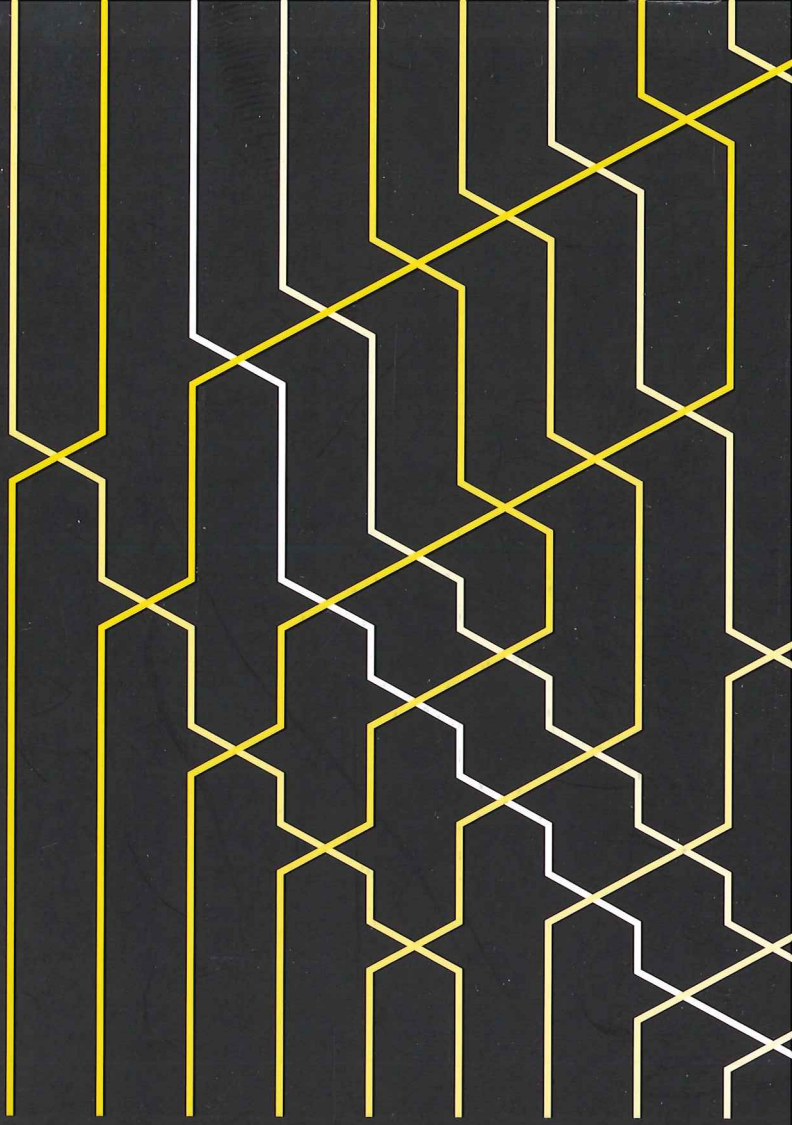

ALGORITHM
TO LIVE B

THE COMPUTER SCIENCE OF HUMAN DECISION

BRIAN CHRISTIAN
& TOM GRIFFITHS

'Practical and highly enjoyable'
POPULAR SCIENCE

'A wonderful book'

DAVID EAGLEMAN, author of *Sum: Tales from the Afterlives*

WHAT SHOULD WE DO, OR LEAVE UNDONE, IN A DAY OR A LIFETIME?

Algorithms to Live By helps to solve common decision-making problems and illuminate the workings of the human mind. Asking us how to have better hunches, when to leave things to chance, how to deal with overwhelming choices or how best to connect with others – it shows how our computers' methods have much to teach us. From finding your spouse to finding a parking spot, and from organizing your inbox to understanding the workings of memory – where you have a dilemma, they have a rule. In this eye-opening book each fascinating algorithm turns the wisdom of computer science into strategies for human living.

