construída depois da guerra.

Atanasoff e Berry, os computadores de válvulas

Nos Estados Unidos, o professor John Atanasoff, da universidade estadual de Iowa, queria resolver sistemas enormes de equações, e achava as calculadoras da época lentas demais. Com seu aluno Clifford Berry, construiu, entre 1939 e 1942, uma máquina que usava válvulas eletrônicas e calculava em binário, a que chamaram Atanasoff-Berry Computer, ou ABC. O ABC era eletrônico, o que o tornava muito mais rápido que os relés. Mas fazia uma única coisa, resolver equações, e não era programável.

A história do ABC ficou famosa por um motivo curioso: numa disputa de patentes que terminou nos anos 1970, um juiz americano concluiu que a patente do ENIAC tinha se baseado em ideias de Atanasoff, e a anulou. Hoje, os historiadores continuam a discutir o peso exato dessa influência.

Stibitz e Aiken, dos Bell Labs a Harvard

Stibitz não parou na cozinha. Em 1940, nos Laboratórios Bell, ele construiu uma calculadora de números complexos, e numa demonstração ficou famoso por operação a distância: uma plateia numa reunião em New Hampshire digitou problemas num terminal e recebeu as respostas por linha telefônica, vindas de uma máquina em Nova York. Foi um dos primeiros usos remotos de um computador.

Howard Aiken, físico de Harvard, queria uma máquina que realizasse a ideia de Babbage, cujo projeto ele havia estudado. Convenceu a IBM a financiar o projeto, e o resultado foi o Harvard Mark I, de 1944: uma máquina de mais de quinze metros de comprimento, cheia de relés, eixos e contadores, lendo as instruções de uma fita de papel perfurada. Era um prodígio de engenharia e calculava, em decimal, ao ritmo de um estalo constante e ensurdecedor. Uma das primeiras programadoras do Mark I foi a jovem oficial da Marinha Grace Hopper, que viria a ser uma das grandes figuras da programação.

Na prática: some dois números de dois bits

Some 3 e 1 em binário, como os circuitos fazem. Em binário, o 3 é 11 e o 1 é 01. Comece pela casa da direita: 1 mais 1 dá 10, então escreva 0 e “vai um” para a casa ao lado. Na casa do meio: 1 mais 0 mais o “vai um” dá 10, então escreva 0 e “vai um” de novo. A última casa recebe esse “vai um” e vira 1. A tabela mostra o resultado: 100, que é o 4.

	Casa 2	Casa 1	Casa 0
vai um	1	1	
A (3)		1	1
B (1)		0	1
soma (4)	1	0	0

Cada coluna dessa tabela pode ser feita por um circuito como o da figura 8.2, com a diferença de que a coluna do meio recebe três entradas, e não duas. Os projetistas resolveram isso encadeando dois somadores. Você pode inventar suas próprias somas de dois bits e conferir o resultado contando nos dedos.

Capítulo 11: Guerra e computação, Colossus, ENIAC e as mulheres que programaram

Em 1943, num terreno de uma antiga mansão inglesa a oitenta quilômetros de Londres, centenas de pessoas entravam todas as manhãs em barracões de madeira com uma regra de ferro: cada um sabia apenas a sua parte. O funcionário que recebia as mensagens interceptadas não sabia o que a sala ao lado fazia com elas. A moça que operava uma máquina não sabia de onde vinham os papéis que alimentava. Nem os vizinhos de casa, nem as famílias, nem em muitos casos os próprios cônjuges podiam saber onde trabalhavam. O lugar chamava-se Bletchley Park, e o segredo que ele guardava era um dos maiores da guerra: a Grã-Bretanha estava lendo as mensagens secretas do inimigo.

Por que a guerra pagou pelos computadores

A Segunda Guerra Mundial foi, entre muitas outras coisas, uma guerra de cálculos. Era preciso calcular trajetórias de projéteis, rotas de aviões, resistência de materiais e, sobretudo, descobrir o conteúdo de mensagens cifradas. Governos que nunca teriam financiado uma máquina de calcular em tempos de paz passaram a pagar pelas mais caras e arriscadas. O resultado foi que, em poucos anos, a eletrônica de válvulas deixou de ser curiosidade de laboratório para virar tecnologia de grande escala.

Havia, porém, um efeito colateral doloroso para a memória. Boa parte das máquinas britânicas foi mantida em segredo por décadas depois da guerra, e algumas foram desmontadas por ordem do governo. Por isso a história que se conta nos livros de computação das décadas seguintes deixou de fora capítulos inteiros.

Cifras: o que as máquinas tinham de vencer

Uma cifra é uma maneira de embaralhar uma mensagem, e a mais simples é a de substituição: cada letra vira outra, seguindo uma regra. A figura abaixo mostra uma versão antiga e simples, em que cada letra anda três casas à frente no alfabeto. Qualquer pessoa decifra isso com paciência, porque as letras mais comuns da língua, como o A e o E em português, continuam sendo as mais comuns na mensagem cifrada, e isso entrega o segredo.

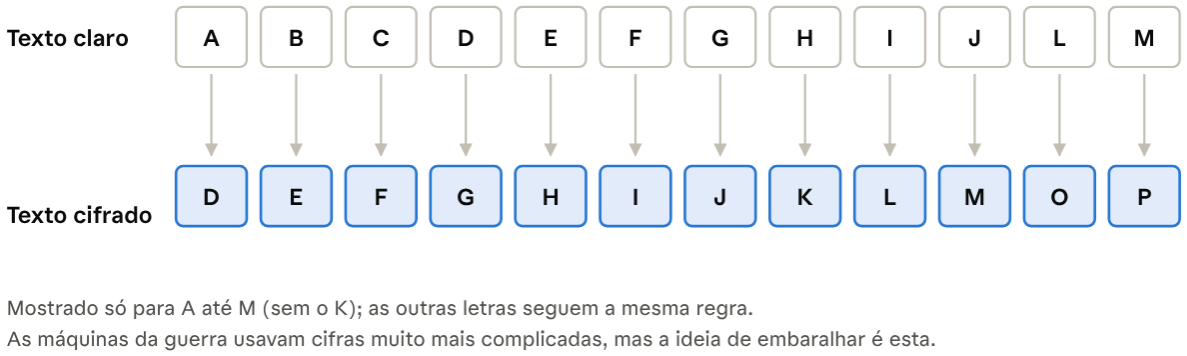


Figura 11.1 · uma cifra de deslocamento de três casas

As cifras alemãs eram muito mais complicadas. A máquina Enigma, usada pelo exército, pela marinha e pela aeronáutica, embaralhava as letras com rotores elétricos que giravam a cada tecla, de modo que a mesma letra virava letras diferentes ao longo da mensagem. Os matemáticos poloneses descobriram os primeiros caminhos para quebrá-la nos anos 1930, e em Bletchley Park, a partir de 1939 e 1940, Alan Turing, Gordon Welchman e outros projetaram a Bombe, uma máquina eletromecânica que testava combinações de rotores em velocidade muito superior à de qualquer pessoa.

Colossus: válvulas contra o código Lorenz

Havia, porém, uma cifra ainda mais difícil, usada pelo alto comando alemão. Os ingleses chamaram-na de Tunny, e ela era produzida por uma máquina chamada Lorenz. Quebrá-la exigia comparar enormes tiras de fita de papel com padrões de bits, um trabalho lentíssimo para máquinas de relés. Um engenheiro dos correios britânicos, Tommy Flowers, teve a ideia ousada de usar válvulas eletrônicas, que eram milhares de vezes mais rápidas que os relés. Os colegas duvidavam, porque as válvulas tinham fama de queimar o tempo todo. Flowers argumentou que elas só queimavam quando ligadas e desligadas, e que, se a máquina nunca fosse desligada, funcionaria.

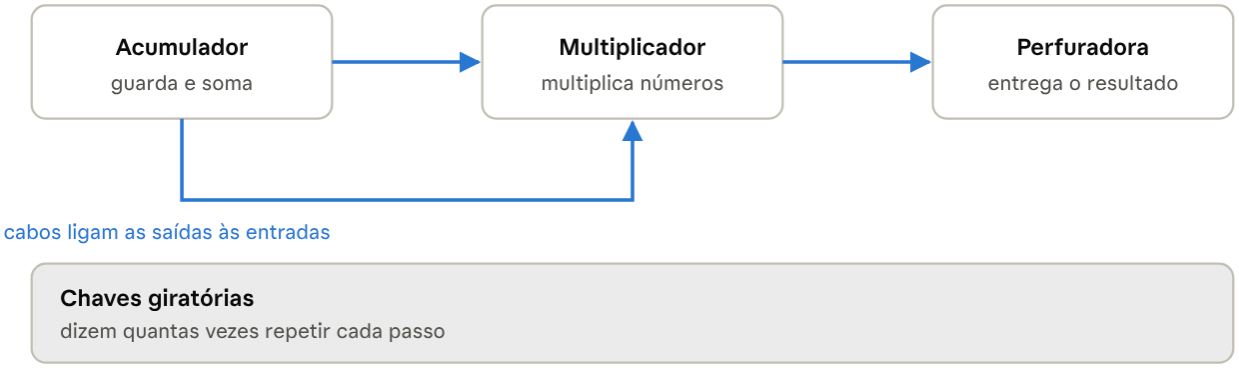
Ele estava certo. O Colossus, que começou a funcionar no final de 1943 ou começo de 1944, tinha cerca de mil e quinhentas válvulas na primeira versão, e uma segunda, de junho de 1944, tinha cerca de duas mil e quatrocentas. O Colossus fazia operações de lógica booleana sobre fitas de bits, e portanto é um parente direto das ideias de Boole e Shannon. Não era um computador de uso geral, porque só servia para esse tipo de tarefa, mas era eletrônico, digital e funcionava. Foram construídos dez exemplares, e a informação que produziram ajudou os Aliados antes do desembarque na Normandia. Depois da guerra, as máquinas foram desmontadas, os projetos destruídos e os envolvidos proibidos de falar. Só nos anos 1970 a história começou a vir a público.

ENIAC: o gigante da Pensilvânia

Do outro lado do oceano, na Universidade da Pensilvânia, o físico John Mauchly e o engenheiro J. Presper Eckert, um jovem de vinte e poucos anos, tinham recebido do exército americano uma missão: construir uma máquina eletrônica que calculasse as tabelas de tiro da artilharia, trabalho que ocupava salas inteiras de calculadoras humanas. O resultado foi o ENIAC, que ficou pronto no final de 1945 e foi apresentado ao público em fevereiro de 1946. Pesava cerca de trinta toneladas, ocupava uma sala enorme e usava quase dezoito mil válvulas. Calculava milhares de somas por segundo, e era muitas centenas de vezes mais rápido que qualquer máquina eletromecânica. Ainda assim, trabalhava em decimal, e não em binário.

O problema era programar. Não havia programa guardado na memória: a máquina era configurada ligando cabos e ajustando chaves em vários painéis, como na figura abaixo. Para mudar de problema, as pessoas passavam dias religando tudo.

No ENIAC, programar era religar cabos e ajustar chaves



Esquema simplificado, desenhado para este livro. Mudar de problema podia levar dias de religação.

Figura 11.2 · como se programava o ENIAC

As seis programadoras

Quem fazia esse trabalho eram seis mulheres, jovens e com formação em matemática, contratadas inicialmente como calculadoras humanas: Kay McNulty, Betty Jennings, Betty Snyder (mais tarde Holberton), Marlyn Wescoff, Fran Bilas e Ruth Lichterman. Elas receberam os diagramas da máquina e tiveram que descobrir, sozinhas, como fazer com que ela resolvesse os problemas. Aprenderam a achar os defeitos, a ligar os cabos certos e a dividir um cálculo em partes. Foram as primeiras programadoras de um computador eletrônico de grande porte.