'A remarkable book ... more applicable to real-life problems than I'd have ever predicted' *FORBES*

'By the end of the book, I was convinced ... algorithms could be a surprisingly useful way to embrace the messy compromises of real, non-Vulcan life'

OLIVER BURKEMAN, *Guardian*

WilliamCollinsBooks.com

ISBN 978-0-00-754799-9

£9.99

9 780007 547999 >

Cover design by Jonathan Pelham

Contents

Introduction — 1
Algorithms to Live By

1 Optimal Stopping — 9
When to Stop Looking

2 Explore/Exploit — 31
The Latest vs. the Greatest

3 Sorting — 59
Making Order

4 Caching — 84
Forget About It

5 Scheduling — 105
First Things First

6 Bayes's Rule — 128
Predicting the Future

7 Overfitting — 149
When to Think Less

8 Relaxation 169
Let It Slide

9 Randomness 182
When to Leave It to Chance

10 Networking 205
How We Connect

11 Game Theory 229
The Minds of Others

Conclusion 256
Computational Kindness

Notes 263

Bibliography 315

Acknowledgments 335

Index 339

Introduction

Algorithms to Live By

Imagine you're searching for an apartment in San Francisco—arguably the most harrowing American city in which to do so. The booming tech sector and tight zoning laws limiting new construction have conspired to make the city just as expensive as New York, and by many accounts more competitive. New listings go up and come down within minutes, open houses are mobbed, and often the keys end up in the hands of whoever can physically foist a deposit check on the landlord first.

Such a savage market leaves little room for the kind of fact-finding and deliberation that is theoretically supposed to characterize the doings of the rational consumer. Unlike, say, a mall patron or an online shopper, who can compare options before making a decision, the would-be San Franciscan has to decide instantly either way: you can take the apartment you are currently looking at, forsaking all others, or you can walk away, never to return.

Let's assume for a moment, for the sake of simplicity, that you care only about maximizing your chance of getting the very best apartment available. Your goal is reducing the twin, Scylla-and-Charybdis regrets of the "one that got away" and the "stone left unturned" to the absolute minimum. You run into a dilemma right off the bat: How are you to know that an apartment is indeed the best unless you have a baseline to judge it by? And how are you to establish that baseline unless you look at (and *lose*) a number of apartments? The more information you gather, the better you'll

know the right opportunity when you see it—but the more likely you to have already passed it by.

So what do you do? How do you make an informed decision when very act of informing it jeopardizes the outcome? It's a cruel situat bordering on paradox.

When presented with this kind of problem, most people will intuiti say something to the effect that it requires some sort of balance betw looking and leaping—that you must look at enough apartments to estab a standard, then take whatever satisfies the standard you've establis This notion of balance is, in fact, precisely correct. What most people *c* say with any certainty is what that balance is. Fortunately, there's an ans

Thirty-seven percent.

If you want the best odds of getting the best apartment, spend 37 your apartment hunt (eleven days, if you've given yourself a month fo search) noncommittally exploring options. Leave the checkbook at h you're just calibrating. But after that point, be prepared to immedi commit—deposit and all—to the very first place you see that beats w ever you've already seen. This is not merely an intuitively satisfying c promise between looking and leaping. It is the *provably optimal* solut

We know this because finding an apartment belongs to a cla mathematical problems known as "optimal stopping" problems. The rule defines a simple series of steps—what computer scientists ca "algorithm"—for solving these problems. And as it turns out, apart hunting is just one of the ways that optimal stopping rears its head in life. Committing to or forgoing a succession of options is a structure appears in life again and again, in slightly different incarnations. many times to circle the block before pulling into a parking sp How far to push your luck with a risky business venture before cas out? How long to hold out for a better offer on that house or car?

The same challenge also appears in an even more fraught set dating. Optimal stopping is the science of serial monogamy.

Simple algorithms offer solutions not only to an apartment hunt b all such situations in life where we confront the question of optimal ping. People grapple with these issues every day—although surely have spilled more ink on the tribulations of courtship than of park and they do so with, in some cases, considerable anguish. But the an is unnecessary. Mathematically, at least, these are solved problems.

Every harried renter, driver, and suitor you see around you as y

through a typical week is essentially reinventing the wheel. They don't need a therapist; they need an algorithm. The therapist tells them to find the right, comfortable balance between impulsivity and overthinking.

The algorithm tells them the balance is thirty-seven percent.

◆

There is a particular set of problems that all people face, problems that are a direct result of the fact that our lives are carried out in finite space and time. What should we do, and leave undone, in a day or in a decade? What degree of mess should we embrace—and how much order is excessive? What balance between *new* experiences and *favored* ones makes for the most fulfilling life?