Na apresentação pública de 1946, o trabalho delas foi pouco mencionado, e em fotografias oficiais da época elas por vezes foram cortadas ou identificadas como modelos. Só décadas depois, graças a pesquisas e documentários, o papel delas voltou ao centro da história.

Na prática: decifre uma mensagem

Use a cifra da figura 11.1. Cada letra do texto cifrado andou três casas à frente; para decifrar, ande três casas para trás. Decifre as duas palavras abaixo e depois invente uma sua.

- ERROH (resposta: BOOLE).
- ORJLFD (resposta: LOGICA).

Agora tente quebrar a cifra sem saber o deslocamento: pegue uma frase longa cifrada e conte qual letra aparece mais vezes. Em português, é provavelmente o A ou o E. Essa técnica, chamada análise de frequência, é exatamente o tipo de trabalho repetitivo que as máquinas fariam muito melhor do que as pessoas.

Capítulo 12: Von Neumann e o programa armazenado

Na manhã de 21 de junho de 1948, num laboratório da Universidade de Manchester, na Inglaterra, um jovem engenheiro chamado Tom Kilburn digitou no teclado de uma máquina feita de válvulas e tubos de raios catódicos um programa de dezessete instruções. A máquina era apelidada de Baby, e parecia mais uma estante de laboratório do que um computador. A memória dela era um tubo de vídeo comum, parecido com os das televisões da época, onde pequenos pontos de luz guardavam zeros e uns. Kilburn apertou o botão. A máquina chiou, as luzes piscaram por quase uma hora, e então parou. No tubo, a resposta que ele procurava estava escrita em pontinhos. Era a primeira vez, na história, que uma máquina rodava um programa guardado na própria memória.

O problema do ENIAC

No capítulo anterior vimos que programar o ENIAC significava religar cabos e ajustar chaves, e que isso podia levar dias. Os engenheiros da Universidade da Pensilvânia sabiam que era um defeito grave. Se o programa pudesse ser guardado numa memória, como os números, bastaria carregar outro programa para a máquina mudar de tarefa, sem tocar num só fio. Eckert e Mauchly já pensavam nisso, e começaram a projetar um sucessor para o ENIAC, o EDVAC, que teria essa propriedade.

O relatório de von Neumann

Foi nessa época que entrou em cena John von Neumann, um matemático húngaro de brilho assombroso, professor em Princeton, que conhecia o projeto por ser consultor do governo. Em junho de 1945 ele escreveu um relatório de cerca de cem páginas, intitulado, em tradução livre, *Primeiro Rascunho de um Relatório sobre o EDVAC*, que descrevia com clareza a organização lógica de um computador de programa armazenado. O relatório circulou entre cientistas do mundo todo e deu a forma que quase todos os computadores seguem até hoje. A figura abaixo mostra essa organização: uma memória onde ficam programa e dados lado a lado, um processador com uma parte que controla e uma parte que calcula, e as entradas e saídas.

Programa e dados dividem a mesma memória

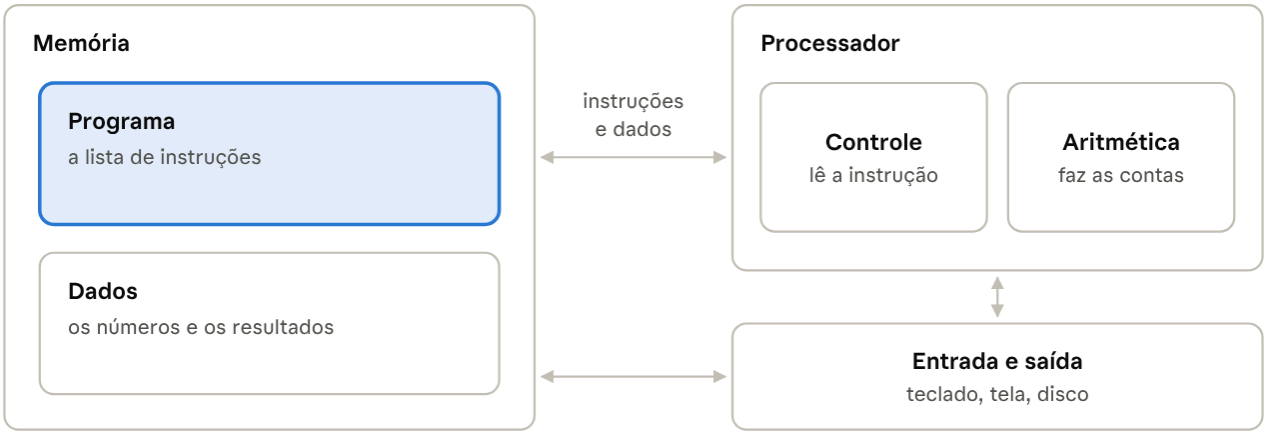


Figura 12.1 · a arquitetura de von Neumann

O nome que ficou, arquitetura de von Neumann, é injusto com os outros envolvidos. O relatório levava só o nome dele, e isso aborreceu Eckert e Mauchly, que sentiram que a ideia também era deles. Von Neumann conhecia o artigo de Turing de 1936 sobre a máquina universal, que também guarda o programa na fita junto com os dados, e os historiadores discutem até hoje o peso de cada influência. O mais honesto é dizer que a ideia estava no ar, e que von Neumann foi quem a escreveu de forma clara para o mundo.

Por que isso muda tudo

Guardar o programa na memória traz três consequências enormes. A primeira é velocidade de mudança: trocar de tarefa vira carregar um arquivo, e não religar uma sala. A segunda é que programas passam a ser dados, e portanto outros programas podem escrevê-los, copiá-los e modificá-los. Foi assim que nasceram os compiladores, os sistemas operacionais e as linguagens de programação. A terceira é que o computador passa a ser, na prática, a máquina universal de Turing em carne e osso: a mesma máquina faz qualquer coisa, desde que haja um programa.

As primeiras máquinas a rodar programas guardados

O primeiro computador a rodar um programa guardado foi o Baby, da cena de abertura: um protótipo pequeno, construído em Manchester por Frederic Williams, Tom Kilburn e Geoff Tootill, para testar o novo tipo de memória de tubos de raios catódicos. Seu primeiro programa, escrito por Kilburn, procurava o maior divisor de um número. Vinha com um resultado fácil de conferir, e por isso ideal para provar que a máquina funcionava.

Em maio de 1949, em Cambridge, o engenheiro Maurice Wilkes pôs para funcionar o EDSAC, que, diferentemente do Baby, era uma máquina completa e prática, usada por cientistas de toda a universidade. O EDVAC, o projeto original da Pensilvânia, só ficou pronto alguns anos depois, no começo dos anos 1950, e a corrida já tinha sido ganha por outros.

Na prática: um programa de quatro instruções

A tabela abaixo mostra uma pequena memória, com endereços numerados. Nos endereços 0 a 3 estão as instruções de um programa. Nos endereços 4 a 6 estão os dados. O processador lê as instruções na ordem, uma por vez, e faz o que cada uma manda. Siga com o dedo e descubra o resultado.

Endereço	Conteúdo	O que significa
0	LER 4	pegue o número guardado no endereço 4
1	SOMAR 5	some a ele o número do endereço 5
2	GUARDAR 6	guarde o resultado no endereço 6
3	PARAR	termine
4	7	um dado
5	5	um dado
6	0	aqui vai o resultado

O programa pega 7, soma 5 e guarda 12 no endereço 6. Agora o ponto principal: as instruções e os números estão na mesma tabela. Se você trocar o conteúdo do endereço 1 de SOMAR 5 para SOMAR 4, o programa muda sem tocar em nenhum fio. Foi isso que mudou o mundo em 1948.

Capítulo 13: Do relé à válvula, ao transistor e ao chip: as portas lógicas

Numa tarde de dezembro de 1947, numa sala dos Laboratórios Bell, em Nova Jersey, três físicos se debruçavam sobre um aparelhinho que parecia uma gambiarra: um pedaço de cristal de germânio, um clipe de papel torcido, duas finas laminárias de ouro e alguns fios. Conectaram o aparelho a uma pilha, mandaram um sinal fraco por um dos lados e viram, no outro, o mesmo sinal sair muito mais forte. Os três, John Bardeen, Walter Brattain e William Shockley, tinham acabado de provar o transistor. Eles não sabiam, mas era o fim de uma era que começara com os relés e o começo de outra, a que vivemos agora.

A peça muda, a ideia fica

Um fato curioso atravessa este livro. A lógica de Boole, o circuito de Shannon e a máquina de Turing não mudaram nada de essencial em oitenta anos. O que mudou foi a peça que faz o papel do interruptor. Um interruptor, aqui, é qualquer coisa que tenha dois estados, ligado e desligado, e que possa ser comandada por outro sinal. Qualquer peça assim serve para fazer um E, um OU ou um NÃO. A história da computação é, em grande parte, a história de trocar essa peça por outra menor, mais rápida e mais barata.

O relé, a válvula e o transistor

O primeiro interruptor eletrônico de uso geral foi o relé, que vimos no capítulo 8: um ímã elétrico puxando uma lâmina. É confiável, mas lento, porque uma lâmina metálica leva tempo para se mexer, e barulhento.

A válvula eletrônica nasceu da lâmpada. Em 1904, o engenheiro britânico John Ambrose Fleming percebeu que uma lâmpada com dois eletrodos deixava a corrente passar num só sentido, e a chamou de díodo. Em 1906, o americano Lee de Forest acrescentou um terceiro eletrodo, uma grade, e criou o tríodo. Com ele, uma tensão pequena na grade controla uma corrente grande, e pode ligar e desligar o fluxo sem nenhuma peça móvel. Como não há nada para se mover, a válvula é milhares de vezes mais rápida que o relé. Mas esquenta, consome muita energia, ocupa espaço e queima.

O transistor resolveu tudo isso. Feito de um material semicondutor, como o germânio e depois o silício, ele faz o papel da válvula sem filamento, sem vácuo e sem calor excessivo, num tamanho muito menor. Bardeen, Brattain e Shockley ganharam o prêmio Nobel de Física por ele, em 1956.

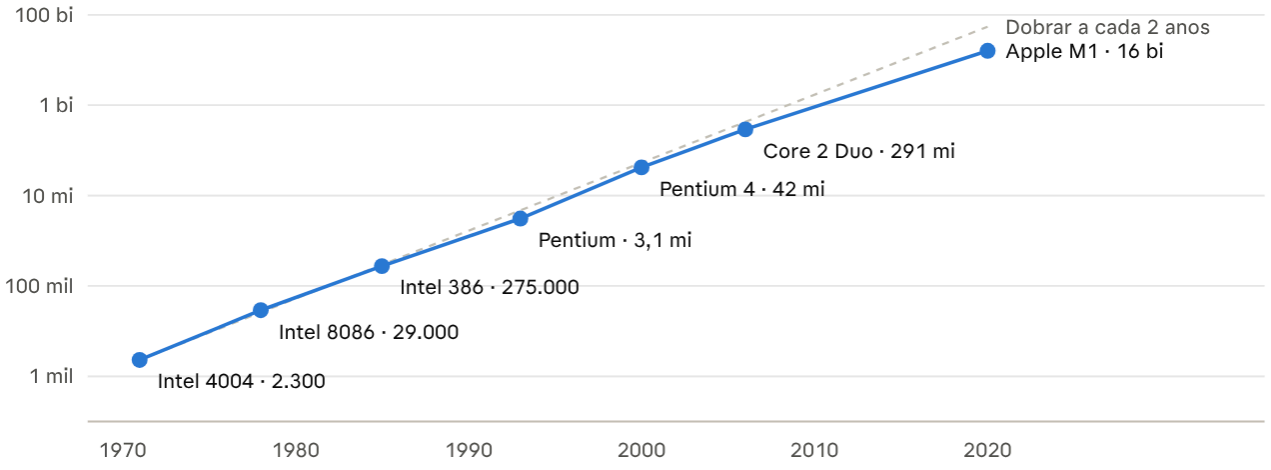
O circuito integrado e o microprocessador

Os primeiros transistores eram peças soltas que tinham de ser ligadas a fio, uma a uma. Em 1958, o engenheiro Jack Kilby, da Texas Instruments, teve a ideia de fabricar vários componentes juntos, num mesmo pedaço de material. No ano seguinte, Robert Noyce, da Fairchild, aperfeiçoou a técnica de modo que ela pudesse ser produzida em escala. Nascia o circuito integrado, o chip.