These might seem like problems unique to humans; they're not. For more than half a century, computer scientists have been grappling with, and in many cases solving, the equivalents of these everyday dilemmas. How should a processor allocate its "attention" to perform all that the user asks of it, with the minimum overhead and in the least amount of time? When should it switch between different tasks, and how many tasks should it take on in the first place? What is the best way for it to use its limited memory resources? Should it collect more data, or take an action based on the data it already has? Seizing the day might be a challenge for humans, but computers all around us are seizing milliseconds with ease. And there's much we can learn from how they do it.

Talking about algorithms for human lives might seem like an odd juxtaposition. For many people, the word "algorithm" evokes the arcane and inscrutable machinations of big data, big government, and big business: increasingly part of the infrastructure of the modern world, but hardly a source of practical wisdom or guidance for human affairs. But an algorithm is just a finite sequence of steps used to solve a problem, and algorithms are much broader—and older by far—than the computer. Long before algorithms were ever used by machines, they were used by people.

The word "algorithm" comes from the name of Persian mathematician al-Khwārizmī, author of a ninth-century book of techniques for doing mathematics by hand. (His book was called *al-Jabr wa'l-Muqābala*—and the "al-jabr" of the title in turn provides the source of our word "algebra.") The earliest known mathematical algorithms, however, predate even al-Khwārizmī's work: a four-thousand-year-old Sumerian clay tablet found near Baghdad describes a scheme for long division.

But algorithms are not confined to mathematics alone. When you
bread from a recipe, you're following an algorithm. When you knit a sw
from a pattern, you're following an algorithm. When you put a sl
edge on a piece of flint by executing a precise sequence of strikes witl
end of an antler—a key step in making fine stone tools—you're follo
an algorithm. Algorithms have been a part of human technology ever
the Stone Age.

◆

In this book, we explore the idea of *human algorithm design*—searc
for better solutions to the challenges people encounter every day. A
ing the lens of computer science to everyday life has consequences at
scales. Most immediately, it offers us practical, concrete suggestion
how to solve specific problems. Optimal stopping tells us when to look
when to leap. The explore/exploit tradeoff tells us how to find the bal
between trying new things and enjoying our favorites. Sorting theory
us how (and whether) to arrange our offices. Caching theory tells us
to fill our closets. Scheduling theory tells us how to fill our time.

At the next level, computer science gives us a vocabulary for u
standing the deeper principles at play in each of these domains. As
Sagan put it, "Science is a way of thinking much more than it is a bo
knowledge." Even in cases where life is too messy for us to expect a
numerical analysis or a ready answer, using intuitions and concepts h
on the simpler forms of these problems offers us a way to understan
key issues and make progress.

Most broadly, looking through the lens of computer science can
us about the nature of the human mind, the meaning of rationality
the oldest question of all: how to live. Examining cognition as a mea
solving the fundamentally computational problems posed by our env
ment can utterly change the way we think about human rationality.

The notion that studying the inner workings of computers might
how to think and decide, what to believe and how to behave, might
many people as not only wildly reductive, but in fact misguided. E
computer science did have things to say about how to think and h
act, would we want to listen? We look at the AIs and robots of scien
tion, and it seems like theirs is not a life any of us would want to live

In part, that's because when we think about computers, we think
coldly mechanical, deterministic systems: machines applying rigid d

tive logic, making decisions by exhaustively enumerating the options, and grinding out the exact right answer no matter how long and hard they have to think. Indeed, the person who first imagined computers had something essentially like this in mind. Alan Turing defined the very notion of computation by an analogy to a human mathematician who carefully works through the steps of a lengthy calculation, yielding an unmistakably right answer.

So it might come as a surprise that this is not what modern computers are actually doing when they face a difficult problem. Straightforward arithmetic, of course, isn't particularly challenging for a modern computer. Rather, it's tasks like conversing with people, fixing a corrupted file, or winning a game of Go—problems where the rules aren't clear, some of the required information is missing, or finding exactly the right answer would require considering an astronomical number of possibilities—that now pose the biggest challenges in computer science. And the algorithms that researchers have developed to solve the hardest classes of problems have moved computers away from an extreme reliance on exhaustive calculation. Instead, tackling real-world tasks requires being comfortable with chance, trading off time with accuracy, and using approximations.

As computers become better tuned to real-world problems, they provide not only algorithms that people can borrow for their own lives, but a better standard against which to compare human cognition itself. Over the past decade or two, behavioral economics has told a very particular story about human beings: that we are irrational and error-prone, owing in large part to the buggy, idiosyncratic hardware of the brain. This self-deprecating story has become increasingly familiar, but certain questions remain vexing. Why are four-year-olds, for instance, still better than million-dollar supercomputers at a host of cognitive tasks, including vision, language, and causal reasoning?