Em 1971, uma equipe da empresa Intel pôs um processador inteiro num único chip, o 4004, com cerca de 2.300 transistores. O microprocessador abriu as portas para os computadores pessoais, e desde então o número de transistores que cabem num chip não parou de crescer. Em 1965, o engenheiro Gordon Moore, que depois cofundou a Intel, observou que esse número dobrava a cada ano ou dois, e a regra ficou conhecida como lei de Moore. O gráfico no roteiro deste capítulo, mais acima, mostra o resultado: de milhares de transistores em 1971 para dezenas de bilhões hoje.

O número de transistores por chip dobrou a cada cerca de dois anos

Em média, cerca de 2,2 anos para dobrar, de 1971 a 2020 (escala logarítmica)



valores aproximados, a conferir nas fontes · chips de 1971 a 2020

A linha pontilhada mostra o que aconteceria se o número dobrasse exatamente a cada dois anos, e os chips reais ficaram perto dela por quase cinco décadas, e só nos últimos anos ficaram um pouco abaixo.

As portas lógicas

Dentro de um chip, a unidade básica é a porta lógica: um pequeno circuito de poucos transistores que implementa uma operação de Boole. A figura abaixo mostra as mais importantes. As três primeiras são as velhas conhecidas, E, OU e NÃO. A quarta, a NAND, faz um E e depois inverte o resultado. Ela merece atenção, porque, apenas com portas NAND, é possível construir todas as outras, e por isso os fabricantes costumam usá-la como peça-padrão.

Quatro portas lógicas e o símbolo de cada uma



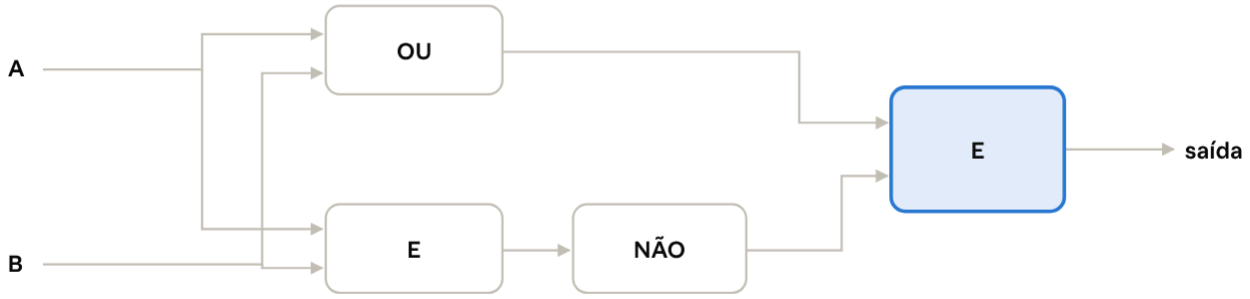
E	OU	NÃO	NAND
1 só se os dois são 1	1 se algum é 1	inverte o valor	0 só se os dois são 1

A porta NAND sozinha basta
Combinando só portas NAND, dá para construir o E, o OU, o NÃO e qualquer outro circuito.

Figura 13.1 · símbolos das portas lógicas

Com as três portas básicas dá para montar qualquer circuito. A figura seguinte mostra, como exemplo, o OU exclusivo, que vimos no somador do capítulo 8, montado só com OU, E e NÃO. A fórmula é simples de ler: a saída vale 1 quando A ou B valem 1 (a porta OU), mas não quando os dois valem 1 (a porta E seguida do NÃO). A última porta E junta as duas condições.

O OU exclusivo se monta com as três portas básicas



Saída = (A OU B) E NÃO (A E B): vale 1 quando só um dos dois vale 1.

Figura 13.2 · OU exclusivo montado com E, OU e NÃO

Na prática: confira o circuito linha a linha

Para cada combinação de A e B, siga o sinal pela figura 13.2 e anote o que sai de cada porta. O resultado deve bater com a coluna final.

A	B	A OU B	A E B	NÃO (A E B)	Saída
0	0	0	0	1	0
0	1	1	0	1	1

A	B	A OU B	A E B	NÃO (A E B)	Saída
1	0	1	0	1	1
1	1	1	1	0	0

O circuito que você acabou de conferir faz o papel da porta “OU exclusivo” do somador do capítulo 8. Junte a ele uma porta E, que dá o “vai um”, e você tem um somador de um bit feito só de portas básicas. Repita isso bilhões de vezes, com outras combinações de portas, e você tem o processador de um celular.

Capítulo 14: No Brasil, os primeiros computadores brasileiros

> **Atenção:** este capítulo foi escrito de memória, sem acesso à pesquisa, e trata de fatos locais que exigem conferência cuidadosa em fontes brasileiras. Datas, nomes e instituições estão listados no fim para você validar antes de publicar. Prefira, ao revisar, os acervos da USP, do ITA, do Museu da Computação e da Sociedade Brasileira de Computação.

A história que contamos até aqui passou por Inglaterra, Alemanha e Estados Unidos, e o leitor brasileiro pode ter achado que o computador é um assunto de outros países. Este capítulo mostra que não foi bem assim. O Brasil recebeu os primeiros computadores quando eles ainda eram novidade mundial, formou engenheiros que aprenderam a construí-los e, nos anos 1970, chegou a projetar máquinas próprias, com decisões políticas que até hoje dividem opiniões.