The solutions to everyday problems that come from computer science tell a different story about the human mind. Life is full of problems that are, quite simply, *hard*. And the mistakes made by people often say more about the intrinsic difficulties of the problem than about the fallibility of human brains. Thinking algorithmically about the world, learning about the fundamental structures of the problems we face and about the properties of their solutions, can help us see how good we actually are, and better understand the errors that we make.

In fact, human beings turn out to consistently confront some of the

hardest cases of the problems studied by computer scientists. Often, p need to make decisions while dealing with uncertainty, time constra partial information, and a rapidly changing world. In some of t cases, even cutting-edge computer science has not yet come up with cient, always-right algorithms. For certain situations it appears that algorithms might not exist at all.

Even where perfect algorithms haven't been found, however, the b between generations of computer scientists and the most intractable world problems has yielded a series of insights. These hard-won pre are at odds with our intuitions about rationality, and they don't sound thing like the narrow prescriptions of a mathematician trying to forc world into clean, formal lines. They say: Don't always consider all options. Don't necessarily go for the outcome that seems best every Make a mess on occasion. Travel light. Let things wait. Trust your inst and don't think too long. Relax. Toss a coin. Forgive, but don't forge thine own self be true.

Living by the wisdom of computer science doesn't sound so bad all. And unlike most advice, it's backed up by proofs.

Just as designing algorithms for computers was originally a subject th into the cracks between disciplines—an odd hybrid of mathematic engineering—so, too, designing algorithms for humans is a topic doesn't have a natural disciplinary home. Today, algorithm design c not only on computer science, math, and engineering but on kir fields like statistics and operations research. And as we consider how rithms designed for machines might relate to human minds, we also to look to cognitive science, psychology, economics, and beyond.

We, your authors, are familiar with this interdisciplinary terr Brian studied computer science and philosophy before going on to g ate work in English and a career at the intersection of the three. studied psychology and statistics before becoming a professor at UC B ley, where he spends most of his time thinking about the relatio between human cognition and computation. But nobody can be an e in all of the fields that are relevant to designing better algorithm humans. So as part of our quest for algorithms to live by, we talked t people who came up with some of the most famous algorithms of th fifty years. And we asked them, some of the smartest people in the w

how their research influenced the way they approached their own lives—from finding their spouses to sorting their socks.

The next pages begin our journey through some of the biggest challenges faced by computers and human minds alike: how to manage finite space, finite time, limited attention, unknown unknowns, incomplete information, and an unforeseeable future; how to do so with grace and confidence; and how to do so in a community with others who are all simultaneously trying to do the same. We will learn about the fundamental mathematical structure of these challenges and about how computers are engineered—sometimes counter to what we imagine—to make the most of them. And we will learn about how the mind works, about its distinct but deeply related ways of tackling the same set of issues and coping with the same constraints. Ultimately, what we can gain is not only a set of concrete takeaways for the problems around us, not only a new way to see the elegant structures behind even the hairiest human dilemmas, not only a recognition of the travails of humans and computers as deeply conjoined, but something even more profound: a new vocabulary for the world around us, and a chance to learn something truly new about ourselves.

1 Optimal Stopping

When to Stop Looking

Though all Christians start a wedding invitation by solemnly declaring their marriage is due to special Divine arrangement, I, as a philosopher, would like to talk in greater detail about this . . .

—JOHANNES KEPLER

If you prefer Mr. Martin to every other person; if you think him the most agreeable man you have ever been in company with, why should you hesitate?

—JANE AUSTEN, *EMMA*

It's such a common phenomenon that college guidance counselors even have a slang term for it: the "turkey drop." High-school sweethearts come home for Thanksgiving of their freshman year of college and, four days later, return to campus single.

An angst-ridden Brian went to his own college guidance counselor his freshman year. His high-school girlfriend had gone to a different college several states away, and they struggled with the distance. They also struggled with a stranger and more philosophical question: how good a relationship did they have? They had no real benchmark of other relationships by which to judge it. Brian's counselor recognized theirs as a classic freshman-year dilemma, and was surprisingly nonchalant in her advice: "Gather data."