Os primeiros computadores chegam

O primeiro computador a funcionar no Brasil foi importado, como acontecia em quase todo o mundo naquela época. A tradição registra que, em 1957, uma máquina da linha UNIVAC foi instalada em São Paulo, para uso de uma grande empresa, e que, nos anos seguintes, governos e universidades compraram outras para calcular censos, folhas de pagamento e projetos de engenharia. O país era, naquele momento, um mercado de compradores, sem fábrica de computadores.

O que torna esse período interessante é o que aconteceu nas universidades. Engenheiros e físicos que cuidavam das máquinas importadas começaram a perguntar como elas eram feitas por dentro, e alguns decidiram construir as suas próprias.

Zezinho e o Patinho Feio

Uma das primeiras tentativas registradas é a do Instituto Tecnológico de Aeronáutica, o ITA, em São José dos Campos, onde, no começo dos anos 1960, um grupo de professores e alunos teria construído um pequeno computador digital apelidado de Zezinho. A história é contada com carinho em várias fontes, mas detalhes como autores, tecnologia e data exata precisam de conferência.

O marco mais conhecido é de 1972, na Escola Politécnica da USP, em São Paulo. Uma equipe de professores e alunos, num projeto que juntou ensino e pesquisa, desenhou e montou um computador de oito bits com circuitos integrados, que ficou famoso pelo apelido de Patinho Feio. O nome era uma brincadeira com a história do filhote feio que cresce e vira cisne: a máquina era simples e modesta, mas foi o primeiro computador projetado e construído no país.

O Patinho Feio é um bom lugar para lembrar o que você aprendeu. Por dentro, ele era feito das mesmas peças dos capítulos anteriores: portas E, OU e NÃO ligadas em circuitos, uma memória, uma unidade de controle e uma aritmética, seguindo a organização de von Neumann da figura 12.1. A álgebra de 1854 estava ali, em tamanho de laboratório.

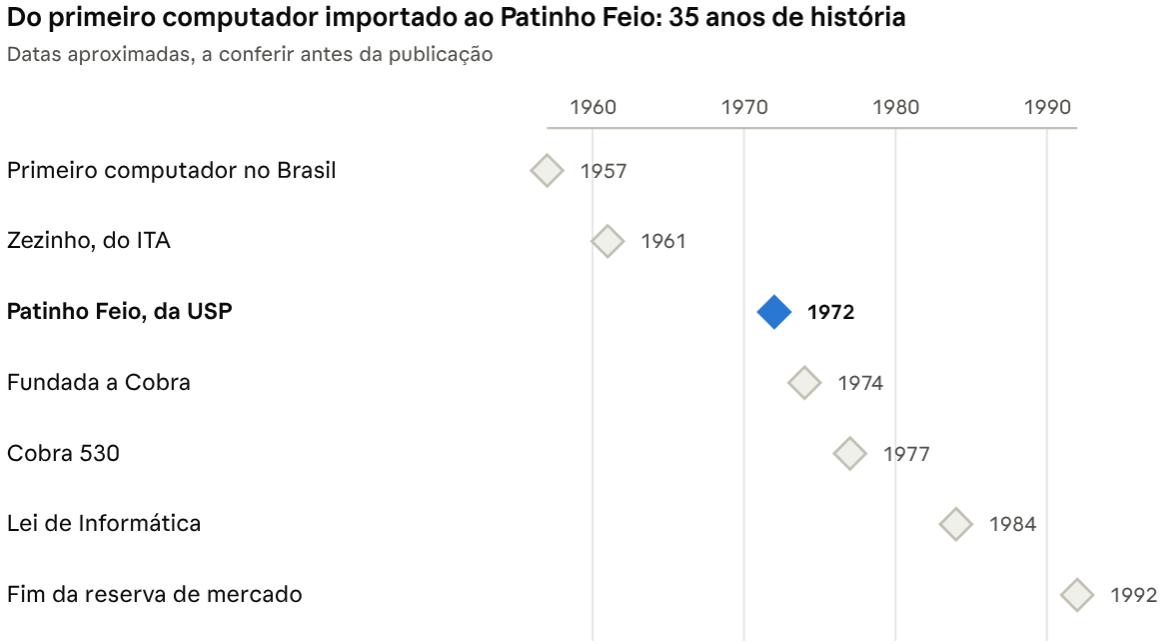


Figura 14.1 · marcos da computação brasileira, 1957 a 1992 (datas a conferir)

A Cobra e a reserva de mercado

O sucesso das universidades chamou a atenção do governo. Em 1974 foi criada a Cobra, uma empresa estatal de computadores, e, em 1977, ela lançou o Cobra 530, um dos primeiros computadores fabricados por uma empresa brasileira. Na mesma década, a política oficial passou a restringir a importação de computadores e componentes, para

proteger a indústria nacional nascente. Esse conjunto de regras ficou conhecido como a reserva de mercado de informática, formalizada pela Lei de Informática de 1984 e encerrada por volta de 1992.

Os historiadores e economistas discutem o resultado. Para uns, a reserva criou uma indústria, formou milhares de técnicos e deixou o país menos dependente. Para outros, ela deixou os computadores brasileiros mais caros e atrasados em relação ao mundo, e os usuários sofreram. Quem viveu os anos 1980 lembra dos micros nacionais, que muita gente usava para jogar e aprender a programar em casa. O leitor pode formar a sua opinião, mas é bom lembrar que o assunto não tem resposta simples.