The nature of serial monogamy, writ large, is that its practitioners are confronted with a fundamental, unavoidable problem. When have you met

2 Explore/Exploit

The Latest vs. the Greatest

Your stomach rumbles. Do you go to the Italian restaurant that you know and love, or the new Thai place that just opened up? Do you take your best friend, or reach out to a new acquaintance you'd like to get to know better? This is too hard—maybe you'll just stay home. Do you cook a recipe that you know is going to work, or scour the Internet for new inspiration? Never mind, how about you just order a pizza? Do you get your "usual," or ask about the specials? You're already exhausted before you get to the first bite. And the thought of putting on a record, watching a movie, or reading a book—*which one?*—no longer seems quite so relaxing.

Every day we are constantly forced to make decisions between options that differ in a very specific dimension: do we try new things or stick with our favorite ones? We intuitively understand that life is a balance between novelty and tradition, between the latest and the greatest, between taking risks and savoring what we know and love. But just as with the look-or-leap dilemma of the apartment hunt, the unanswered question is: what balance?

In the 1974 classic *Zen and the Art of Motorcycle Maintenance*, Robert Pirsig decries the conversational opener "What's new?"—arguing that the question, "if pursued exclusively, results only in an endless parade of trivia and fashion, the silt of tomorrow." He endorses an alternative as vastly superior: "What's best?"

But the reality is not so simple. Remembering that every "best" song and restaurant among your favorites began humbly as something merely

3 Sorting

Making Order

Nowe if the word, which thou art desirous to finde, begin with (a) then looke in the beginning of this Table, but if with (v) looke towards the end. Againe, if thy word beginne with (ca) looke in the beginning of the letter (c) but if with (cu) then looke toward the end of that letter. And so of all the rest. &c.

—ROBERT CAWDREY, *A TABLE ALPHABETICALL* (1604)

Before Danny Hillis founded the Thinking Machines corporation, before he invented the famous Connection Machine parallel supercomputer, he was an MIT undergraduate, living in the student dormitory, and horrified by his roommate's socks.

What horrified Hillis, unlike many a college undergraduate, wasn't his roommate's hygiene. It wasn't that the roommate didn't *wash* the socks; he did. The problem was what came next.

The roommate pulled a sock out of the clean laundry hamper. Next he pulled another sock out at random. If it didn't match the first one, he tossed it back in. Then he continued this process, pulling out socks one by one and tossing them back until he found a match for the first.

With just 10 different pairs of socks, following this method will take on average *19* pulls merely to complete the first pair, and 17 more pulls to complete the second. In total, the roommate can expect to go fishing in the hamper 110 times just to pair 20 socks.

4 Caching

Forget About It

In the practical use of our intellect, forgetting is as important a functio
remembering.

—WILLIAM JAMES

You have a problem. Your closet is overflowing, spilling shoes, shirts,
underwear onto the floor. You think, "It's time to get organized." Now
have two problems.

Specifically, you first need to decide what to keep, and second, ho
arrange it. Fortunately, there is a small industry of people who think a
these twin problems for a living, and they are more than happy to
their advice.

On what to keep, Martha Stewart says to ask yourself a few quest
"How long have I had it? Does it still function? Is it a duplicate of s
thing I already own? When was the last time I wore it or used it?" On
to organize what you keep, she recommends "grouping like t
together," and her fellow experts agree. Francine Jay, in *The Joy of*
stipulates, "Hang all your skirts together, pants together, dresses tog
and coats together." Andrew Mellen, who bills himself as "The
Organized Man in America," dictates, "Items will be sorted by typ
slacks together, shirts together, coats, etc. Within each type, they're f
sorted by color and style—long-sleeved or short-sleeved, by nec
etc." Other than the sorting problem this could entail, it looks lik
advice; it certainly seems unanimous.

5 Scheduling

First Things First

How we spend our days is, of course, how we spend our lives.

—ANNIE DILLARD

"Why don't we write a book on scheduling theory?" I asked. . . . "It shouldn't take much time!" Book-writing, like war-making, often entails grave miscalculations. Fifteen years later, Scheduling *is still unfinished.*

—EUGENE LAWLER

It's Monday morning, and you have an as-yet blank schedule and a long list of tasks to complete. Some can be started only after others are finished (you can't load the dishwasher unless it's unloaded first), and some can be started only after a certain time (the neighbors will complain if you put the trash out on the curb before Tuesday night). Some have sharp deadlines, others can be done whenever, and many are fuzzily in between. Some are urgent, but not important. Some are important, but not urgent. "We are what we repeatedly do," you seem to recall Aristotle saying—whether it's mop the floor, spend more time with family, file taxes on time, learn French.