Uma tabela para guardar

Marco	Onde	Quando (aprox.)	Por que importa
Primeiro computador instalado	São Paulo	1957	o país entra na era dos computadores eletrônicos
Zezinho	ITA, São José dos Campos	início dos anos 1960	primeira tentativa de construir um computador digital no país
Patinho Feio	Escola Politécnica da USP	1972	primeiro computador projetado e construído no Brasil
Cobra	empresa estatal	1974	indústria nacional de computadores
Cobra 530	empresa Cobra	1977	computador fabricado por empresa brasileira
Lei de Informática	Governo federal	1984	formaliza a reserva de mercado

Na prática: converse com quem viveu

Peça a uma pessoa mais velha da sua família que conte quando viu um computador pela primeira vez e em que lugar. Pergunte também se ela chegou a usar um micro nacional dos anos 1980. Anote o relato com o nome da pessoa, a cidade e o ano. Registros assim, que costumam ficar só na memória de família, são parte da história da computação brasileira, e você pode colocar o seu como depoimento no livro.

Capítulo 15: A lógica no bolso, a álgebra booleana hoje

Você está num café, com o celular na mesa. Quer achar uma receita de bolo, mas não quer chocolate. Digita no buscador: receita de bolo, um espaço, um sinal de menos e a palavra chocolate. Aperta a tecla. Em menos de um segundo, milhares de páginas são examinadas e as que falam de chocolate desaparecem da lista. Você não percebeu, mas acabou de escrever uma fórmula de álgebra booleana: receita E bolo E NÃO chocolate. Em 1854, um professor de Cork escreveu algo muito parecido num livro que quase ninguém leu.

Três palavras em toda parte

O E, o OU e o NÃO não ficaram nos livros de lógica. Estão em quase tudo o que você faz com um computador. Quando você marca os filtros de uma loja online, escolhendo cor azul E tamanho M, está usando um E. Quando um programa decide mostrar um aviso se a bateria está baixa OU o aparelho está quente, usa um OU. Quando um aplicativo não deixa você avançar a menos que o campo NÃO esteja vazio, usa um NÃO.

Em quase toda linguagem de programação existe a construção “se ... então”, com condições feitas de E, OU e NÃO. Nos bancos de dados, a linguagem usada para consultar informações, chamada SQL, permite pedir “clientes de São Paulo E com compras acima de mil reais, mas NÃO inativos”. E, no nível mais baixo, dentro do chip, cada transistor é uma peça de uma porta lógica, como vimos no capítulo 13. A figura abaixo mostra a mesma frase de Boole escrita de três maneiras diferentes, e a álgebra é a mesma.

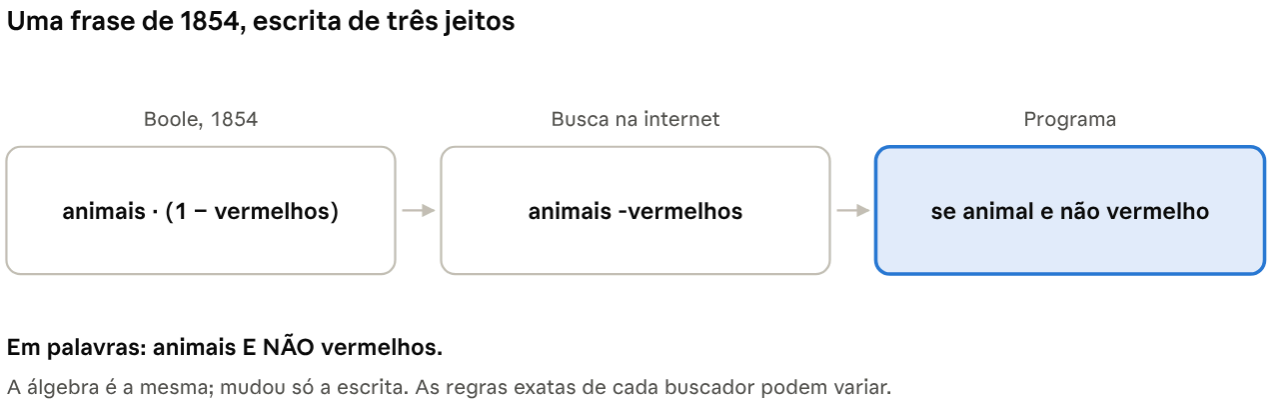


Figura 15.1 · a mesma frase em três notações

O que mudou, o que não mudou

O que mudou foi quase tudo na escala: de uma pena sobre papel a dezenas de bilhões de transistores num chip, e de um professor num escritório a bilhões de pessoas usando lógica sem saber. O que não mudou foi a ideia central de Boole: que afirmações de sim ou não podem ser combinadas por regras exatas, como se fossem números de 0 e 1.

E a inteligência artificial?

Uma pergunta natural é se a inteligência artificial atual, que escreve textos, reconhece imagens e conversa, tem a ver com tudo isso. A resposta tem duas partes, e vale a pena não simplificar demais.

Por um lado, os sistemas mais famosos de hoje não são feitos de regras lógicas escritas à mão, do tipo “se isso, então aquilo”. São redes neurais, que aprendem padrões a partir de enormes quantidades de exemplos, e ninguém consegue apontar um passo lógico simples que explique cada decisão delas. Nesse sentido, o sonho de Leibniz, de uma linguagem em que todo raciocínio vira um cálculo transparente, não foi realizado do jeito que ele imaginou.

Por outro lado, essas redes rodam em computadores comuns, feitos das mesmas portas lógicas de sempre. Por baixo das contas de somar e multiplicar que formam uma rede neural, há números guardados como zeros e uns, e, por baixo deles, transistores ligados e desligados, como mostra a figura abaixo. A álgebra de Boole continua lá, em silêncio, segurando tudo.

Por baixo da rede neural, ainda são zeros e uns



Figura 15.2 · da rede neural aos zeros e uns

A pergunta que abriu o livro

Voltemos a Leibniz, àquele envelope de 1701. Ele perguntava se pensar pode ser calculado. Ao longo destas páginas, vimos Boole mostrar que parte do pensamento, o de sim ou não, pode ser tratado como álgebra. Vimos Shannon transformar essa álgebra em circuitos, e Turing descrever, com uma fita de papel, tudo o que é calculável e também o que não é. Vimos engenheiros trocarem relés por válvulas, válvulas por transistores, e hoje carregamos no bolso o que antes ocupava um prédio.