So what to do, and when, and in what order? Your life is waiting.

Though we always manage to find *some* way to order the things we do in our days, as a rule we don't consider ourselves particularly good at it—hence the perennial bestseller status of time-management guides. Unfortunately, the guidance we find in them is frequently divergent and inconsistent. *Getting Things Done* advocates a policy of immediately doing

6 Bayes's Rule

Predicting the Future

All human knowledge is uncertain, inexact, and partial.

—BERTRAND RUSSELL

The sun'll come out tomorrow. You can bet your bottom dollar there'll be s

—*ANNIE*

In 1969, before embarking on a doctorate in astrophysics at Princet J. Richard Gott III took a trip to Europe. There he saw the Berlin W which had been built eight years earlier. Standing in the shadow of wall, a stark symbol of the Cold War, he began to wonder how much l ger it would continue to divide the East and West.

On the face of it, there's something absurd about trying to make t kind of prediction. Even setting aside the impossibility of forecast geopolitics, the question seems mathematically laughable: it's trying make a prediction from *a single data point.*

But as ridiculous as this might seem on its face, we make such pre tions all the time, by necessity. You arrive at a bus stop in a foreign and learn, perhaps, that the other tourist standing there has been wai seven minutes. When is the next bus likely to arrive? Is it worthwhil wait—and if so, how long should you do so before giving up?

Or perhaps a friend of yours has been dating somebody for a mo and wants your advice: is it too soon to invite them along to an upcom

7 Overfitting

When to Think Less

When Charles Darwin was trying to decide whether he should propose to his cousin Emma Wedgwood, he got out a pencil and paper and weighed every possible consequence. In favor of marriage he listed children, companionship, and the "charms of music & female chit-chat." Against marriage he listed the "terrible loss of time," lack of freedom to go where he wished, the burden of visiting relatives, the expense and anxiety provoked by children, the concern that "perhaps my wife won't like London," and having less money to spend on books. Weighing one column against the other produced a narrow margin of victory, and at the bottom Darwin scrawled, "Marry—Marry—Marry Q.E.D." *Quod erat demonstrandum*, the mathematical sign-off that Darwin himself then restated in English: "It being proved necessary to Marry."

The pro-and-con list was already a time-honored algorithm by Darwin's time, being endorsed by Benjamin Franklin a century before. To get over "the Uncertainty that perplexes us," Franklin wrote,

> my Way is, divide half a Sheet of Paper by a Line into two Columns, writing over the one Pro, and over the other Con. Then during three or four Days Consideration I put down under the different Heads short Hints of the different Motives that at different Times occur to me for or against the Measure. When I have thus got them all together in one View, I endeavour to estimate their respective Weights; and where I find two, one on each side, that seem equal, I strike them both out: If I find a Reason pro equal to

8 Relaxation

Let It Slide

In 2010 Meghan Bellows was working on her PhD in chemical engineering at Princeton by day and planning her wedding by night. Her research revolved around finding the right places to put amino acids in a protein chain to yield a molecule with particular characteristics. ("If you maximize the binding energy of two proteins then you can successfully design a peptidic inhibitor of some biological function so you can actually inhibit a disease's progress.") On the nuptial front, she was stuck on the problem of seating.

There was a group of nine college friends, and Bellows agonized over who else to throw into the midst of such a mini-reunion to make a table of ten. Even worse, she'd counted up eleven close relatives. Who would get the boot from the honored parents' table, and how could she explain it to them? And what about folks like her childhood neighbors and babysitter, or her parents' work colleagues, who didn't really know anyone at the wedding at all?

The problem seemed every bit as hard as the protein problem she was working on at the lab. Then it hit her. It *was* the problem she was working on at the lab. One evening, as she stared at her seating charts, "I realized that there was literally a one-to-one correlation between the amino acids and proteins in my PhD thesis and people sitting at tables at my wedding." Bellows called out to her fiancé for a piece of paper and began scribbling equations. Amino acids became guests, binding energies became relationships, and the molecules' so-called nearest-neighbor

9 Randomness

When to Leave It to Chance

I must admit that after many years of work in this area, the efficacy of ra
domness for so many algorithmic problems is absolutely mysterious to n
It is efficient, it works; but why and how is absolutely mysterious.