A resposta honesta à pergunta de Leibniz é que ela continua aberta. Sabemos que muitas coisas que pareciam pertencer só à mente humana, como jogar xadrez, traduzir idiomas ou escrever um poema simples, podem ser feitas por máquinas. Não sabemos se tudo o que é pensar pode ser, e há pessoas sérias dos dois lados do debate. O que este livro lhe dá, espero, é o caminho que levou até a pergunta e as ferramentas para participar da discussão.

E tudo isso começou com um menino de Lincoln, filho de um sapateiro, que não tinha dinheiro para a universidade e achava que as palavras, de algum jeito, podiam ser contas.

Na prática: reescreva três buscas como fórmulas

Reescreva cada busca abaixo com E, OU e NÃO, e adivinhe, antes de testar, o que o buscador vai mostrar.

- Receita de bolo, sem chocolate. (Resposta: receita E bolo E NÃO chocolate.)
- Filmes de comédia ou de aventura, sem terror. (Resposta: filme E (comédia OU aventura) E NÃO terror.)
- Um texto que tenha a frase “álgebra de Boole” ou “lógica booleana”. (Resposta: “álgebra de Boole” OU “lógica booleana”.)

Depois teste no seu buscador preferido. As regras exatas (aspas, sinal de menos, a palavra OR em maiúsculas) variam de um buscador para outro, e algumas já mudaram com o tempo. Anote o que funcionou: esse é um ótimo exercício de curiosidade.

Linha do tempo e glossário

Em 268 anos, da ideia binária de Leibniz ao primeiro microprocessador, a história do livro cabe em 18 marcos, mais o bicentenário de Boole. As datas abaixo seguem a ordem dos capítulos e estão na lista de verificação no fim.

Ano	Marco	Capítulo
1703	Leibniz descreve o sistema binário	1
1804	Jacquard usa cartões perfurados num tear	7
1815	Nasce George Boole, em Lincoln	2
1837	Babbage projeta a Máquina Analítica	6
1843	Ada Lovelace publica suas notas sobre a máquina	6
1847	Boole publica *The Mathematical Analysis of Logic*	3
1854	Boole publica *The Laws of Thought*	3

Ano	Marco	Capítulo
1880	Venn apresenta seus diagramas	5
1890	Hollerith apura o censo dos EUA com cartões	7
1936	Turing publica o artigo da máquina universal	9
1937	Shannon defende a tese sobre relés e álgebra booleana	8
1941	Zuse conclui o Z3	10
1943	Colossus começa a funcionar em Bletchley Park	11
1945	ENIAC fica pronto; von Neumann escreve o relatório do EDVAC	11 e 12
1947	Transistor nos Laboratórios Bell	13
1948	Manchester Baby roda o primeiro programa guardado; Shannon publica a teoria da informação	12 e 8
1958	Kilby apresenta o circuito integrado	13
1971	Intel lança o 4004	13
2015	O bicentenário de Boole	14

Termo	O que é
Bit	A menor informação possível: um 0 ou um 1
Porta lógica	Circuito que faz uma operação E, OU ou NÃO com bits
Tabela-verdade	Quadro que lista o resultado de uma operação para cada combinação de 0 e 1
Relé	Interruptor ligado e desligado por um ímã elétrico
Válvula	Tubo de vidro que controla corrente elétrica sem peças móveis
Transistor	Interruptor minúsculo feito de material semicondutor
Algoritmo	Receita de passos que leva sempre ao mesmo resultado

Termo	O que é
Programa armazenado	Programa guardado na mesma memória que os dados

Fontes

Obras e instituições de referência para este livro.

Textos originais

- Boole, George. *The Mathematical Analysis of Logic* (1847) e *An Investigation of the Laws of Thought* (1854).
- Leibniz, Gottfried Wilhelm. "Explication de l'arithmétique binaire" (1703).
- Jevons, William Stanley. "On the Mechanical Performance of Logical Inference" (1870).
- Venn, John. "On the Diagrammatic and Mechanical Representation of Propositions and Reasoning" (1880).
- Menabrea, Luigi; Lovelace, Ada. *Sketch of the Analytical Engine, with notes by the translator* (1843).
- Shannon, Claude. "A Symbolic Analysis of Relay and Switching Circuits" (dissertação de mestrado, MIT, 1937) e "A Mathematical Theory of Communication" (1948).
- Turing, Alan. "On Computable Numbers, with an Application to the Entscheidungsproblem" (1936).
- von Neumann, John. "First Draft of a Report on the EDVAC" (1945).

Livros

- MacHale, Desmond. *George Boole: His Life and Work* (1985).
- Hodges, Andrew. *Alan Turing: The Enigma* (1983).
- Isaacson, Walter. *Os Inovadores* (*The Innovators*, 2014).
- Gleick, James. *A Informação* (*The Information*, 2011).
- Petzold, Charles. *Code: The Hidden Language of Computer Hardware and Software*.
- Davis, Martin. *The Universal Computer: The Road from Leibniz to Turing*.
- Dyson, George. *Turing's Cathedral*.

Museus e acervos

- Computer History Museum (Califórnia), para ENIAC, UNIVAC, transistor e circuito integrado.

- Bletchley Park Trust, para Enigma, Bombe e Colossus.
- Science Museum (Londres), para as máquinas de Babbage.
- Museum of Science and Industry (Manchester), para o Manchester Baby.
- MacTutor History of Mathematics (Universidade de St Andrews), para as biografias de Boole, De Morgan, Venn e outros.

Brasil (capítulo 14)

- Sociedade Brasileira de Computação (SBC), com os textos de história da computação no país.
- Arquivos da USP (Escola Politécnica) e do ITA, sobre o Patinho Feio e o Zezinho.
- Documentos da Cobra Computadores e da legislação de informática de 1984.