—MICHAEL RABIN

Randomness seems like the opposite of reason—a form of giving up
a problem, a last resort. Far from it. The surprising and increasin
important role of randomness in computer science shows us that m
ing use of chance can be a deliberate and effective part of approach
the hardest sets of problems. In fact, there are times when nothing
will do.

In contrast to the standard "deterministic" algorithms we typic
imagine computers using, where one step follows from another in exa
the same way every time, a randomized algorithm uses randomly ge
ated numbers to solve a problem. Recent work in computer science
shown that there are cases where randomized algorithms can prod
good approximate answers to difficult questions faster than all kn
deterministic algorithms. And while they do not always guarantee
optimal solutions, randomized algorithms can get surprisingly clos
them in a fraction of the time, just by strategically flipping a few c
while their deterministic cousins sweat it out.

There is a deep message in the fact that on certain problems, random
approaches can outperform even the best deterministic ones. Sometime

10 Networking

How We Connect

The term connection *has a wide variety of meanings. It can refer to a physical or logical path between two entities, it can refer to the flow over the path, it can inferentially refer to an action associated with the setting up of a path, or it can refer to an association between two or more entities, with or without regard to any path between them.*

—VINT CERF AND BOB KAHN

Only connect.

—E. M. FORSTER

The long-distance telegraph began with a portent—Samuel F. B. Morse, standing in the chambers of the US Supreme Court on May 24, 1844, wiring his assistant Alfred Vail in Baltimore a verse from the Old Testament: "WHAT HATH GOD WROUGHT." The first thing we ask of any new connection is how it began, and from that origin can't help trying to augur its future.

The first telephone call in history, made by Alexander Graham Bell to his assistant on March 10, 1876, began with a bit of a paradox. "Mr. Watson, come here; I want to see you"—a simultaneous testament to its ability *and* inability to overcome physical distance.

The cell phone began with a boast—Motorola's Martin Cooper walking down Sixth Avenue on April 3, 1973, as Manhattan pedestrians gawked, calling his rival Joel Engel at AT&T: "Joel, I'm calling you from a cellular

11 Game Theory

The Minds of Others

I'm an optimist in the sense that I believe humans are noble and honorable, and some of them are really smart. . . . I have a somewhat more pessimistic view of people in groups.

—STEVE JOBS

An investor sells a stock to another, one convinced it's headed down and the other convinced it's going up; I think I know what you think but have no idea what you think I think; an economic bubble bursts; a prospective lover offers a gift that says neither "I want to be more than friends" nor "I don't want to be more than friends"; a table of diners squabbles over who should treat whom and why; someone trying to be helpful unintentionally offends; someone trying hard to be cool draws snickers; someone trying to break from the herd finds, dismayingly, the herd following his lead. "I love you," says one lover to another; "I love you, too," the other replies; and both wonder what exactly the other means by that.

What does computer science have to say about all this?

Schoolchildren are taught to conceive of literary plots as belonging to one of several categories: man vs. nature, man vs. self, man vs. man, man vs. society. Thus far in this book we have considered primarily cases in the first two categories—that is to say, computer science has thus far been our guide to problems created by the fundamental structure of the world, and by our limited capacities for processing information. Optimal stopping problems spring from the irreversibility and irrevocability of time; the

Conclusion

Computational Kindness

I firmly believe that the important things about humans are so character and that relief by machines from many of our present de ing intellectual functions will finally give the human race time and tive to learn how to live well together.

—MERRILL FLOC

Any dynamic system subject to the constraints of space and ti up against a core set of fundamental and unavoidable problems. problems are computational in nature, which makes computers no our tools but also our comrades. From this come three simple pie wisdom.

First, there are cases where computer scientists and mathemat have identified good algorithmic approaches that can simply be trans over to human problems. The 37% Rule, the Least Recently Used cri for handling overflowing caches, and the Upper Confidence Boun guide to exploration are all examples of this.

Second, knowing that you are using an optimal algorithm shou relief even if you don't get the results you were looking for. The 37 fails 63% of the time. Maintaining your cache with LRU doesn't gua that you will always find what you're looking for; in fact, neither clairvoyance. Using the Upper Confidence Bound approach to the ex exploit tradeoff doesn't mean that you will have *no* regrets, just that regrets will accumulate ever more slowly as you go through life. Ev